Guide d'administration d'Oracle® VM Server for SPARC 3.4

Référence: E71807
Août 2016

Référence: E71807

Copyright © 2007, 2016, Oracle et/ou ses affiliés. Tous droits réservés.

Ce logiciel et la documentation qui l'accompagne sont protégés par les lois sur la propriété intellectuelle. Ils sont concédés sous licence et soumis à des restrictions d'utilisation et de divulgation. Sauf stipulation expresse de votre contrat de licence ou de la loi, vous ne pouvez pas copier, reproduire, traduire, diffuser, modifier, accorder de licence, transmettre, distribuer, exposer, exécuter, publier ou afficher le logiciel, même partiellement, sous quelque forme et par quelque procédé que ce soit. Par ailleurs, il est interdit de procéder à toute ingénierie inverse du logiciel, de le désassembler ou de le décompiler, excepté à des fins d'interopérabilité avec des logiciels tiers ou tel que prescrit par la loi.

Les informations fournies dans ce document sont susceptibles de modification sans préavis. Par ailleurs, Oracle Corporation ne garantit pas qu'elles soient exemptes d'erreurs et vous invite, le cas échéant, à lui en faire part par écrit.

Si ce logiciel, ou la documentation qui l'accompagne, est livré sous licence au Gouvernement des Etats-Unis, ou à quiconque qui aurait souscrit la licence de ce logiciel pour le compte du Gouvernement des Etats-Unis, la notice suivante s'applique :

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

Ce logiciel ou matériel a été développé pour un usage général dans le cadre d'applications de gestion des informations. Ce logiciel ou matériel n'est pas conçu ni n'est destiné à être utilisé dans des applications à risque, notamment dans des applications pouvant causer un risque de dommages corporels. Si vous utilisez ce logiciel ou matériel dans le cadre d'applications dangereuses, il est de votre responsabilité de prendre toutes les mesures de secours, de sauvegarde, de redondance et autres mesures nécessaires à son utilisation dans des conditions optimales de sécurité. Oracle Corporation et ses affiliés déclinent toute responsabilité quant aux dommages causés par l'utilisation de ce logiciel ou matériel pour des applications dangereuses.

Oracle et Java sont des marques déposées d'Oracle Corporation et/ou de ses affiliés. Tout autre nom mentionné peut correspondre à des marques appartenant à d'autres propriétaires qu'Oracle.

Intel et Intel Xeon sont des marques ou des marques déposées d'Intel Corporation. Toutes les marques SPARC sont utilisées sous licence et sont des marques ou des marques déposées de SPARC International, Inc. AMD, Opteron, le logo AMD et le logo AMD Opteron sont des marques ou des marques déposées d'Advanced Micro Devices. UNIX est une marque déposée de The Open Group.

Ce logiciel ou matériel et la documentation qui l'accompagne peuvent fournir des informations ou des liens donnant accès à des contenus, des produits et des services émanant de tiers. Oracle Corporation et ses affiliés déclinent toute responsabilité ou garantie expresse quant aux contenus, produits ou services émanant de tiers, sauf mention contraire stipulée dans un contrat entre vous et Oracle. En aucun cas, Oracle Corporation et ses affiliés ne sauraient être tenus pour responsables des pertes subies, des coûts occasionnés ou des dommages causés par l'accès à des contenus, produits ou services tiers, ou à leur utilisation, sauf mention contraire stipulée dans un contrat entre vous et Oracle.

Accessibilité de la documentation

Pour plus d'informations sur l'engagement d'Oracle pour l'accessibilité à la documentation, visitez le site Web Oracle Accessibility Program, à l'adresse http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Accès aux services de support Oracle

Les clients Oracle qui ont souscrit un contrat de support ont accès au support électronique via My Oracle Support. Pour plus d'informations, visitez le site http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info ou le site http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs si vous êtes malentendant.

Table des matières

Utilisation de cette documentation

- **Présentation** – Fournit aux administrateurs système du SE Oracle Solaris des procédures et des informations détaillées sur l'installation, la configuration et l'utilisation du logiciel Oracle VM Server for SPARC 3.4.
- **Public visé** – Des administrateurs système qui gèrent la virtualisation sur des serveurs SPARC.
- **Connaissances nécessaires** – Les administrateurs système de ces serveurs doivent posséder des connaissances fonctionnelles des systèmes UNIX et du système d'exploitation Oracle Solaris (SE Oracle Solaris).

Bibliothèque de documentation produit

La documentation et les ressources de ce produit et des produits associés sont disponibles sur le site Web http://www.oracle.com/technetwork/documentation/vm-sparc-194287.html.

Commentaires

Faites part de vos commentaires sur cette documentation à l'adresse : http://www.oracle.com/goto/docfeedback.

♦ ♦ ♦ CHAPITRE 1

Présentation du logiciel Oracle VM Server for SPARC

Ce chapitre fournit une présentation du logiciel Oracle VM Server for SPARC.

Oracle VM Server for SPARC offre des fonctions de virtualisation de qualité professionnelle très efficaces pour les serveurs Oracle des séries SPARC T-, SPARC M- et SPARC S- ainsi que les Plates-formes Fujitsu M10. Grâce au logiciel Oracle VM Server for SPARC, vous pouvez créer jusqu'à 128 serveurs virtuels, appelés domaines logiques, sur un seul et même système. Ce type de configuration permet de bénéficier de la puissance d'exécution offerte par les serveurs SPARC des séries T, M et S, les Plates-formes Fujitsu M10 et le SE Oracle Solaris.

Ce chapitre aborde les sujets suivants :

Remarque - Les fonctions décrites dans ce manuel peuvent être utilisées avec l'ensemble des plates-formes matérielles et logiciels système pris en charge, répertoriés dans le *Guide d'installation d'Oracle VM Server for SPARC 3.4*. Cependant, certaines fonctions sont uniquement disponibles sur un sous-ensemble du logiciel système et des plates-formes matérielles pris en charge. Pour plus d'informations sur ces exceptions, consultez "Nouveautés dans cette version" du manuel *Notes de version d'Oracle VM Server for SPARC 3.4* et What's New in Oracle VM Server for SPARC Software (`http://www.oracle.com/technetwork/server-storage/vm/documentation/sparc-whatsnew-330281.html`).

A propos des versions d'Oracle VM Server for SPARC et d'SE Oracle Solaris

Le logiciel Oracle VM Server for SPARC dépend de versions spécifiques du SE Oracle Solaris, des patchs logiciels requis et de versions spécifiques du microprogramme du système. Pour plus d'informations, reportez-vous à la section "Versions du Système d’exploitation Oracle Solaris complètes" du manuel *Guide d’installation d’Oracle VM Server for SPARC 3.4*.

La version du SE Oracle Solaris qui s'exécute sur un domaine invité est *indépendante* de la version du SE Oracle Solaris qui s'exécute dans le domaine `primary`. Donc, si vous exécutez le SE Oracle Solaris 11 dans le domaine `primary`, vous pouvez tout de même exécuter le SE Oracle Solaris 10 dans un domaine invité.

Remarque - Le système d'exploitation Oracle Solaris 10 n'est plus pris en charge dans le domaine `primary`. Vous pouvez continuer d'exécuter le système d'exploitation Oracle Solaris 10 dans les domaines invités.

Hyperviseur et Logical Domains

Cette section présente l'hyperviseur SPARC qui prend en charge les domaines logiques.

L'*hyperviseur*SPARC est une petite couche de microprogramme qui fournit une architecture de machine virtualisée stable sur laquelle on peut installer un système d'exploitation. Les serveurs SPARC utilisant cet hyperviseur fournissent des fonctions matérielles qui prennent en charge le contrôle de l'hyperviseur sur les activités du système d'exploitation logique.

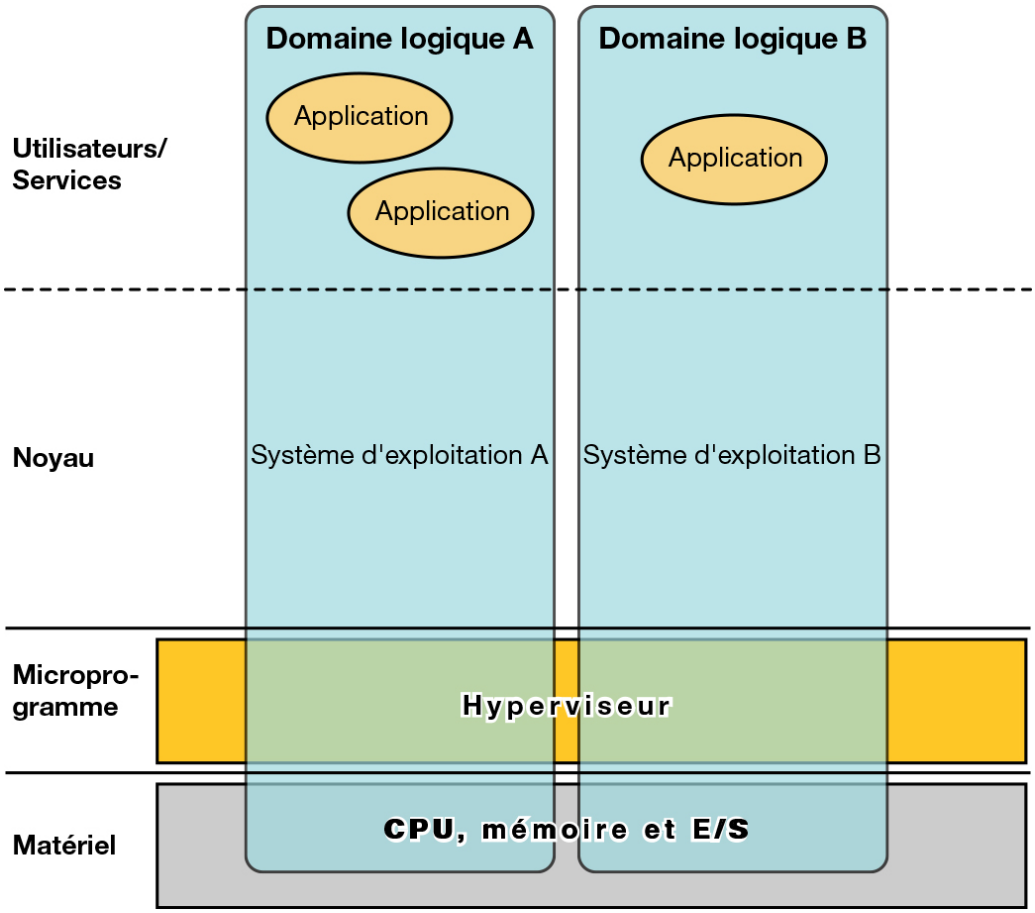
Un *domaine logique* est une machine virtuelle comprenant un groupement logique discret de ressources. Un domaine logique a son propre système d'exploitation et sa propre identité dans un système informatique unique. Chaque domaine logique peut être créé, supprimé, reconfiguré et réinitialisé individuellement, sans nécessiter de cycle d'alimentation du serveur. Il est possible d'exécuter une grande variété d'applications dans des domaines logiques différents et de préserver l'indépendance de ceux-ci à des fins de performances ou de sécurité.

Chaque domaine logique est uniquement autorisé à observer et à interagir avec les ressources du serveur qui sont mises à sa disposition par l'hyperviseur. Logical Domains Manager vous permet de spécifier ce que l'hyperviseur doit faire dans le domaine logique. Ainsi, l'hyperviseur force le partitionnement des ressources du serveur et fournit des sous-ensembles limités à plusieurs environnements de système d'exploitation. Ce partitionnement et cette mise à disposition sont le mécanisme fondamental de création des domaines logiques. Le schéma

suivant représente l'hyperviseur prenant en charge deux domaines logiques. Il montre également les couches suivantes constituant la fonctionnalité d'Oracle VM Server for SPARC :

- Utilisateurs/services (applications)
- Noyau (systèmes d'exploitation)
- Microprogramme (hyperviseur)
- Matériel, y compris la CPU, la mémoire et les E/S

FIGURE 1 Hyperviseur prenant en charge deux domaines



Le nombre et les fonctions de chaque domaine logique qu'un hyperviseur SPARC spécifique prend en charge sont des fonctions dépendantes du serveur. L'hyperviseur peut allouer des sous-ensembles des ressources globales de CPU, de mémoire et d'E/S d'un serveur à un domaine logique donné. Cette fonctionnalité permet la prise en charge simultanée de plusieurs systèmes d'exploitation, chacun dans son propre domaine logique. Les ressources peuvent être réorganisées entre des domaines logiques distincts avec un niveau de précision quelconque. Il est par exemple possible d'assigner des CPU à un domaine logique avec une précision de l'ordre du thread de CPU.

Chaque domaine logique peut être géré comme une machine totalement indépendante avec ses propres ressources, notamment :

- Le noyau, les patchs et les paramètres de réglage
- Les comptes utilisateur et les administrateurs
- Les disques
- Les interfaces réseau, les adresses MAC et IP

Chaque domaine logique peut être arrêté, démarré et réinitialisé indépendamment des autres sans nécessiter de cycle d'alimentation du serveur.

Le logiciel de l'hyperviseur est responsable du maintien de la séparation entre les domaines logiques. Le logiciel de l'hyperviseur fournit également les canaux de domaine logique (LDC) qui permettent aux domaines logiques de communiquer les uns avec les autres. Les LCD permettent aux domaines de se fournir des services mutuellement, notamment des services de mise en réseau ou de disque.

Le processeur de service (SP), également connu sous le nom de contrôleur système (SC), surveille et exécute la machine virtuelle , mais il ne gère pas les domaines logiques. Les domaines logiques sont gérés par Logical Domains Manager.

En outre l'utilisation de la commande `ldm` pour gérer le logiciel Oracle VM Server for SPARC, vous pouvez désormais utiliser Oracle VM Manager.

Oracle VM Manager est une interface utilisateur Web permettant de gérer l'environnement Oracle VM. Les versions précédentes de cette interface utilisateur permettent uniquement de gérer le logiciel Oracle VM Server x86, mais à partir de Oracle VM Manager 3.2 et Oracle VM Server for SPARC 3.0, vous pouvez également gérer le logiciel Oracle VM Server for SPARC. Pour plus d'informations sur Oracle VM Manager, reportez-vous à la Oracle VM Documentation (`http://www.oracle.com/technetwork/documentation/vm-096300.html`).

Logical Domains Manager

Logical Domains Manager est utilisé pour créer et gérer les domaines logiques, ainsi que pour mapper les domaines logiques sur des ressources physiques. Une seule instance de Logical Domains Manager peut s'exécuter sur un serveur.

Rôles des domaines

Tous les domaines logiques sont identiques et peuvent se différencier les uns des autres en fonction de rôles que vous définissez pour eux. Les domaines logiques peuvent jouer les rôles suivants :

- **Domaine de contrôle.** Logical Domains Manager s'exécute dans ce domaine, ce qui vous permet de créer et de gérer d'autres domaines logiques et d'allouer des ressources virtuelles aux autres domaines. Il ne peut y avoir qu'un seul domaine de contrôle par serveur. Le domaine de contrôle est le premier domaine créé lorsque vous installez le logiciel Oracle VM Server for SPARC. Le domaine de contrôle est nommé `primary`.
- **Domaine de service.** Un domaine de service fournit des services de périphérique virtuel aux autres domaines, notamment en tant que commutateur virtuel, concentrateur de console virtuelle ou serveur de disque virtuel. Vous pouvez avoir plus d'un domaine de service, et tout domaine peut être configuré comme un domaine de service.
- **Domaine d'E/S.** Un domaine d'E/S dispose d'un accès direct à un périphérique d'E/S physique tel qu'une carte réseau dans un contrôleur PCI EXPRESS (PCIe). Un domaine d'E/S peut posséder les éléments suivants :
  - Un complexe root PCIe.
  - Un emplacement PCIe ou un périphérique PCIe intégré à l'aide de la fonction d'E/S directes (DIO). Voir la section "Création d'un domaine d'E/S en assignant les périphériques d'extrémité PCIe" à la page 155.
  - Une fonction virtuelle SR-IOV PCIe. Reportez-vous à la section Chapitre 8, Création d'un domaine d'E/S à l'aide des fonctions virtuelles SR-IOV PCIe.

  Un domaine d'E/S peut partager des périphériques d'E/S physiques avec d'autres domaines sous la forme de périphériques virtuels lorsque le domaine d'E/S est également utilisé en tant que domaine de service.
- **Domaine root.** Un domaine root a un complexe root PCIe qui lui est assigné. Ce domaine possède la topologie Fabric PCIe et fournit tous les services associés à la Fabric, notamment le traitement des erreurs Fabric. Un domaine root est également un domaine d'E/S, car il possède et a un accès direct aux périphériques d'E/S.

Le nombre de domaines root que vous pouvez avoir dépend de l'architecture de votre plate-forme. Par exemple, si vous utilisez un serveur Oracle SPARC T5-8 à huit sockets, vous pouvez avoir jusqu'à 16 domaines root.

Le domaine root par défaut est `primary`. Vous pouvez utiliser des domaines non-`primary` en tant que domaines root.

- **Domaine invité.** Un domaine invité est un domaine non E/S qui consomme des services de périphérique virtuel fournis par un ou plusieurs domaines de service. Un domaine invité n'a pas de périphérique d'E/S physiques, mais a uniquement des périphériques d'E/S virtuels, tels que des disques virtuels et des interfaces réseau virtuelles.

Vous pouvez installer Logical Domains Manager sur un système existant qui n'est pas déjà configuré avec Oracle VM Server for SPARC. Dans ce cas, l'instance actuelle du SE devient le domaine de contrôle. En outre, le système est configuré avec un domaine unique, le domaine de contrôle. Après la configuration du domaine de contrôle, vous pouvez équilibrer la charge des applications au sein des autres domaines pour une utilisation optimale de tout le système en ajoutant des domaines et en déplaçant ces applications du domaine de contrôle vers les nouveaux domaines.

Interface de ligne de commande

Logical Domains Manager utilise une interface de ligne de commande (CLI) pour créer et configurer des domaines logiques. La CLI est une commande simple, `ldm`, ayant plusieurs sous-commandes. Reportez-vous à la page de manuel `ldm`(1M).

Le démon de Logical Domains Manager, `ldmd`, doit être en cours d'exécution pour utiliser la CLI de Logical Domains Manager

Entrée/sortie virtuelle

Dans un environnement Oracle VM Server for SPARC, vous pouvez configurer jusqu'à 128 domaines sur un système (jusqu'à 256 sur un Serveur Fujitsu M10). Certains serveurs, en particulier les systèmes monoprocesseurs et certains systèmes biprocesseurs, disposent d'un nombre limité de bus d'E/S et d'emplacements d'E/S physiques. En conséquence, vous risquez de ne pas pouvoir fournir un accès exclusif à des périphériques physiques de disque ou réseau à tous les domaines sur ces systèmes. Vous pouvez assigner un bus PCIe ou un périphérique d'extrémité à un domaine pour lui fournir l'accès à un périphérique physique. Notez que cette solution est insuffisante pour fournir un accès exclusif au périphérique à tous les domaines. Cette limitation sur le nombre de périphériques d'E/S physiques à accès direct est traitée en implémentant un modèle d'E/S virtualisé. Reportez-vous à la section Chapitre 6, Configuration de domaines d'E/S.

Les domaines logiques n'ayant pas d'accès E/S physique sont configurés avec des périphériques d'E/S virtuels communiquant avec un domaine de service. Le domaine de service exécute le service de périphérique virtuel pour fourni l'accès au périphérique physique ou à ses fonctions. Dans ce modèle client-serveur, les périphériques d'E/S virtuels communiquent les uns avec les autres ou avec un service homologue via les canaux de communication interdomaine appelés canaux de domaine logique (LDC). La fonctionnalité d'E/S virtualisée comprend la prise en charge de la mise en réseau, du stockage et des consoles virtuels.

Réseau virtuel

Oracle VM Server for SPARC utilise le périphérique réseau virtuel et le périphérique de commutation de réseau virtuel pour implémenter la mise en réseau virtuelle. Le périphérique réseau virtuel (`vnet`) émule un périphérique Ethernet et communique avec les autres périphériques `vnet` dans le système à l'aide d'un canal point à point. Le périphérique de commutateur virtuel (`vsw`) fonctionne comme un multiplexeur de tous les paquets entrants et sortants du réseau virtuel. Le périphérique `vsw` communique directement avec un adaptateur réseau physique sur un domaine de service, et envoie et reçoit des paquets pour le compte d'un réseau virtuel. Le périphérique `vsw` fonctionne également comme un commutateur simple de couche 2 et commute les paquets entre les périphériques `vnet` connectés à celui-ci dans le système.

Stockage virtuel

L'infrastructure de stockage virtuel utilise un modèle client-serveur pour activer les domaines logiques afin d'accéder au stockage au niveau du bloc qui n'est pas directement assigné à ces derniers. Le modèle utilise les composants suivants :

- Un client de disque virtuel (`vdc`), qui exporte l'interface de périphérique de bloc
- Un service de disque virtuel (`vds`) qui traite les demandes de disque pour le compte du client de disque virtuel et les envoie au stockage backend qui réside sur le domaine de service

Bien que les disques virtuels apparaissent comme des disques classiques sur le domaine client, la plupart des opérations de disque sont transmises au service de disque virtuel et traitées sur le domaine de service.

Console virtuelle

Dans un environnement Oracle VM Server for SPARC, la console d'E/S du domaine `primary` est dirigée vers le processeur de service. Les E/S de la console de tous les autres domaines sont redirigées vers le domaine de service exécutant le concentrateur de console virtuelle (`vcc`). Le

domaine exécutant le `vcc` est généralement le domaine `primary`. Le service de concentrateur de consoles virtuelles fonctionne comme un concentrateur pour le trafic de la console de tous les domaines et communique avec le démon du serveur de terminal réseau virtuel (`vntsd`) pour fournir l'accès à chaque console via un socket UNIX.

Configuration des ressources

Un système exécutant le logiciel Oracle VM Server for SPARC peut configurer des ressources, notamment des CPU virtuelles, des périphériques d'E/S virtuels, des unités cryptographiques et de la mémoire. Certaines ressources peuvent être configurées de manière dynamique sur un domaine en cours d'exécution, tandis que d'autres doivent être configurées sur un domaine arrêté. Si une ressource ne peut pas être configurée de manière dynamique sur le domaine de contrôle, vous devez d'abord lancer une reconfiguration retardée. La reconfiguration retardée reporte les activités de configuration à après la réinitialisation du domaine de contrôle. Pour plus d'informations, reportez-vous à la section "Reconfiguration des ressources" à la page 345.

Configurations persistantes

Vous pouvez utiliser la commande `ldm` pour stocker la configuration actuelle d'un domaine logique sur le processeur de service. Vous pouvez ajouter une configuration, spécifier la configuration à utiliser, supprimer une configuration et répertorier les configurations. Pour plus d'informations, reportez-vous à la page de manuel `ldm`(1M). Vous pouvez également indiquer une configuration pour démarrer à partir du SP, comme décrit dans la section "Utilisation d'Oracle VM Server for SPARC avec le processeur de service" à la page 412.

Pour plus d'informations sur la gestion des configurations, reportez-vous à la section "Gestion des configurations de domaine" à la page 389.

Base MIB (Management Information Base) Oracle VM Server for SPARC

Oracle VM Server for SPARC Management Information Base (MIB) permet aux applications tierces de gestion de système de surveiller à distance des domaines et de démarrer et d'arrêter des domaines logiques (domaines) via le protocole SNMP (Simple Network Management

Protocol). Pour plus d'informations, reportez-vous au *Guide d'utilisation d'Oracle VM Server for SPARC Management Information Base*.

Dépannage de Oracle VM Server for SPARC

Vous pouvez obtenir des informations sur des questions particulières concernant le logiciel Oracle VM Server for SPARC à partir des publications suivantes :

- "Problèmes connus" du manuel *Notes de version d'Oracle VM Server for SPARC 3.4*
- Information Center: Overview of Oracle VM Server for SPARC (LDoms) (Doc ID 1589473.2) (`https://mosemp.us.oracle.com/epmos/faces/DocumentDisplay?_afrLoop=227880986952919&id=1589473.2&_afrWindowMode=0&_adf.ctrl-state=wu098o5r6_96`)

♦ ♦ ♦ CHAPITRE 2

Sécurité d'Oracle VM Server for SPARC

Ce chapitre décrit certaines fonctions de sécurité que vous pouvez activer sur votre système Oracle VM Server for SPARC.

Ce chapitre aborde les sujets suivants :

Remarque - Les exemples de ce manuel sont présentés comme s'ils étaient réalisés par un superutilisateur. Cependant, les profils d'utilisateur permettent aux utilisateurs d'obtenir des autorisations plus détaillées pour effectuer les tâches de gestion.

Délégation de la gestion de Logical Domains à l'aide de droits

Le package Logical Domains Manager ajoute deux profils de droits prédéfinis à la configuration de droits locaux. Ces profils de droits délèguent des privilèges administratifs à des utilisateurs dépourvus de privilèges :

- Le profil `LDoms Management` (Gestion de domaines logiques) permet à un utilisateur d'utiliser toutes les sous-commandes `ldm`.
- Le profil `LDoms Review` (Vérification de domaines logiques) permet à un utilisateur d'utiliser toutes les sous-commandes liées aux listes de `ldm`.

Ces profils de droits peuvent être affectés directement à des utilisateurs ou ils peuvent être affectés à un rôle qui sera à son tour affecté à des utilisateurs. Lorsque l'un de ces profils est directement affecté à un utilisateur, vous devez utiliser la commande `pfexec` ou un shell de profil tel que `pfbash` ou `pfksh` pour utiliser correctement la commande `ldm` afin de gérer les domaines. Optez pour les rôles d'utilisateur ou les profils de droits en fonction de la configuration de vos droits. Reportez-vous au manuel *System Administration Guide: Security Services* or *Securing Users and Processes in Oracle Solaris 11.3*.

Les utilisateurs, autorisations, profils de droits et rôles peuvent être configurés de l'une des manières suivantes :

- De manière locale sur le système à l'aide de fichiers
- De manière centrale dans un service de noms tel que LDAP

L'installation de Logical Domains Manager ajoute les profils de droits et autorisations nécessaires dans les fichiers locaux. Pour configurer les profils et les rôles dans un service de noms, reportez-vous au manuel *System Administration Guide: Naming and Directory Services (DNS, NIS, and LDAP)*. Pour une vue d'ensemble des autorisations et des attributs d'exécution fournis par le package Logical Domains Manager, reportez-vous à la section "Profils contenus dans Logical Domains Manager" à la page 35. Tous les exemples de ce chapitre supposent que la configuration des droits utilise des fichiers locaux.

Utilisation des profils de droits et des rôles

Attention - Faites preuve de prudence lorsque vous utilisez les commandes `usermod` et `rolemod` pour ajouter des autorisations, des profils de droits ou des rôles.

- Pour le SE Oracle Solaris 11, ajoutez des valeurs en utilisant le signe plus (+) pour chaque autorisation que vous ajoutez.

  Par exemple, la commande `usermod -A +`*`auth username`* accorde l'autorisation *auth* à l'utilisateur *username* ; il en va de même pour la commande `rolemod`.
- Pour le système d'exploitation Oracle Solaris 10, la commande `usermod` ou `rolemod` remplace toutes les valeurs existantes.

  Pour ajouter des valeurs au lieu de les remplacer, spécifiez une liste séparée par des virgules comprenant les valeurs existantes et les nouvelles valeurs.

Gestion des profils de droits utilisateurs

Les procédures suivantes permettent de gérer les profils de droits utilisateurs sur le système à l'aide de fichiers locaux. Pour gérer les profils utilisateur dans un service de noms, reportez-vous au *System Administration Guide: Naming and Directory Services (DNS, NIS, and LDAP)*.

▼ Procédure d'affectation d'un profil de droits à un utilisateur

Les utilisateurs auxquels le profil `LDoms Management` a été affecté directement *doivent* appeler un shell de profil pour exécuter la commande `ldm` avec des attributs de sécurité. Pour plus

d'informations, reportez-vous au manuel *System Administration Guide: Security Services* dans *Securing Users and Processes in Oracle Solaris 11.3*.

1. **Connectez-vous en tant qu'administrateur.**

   Pour Oracle Solaris 11.3, reportez-vous au Chapitre 1, "About Using Rights to Control Users and Processes" du manuel *Securing Users and Processes in Oracle Solaris 11.3*.

2. **Affectez un profil d'administration à un compte utilisateur local.**

   Vous pouvez affecter le profil `LDoms Review` ou le profil `LDoms Management` à un compte utilisateur.

   ```
   # usermod -P "profile-name" username
   ```

   La commande suivante affecte le profil `LDoms Management` à l'utilisateur `sam`.

   ```
   # usermod -P "LDoms Management" sam
   ```

Assignation de rôles aux utilisateurs

La procédure suivante permet de créer un rôle et de l'affecter à un utilisateur à l'aide de fichiers locaux. Pour gérer les rôles dans un service de noms, reportez-vous au *System Administration Guide: Naming and Directory Services (DNS, NIS, and LDAP)*.

Cette procédure présente l'avantage que les rôles ne sont pris que par les utilisateurs à qui ils ont été affectés. Lorsqu'un utilisateur prend un rôle, il doit fournir le mot de passe qui a été affecté au rôle, le cas échéant. Ces deux niveaux de sécurité empêchent qu'un utilisateur prenne un rôle qui ne lui a pas été affecté, même s'il dispose du mot de passe correspondant.

▼ Procédure de création d'un rôle et d'affectation du rôle à un utilisateur

1. **Connectez-vous en tant qu'administrateur.**

   Pour Oracle Solaris 11.3, reportez-vous au Chapitre 1, "About Using Rights to Control Users and Processes" du manuel *Securing Users and Processes in Oracle Solaris 11.3*.

2. **Créez un rôle.**

   ```
   # roleadd -P "profile-name" role-name
   ```

3. **Assignez un mot de passe au rôle.**

   Vous êtes invité à spécifier un nouveau mot de passe et à le vérifier.

   ```
   # passwd role-name
   ```

4. **Assignez le rôle à un utilisateur.**

```
# useradd -R role-name username
```

5. **Assignez un mot de passe à un utilisateur.**

   Vous êtes invité à spécifier un nouveau mot de passe et à le vérifier.

   ```
   # passwd username
   ```

6. **Devenez l'utilisateur et fournissez le mot de passe, si nécessaire.**

   ```
   # su username
   ```

7. **Vérifiez que l'utilisateur peut accéder au rôle affecté.**

   ```
   $ id
   uid=nn(username) gid=nn(group-name)
   $ roles
   role-name
   ```

8. **Prenez le rôle et fournissez le mot de passe, si nécessaire.**

   ```
   $ su role-name
   ```

9. **Vérifiez que l'utilisateur a pris le rôle.**

   ```
   $ id
   uid=nn(role-name) gid=nn(group-name)
   ```

Exemple 1 Création d'un rôle et affectation du rôle à un utilisateur

Cet exemple illustre comment créer le rôle `ldm_read`, affecter le rôle à l'utilisateur `user_1`, devenir l'utilisateur `user_1` et prendre le rôle `ldm_read`.

```
# roleadd -P "LDoms Review" ldm_read
# passwd ldm_read
New Password:
Re-enter new Password:
passwd: password successfully changed for ldm_read
# useradd -R ldm_read user_1
# passwd user_1
New Password:
Re-enter new Password:
passwd: password successfully changed for user_1
# su user_1
Password:
$ id
uid=95555(user_1) gid=10(staff)
$ roles
ldm_read
$ su ldm_read
Password:
$ id
uid=99667(ldm_read) gid=14(sysadmin)
```

Profils contenus dans Logical Domains Manager

Le package Logical Domains Manager ajoute les profils de droits suivants à la base de données de description des profils de droits locale :

```
LDoms Power Mgmt Observability:::View LDoms Power Consumption:auths=solaris.ldoms.ldmpower
LDoms Review:::Review LDoms configuration:profiles=LDoms Power Mgmt Observability;
auths=solaris.ldoms.read
LDoms Management:::Manage LDoms domains:profiles=LDoms Power Mgmt Observability;
auths=solaris.ldoms.*
```

Le package Logical Domains Manager ajoute également l'attribut d'exécution suivant, qui est associé aux profils `LDoms Management` et `LDoms Power Mgmt Observability`, à la base de données des profils d'exécution locale :

```
LDoms Management:suser:cmd:::/usr/sbin/ldm:privs=file_dac_read,file_dac_search
LDoms Power Mgmt Observability:suser:cmd:::/usr/sbin/ldmpower:privs=file_dac_search
```

Le tableau suivant répertorie les sous-commandes `ldm` avec l'autorisation utilisateur correspondante requise pour utiliser les commandes.

TABLEAU 1 Sous-commandes `ldm` et autorisations utilisateur

Sous-commande ldm[†]	Autorisation utilisateur
`add-*`	`solaris.ldoms.write`
`bind-domain`	`solaris.ldoms.write`
`list`	`solaris.ldoms.read`
`list-*`	`solaris.ldoms.read`
`panic-domain`	`solaris.ldoms.write`
`remove-*`	`solaris.ldoms.write`
`set-*`	`solaris.ldoms.write`
`start-domain`	`solaris.ldoms.write`
`stop-domain`	`solaris.ldoms.write`
`unbind-domain`	`solaris.ldoms.write`

[†]Concerne toutes les ressources que vous pouvez ajouter, répertorier, supprimer ou configurer.

Utilisation de Verified Boot

Le Logical Domains Manager utilise la technologie Verified Boot d'SE Oracle Solaris pour vérifier la signature numérique des modules de noyau au moment de l'initialisation. La vérification de signature se fait en mode silencieux sauf si les stratégies Verified Boot sont activées. Selon la valeur de `boot-policy`, un domaine invité peut ne pas s'initialiser si le module

de noyau n'est pas signé avec des fichiers de certification de version Oracle Solaris ou s'il est corrompu.

Utilisez la commande `ldm add-domain` ou `ldm set-domain` pour spécifier les valeurs pour les propriétés `boot-policy` et `module-policy`. Reportez-vous à la page de manuel `ldm`(1M).

Pour utiliser cette fonctionnalité, votre système doit exécuter au minimum les versions suivantes du microprogramme système et du système d'exploitation :

- **Microprogramme système –** Version 9.5.0 pour les serveurs Oracle SPARC, toute version publiée pour les serveurs SPARC de série S7 et XCP 2280 pour les Serveurs Fujitsu M10
- **Système d'exploitation –** Oracle Solaris 11.2

Remarque - Par défaut, tout domaine créé à l'aide d'une version d'Oracle VM Server for SPARC antérieure à la 3.4 définit `boot-policy=warning`. Ce paramètre provoque l'affichage de messages d'avertissement lors de l'initialisation du domaine après une mise à jour d'Oracle VM Server for SPARC si le module de noyau n'est pas signé ou est corrompu.

♦ ♦ ♦ CHAPITRE 3

Configuration des services et du domaine de contrôle

Ce chapitre décrit la procédure de configuration des services par défaut et de votre domaine de contrôle.

Ce chapitre aborde les sujets suivants :

Messages de sortie

Si une ressource ne peut pas être configurée de manière dynamique sur le domaine de contrôle, la pratique recommandée consiste à commencer par lancer une reconfiguration retardée. La reconfiguration retardée reporte les activités de configuration à après la réinitialisation du domaine de contrôle.

Vous recevez le message suivant lorsque vous démarrez une reconfiguration retardée sur le domaine `primary` :

```
Initiating a delayed reconfiguration operation on the primary domain.
All configuration changes for other domains are disabled until the
primary domain reboots, at which time the new configuration for the
primary domain also takes effect.
```

Vous recevez l'avertissement suivant après chaque opération suivante sur le domaine `primary` jusqu'à la réinitialisation :

```
Notice: The primary domain is in the process of a delayed reconfiguration.
Any changes made to the primary domain will only take effect after it reboots.
```

Création des services par défaut

Les services de périphérique virtuel suivants doivent être créés pour utiliser le domaine de contrôle comme un domaine de service et pour créer des périphériques virtuels pour les autres domaines.

- `vcc` – Service de concentrateur de console virtuelle
- `vds` – Serveur de disque virtuel
- `vsw` – Service de commutateur virtuel

▼ Procédure de création des services par défaut

1. **Créez un service de concentrateur de console virtuelle (`vcc`) à utiliser par le démon du serveur de terminal de réseau virtuel (`vntsd`) et en tant que concentrateur pour toutes les consoles de domaine logique.**

   Par exemple, la commande suivante ajouterait un service de concentrateur de console virtuelle (`primary-vcc0`) avec une plage de port allant de 5000 à 5100 au domaine de contrôle (`primary`).

   ```
   primary# ldm add-vcc port-range=5000-5100 primary-vcc0 primary
   ```

2. **Créez un serveur de disque virtuel (`vds`) pour permettre l'importation de disques virtuels dans un domaine logique.**

   Par exemple, la commande suivante ajoute un serveur de disque virtuel (`primary-vds0`) au domaine de contrôle (`primary`).

   ```
   primary# ldm add-vds primary-vds0 primary
   ```

3. **Créez un service de commutateur virtuel (`vsw`) pour permettre la mise en réseau des périphériques du réseau virtuel (`vnet`) dans les domaines logiques.**

   Assignez un adaptateur réseau compatible GLDv3 au commutateur virtuel si chaque domaine logique doit communiquer différemment par l'intermédiaire du commutateur virtuel.

   Ajoutez un service de commutateur virtuel (`primary-vsw0`) sur un périphérique réseau que vous souhaitez utiliser pour la mise en réseau du domaine invité.

   ```
   primary# ldm add-vsw net-dev=network-device  vsw-service  primary
   ```

   Par exemple, la commande suivante ajoute un service de commutateur virtuel (`primary-vsw0`) sur l'adaptateur réseau `net0` dans le domaine de contrôle (`primary`) :

```
primary# ldm add-vsw net-dev=net0 primary-vsw0 primary
```

Vous pouvez utiliser la commande `ldm list-netdev -b` pour déterminer les périphériques du réseau de backend disponibles pour le commutateur virtuel. Voir la section "Commutateur virtuel" à la page 255.

Vous pouvez mettre à jour dynamiquement la valeur de la propriété `net-dev` à l'aide de la commande `ldm set-vsw`.

4. **Vérifiez que les services ont été créés à l'aide de la sous-commande `list-services`.**

   Votre sortie doit ressembler à ce qui suit :

```
primary# ldm list-services primary
VDS
    NAME             VOLUME         OPTIONS          DEVICE
    primary-vds0

VCC
    NAME             PORT-RANGE
    primary-vcc0     5000-5100

VSW
    NAME             MAC               NET-DEV   DEVICE     MODE
    primary-vsw0     02:04:4f:fb:9f:0d net0      switch@0   prog,promisc
```

Configuration initiale du domaine de contrôle

Initialement, toutes les ressources du système sont allouées au domaine de contrôle. Pour permettre la création d'autres domaines logiques, vous devez libérer certaines de ces ressources.

Configuration du domaine de contrôle

▼ Procédure de configuration du domaine de contrôle

Cette procédure contient des exemples de ressources pour configurer votre domaine de contrôle. Ces nombres sont des exemples uniquement et les valeurs utilisées peuvent ne pas être adaptées à votre domaine de contrôle.

Pour connaître les recommandations concernant les redimensionnements de domaine, reportez-vous à *Oracle VM Server for SPARC Best Practices* (`http://www.oracle.com/technetwork/server-storage/vm/ovmsparc-best-practices-2334546.pdf`).

1. **Assignez des CPU virtuelles au domaine de contrôle.**

Les domaines de service, y compris le domaine de contrôle, nécessitent que les ressources de la CPU et de la mémoire effectuent des opérations d'E/S de disque et de réseau virtuels pour les domaines invités. La quantité de ressources à attribuer pour la CPU et la mémoire dépend de la charge de travail du domaine invité.

Par exemple, la commande suivante attribue deux coeurs de CPU (16 threads de CPU virtuels) au domaine de contrôle `primary`. Le reste des threads de CPU virtuels sont disponibles pour les domaines invités.

```
primary# ldm set-core 2 primary
```

Vous pouvez modifier de manière dynamique la répartition réelle de CPU en fonction des exigences applicatives. Utilisez la commande `ldm list` pour déterminer l'utilisation de la CPU par le domaine de contrôle. Si le domaine de contrôle utilise une quantité importante de CPU, utilisez les commandes `ldm add-core` et `ldm set-core` pour ajouter des ressources de CPU à un domaine de service.

2. **Déterminez si vous avez besoin de périphériques cryptographiques dans le domaine de contrôle.**

   Notez que seules les plates-formes UltraSPARC T2, UltraSPARC T2 Plus et SPARC T3 sont équipées de périphériques cryptographiques (MAU). Les plates-formes les plus récentes, telles que les systèmes SPARC T4 et les Serveurs Fujitsu M10, fournissent déjà une accélération cryptographique.

   Si vous utilisez un processeur plus ancien, affectez une unité cryptographique à chaque coeur complet de CPU dans le domaine de contrôle.

   L'exemple suivant affecte deux ressources cryptographiques au domaine de contrôle `primary`.

   ```
   primary# ldm set-crypto 2 primary
   ```

3. **Démarrez une reconfiguration retardée sur le domaine de contrôle.**

   ```
   primary# ldm start-reconf primary
   ```

4. **Assignez de la mémoire au domaine de contrôle.**

   Par exemple, la commande suivante affecte 16 Go de mémoire au domaine de contrôle `primary`. Cette configuration laisse le reste de la mémoire à la disposition des domaines invités.

   ```
   primary# ldm set-memory 16G primary
   ```

5. **Enregistrez la configuration du domaine dans le processeur de service (SP).**

   Par exemple, la commande suivante ajoute une configuration appelée `initial`.

   ```
   primary# ldm add-config initial
   ```

6. **Vérifiez que la configuration est prête à être utilisée à la prochaine réinitialisation.**

```
primary# ldm list-config
factory-default
initial [current]
```

La commande `ldm list-config` indique que la configuration `initiale` définie sera utilisée après le cycle d'alimentation suivant.

7. **Réinitialisez le domaine de contrôle pour appliquer les modifications.**

Réduction des ressources de la CPU et de la mémoire par rapport à la configuration `factory-default` initiale du domaine de contrôle

Vous pouvez utiliser la reconfiguration dynamique de CPU pour diminuer le nombre de coeurs du domaine de contrôle à partir d'une configuration `factory-default` initiale. Vous devez toutefois utiliser une reconfiguration retardée au lieu d'une reconfiguration dynamique de mémoire pour réduire la mémoire du domaine de contrôle.

En configuration `factory-default`, le domaine de contrôle possède toute la mémoire du système hôte. La fonction de reconfiguration dynamique de mémoire n'est pas adaptée à cet objectif, car il n'est pas certain qu'un domaine actif ajoute, ou plus généralement supprime, toute la mémoire demandée. Au lieu de cela, le SE s'exécutant sur ce domaine fait de son mieux pour remplir la demande. Par ailleurs, la suppression de mémoire peut être une opération de longue durée. Ces problèmes sont amplifiés lorsque des opérations importantes sur la mémoire sont impliquées, comme c'est le cas pour la réduction initiale de la mémoire du domaine de contrôle.

Remarque - Lorsque le SE Oracle Solaris est installé sur un système de fichiers ZFS, il dimensionne et crée automatiquement des zones de swap et de vidage en tant que volumes ZFS dans le pool root ZFS, en fonction de la quantité de mémoire physique présente. Si vous modifiez l'attribution de mémoire du domaine, vous risquez de modifier la taille recommandée de ces volumes. Les allocations pourront être plus grandes que nécessaire après réduction de la mémoire du domaine de contrôle. Pour connaître les recommandations concernant l'espace de swap, reportez-vous à "Planning for Swap Space" du manuel *Managing File Systems in Oracle Solaris 11.3*. Avant de libérer de l'espace, vous pouvez éventuellement modifier l'espace de swap et de vidage. Reportez-vous à la section "Managing ZFS Swap and Dump Devices" du manuel *Managing ZFS File Systems in Oracle Solaris 11.3*.

▼ Procédure de diminution des ressources de la CPU et de la mémoire à partir de la configuration `factory-default` initiale du domaine de contrôle

Cette procédure explique comment diminuer les ressources de la CPU et de la mémoire à partir de la configuration `factory-default` initiale du domaine de contrôle. Utilisez d'abord la reconfiguration dynamique de CPU pour diminuer le nombre de coeurs, puis lancez une reconfiguration retardée avant de réduire la quantité de mémoire.

Les exemples de valeurs correspondent aux tailles de CPU et de mémoire d'un petit domaine de contrôle qui dispose d'assez de ressources pour exécuter le démon `ldmd` et effectuer des migrations. Cependant, si vous souhaitez utiliser le domaine de contrôle pour des fonctions supplémentaires, vous pouvez affecter un plus grand nombre de coeurs et une plus grande quantité de mémoire au domaine de contrôle selon vos besoins.

1. **Initialisez la configuration `factory-default`.**

2. **Configurez le domaine de contrôle.**

   Voir la section "Procédure de configuration du domaine de contrôle" à la page 39.

Réinitialisation pour utiliser les domaines

Vous devez réinitialiser le domaine de contrôle pour que les modifications de configuration soient appliquées et que les ressources soient libérées pour une utilisation par les autres domaines logiques.

▼ Procédure de réinitialisation

- **Arrêtez et réinitialisez le domaine de contrôle.**

```
primary# shutdown -y -g0 -i6
```

Remarque - Une réinitialisation ou un cycle d'alimentation instancie la nouvelle configuration. Seul un cycle d'alimentation initialise réellement la configuration enregistrée sur le processeur de service (SP), ce qui est ensuite répercuté dans la sortie `list-config`.

Activation de la mise en réseau entre le domaine de service Oracle Solaris 10 et les autres domaines

Par défaut, la mise en réseau entre le domaine de service Oracle Solaris 10 et les autres domaines du système est désactivée. Comme la mise en réseau n'est pas activée par défaut dans le SE Oracle Solaris 10, vous devez l'activer en configurant le commutateur virtuel en tant que périphérique réseau. Le commutateur virtuel peut remplacer le périphérique sous-jacent (`nxge0` dans cet exemple) en tant qu'interface principale ou être configuré en tant qu'interface réseau supplémentaire dans le domaine.

Les domaines invités peuvent communiquer automatiquement avec le domaine de service Oracle Solaris 10 tant que le périphérique backend réseau correspondant est configuré dans le même LAN virtuel ou réseau virtuel.

▼ Procédure de configuration du commutateur virtuel en tant qu'interface primary

---

Remarque - Procédez comme suit à partir de la console du domaine de service Oracle Solaris 10, car cette procédure peut interrompre momentanément la connectivité au domaine.

Si nécessaire, vous pouvez configurer le commutateur virtuel ainsi que le périphérique réseau physique. Dans ce cas, créez le commutateur virtuel comme à l'étape 2 et ne supprimez pas le périphérique physique (ignorez l'étape 3). Vous devez ensuite configurer le commutateur virtuel avec une adresse IP statique ou une adresse IP dynamique. Vous pouvez obtenir une adresse IP dynamique à partir d'un serveur DHCP. Pour plus d'informations et pour obtenir un exemple de ce cas, reportez-vous à la section "Configuration d'un commutateur virtuel et du domaine de service pour NAT et le routage" à la page 274

1. **Imprimez les informations d'adressage pour toutes les interfaces.**

```
# ifconfig -a
```

2. **Configurez l'interface réseau du commutateur virtuel.**

```
# ifconfig vsw0 plumb
```

3. **Supprimez l'interface physique du périphérique affecté au commutateur virtuel (`net-dev`).**

```
# ifconfig nxge0 down unplumb
```

4. **Pour migrer les propriétés du périphérique réseau physique (`nxge0`) vers le périphérique commutateur virtuel (`vsw0`), effectuez l'une des opérations suivantes :**

   - **Si la mise en réseau est configurée avec une adresse IP statique, réutilisez l'adresse IP et le masque réseau de `nxge0` pour le commutateur virtuel.**

     ```
     # ifconfig vsw0 IP-of-nxge0 netmask netmask-of-nxge0 broadcast + up
     ```

   - **Si la mise en réseau est configurée à l'aide de DHCP, activez DHCP pour le commutateur virtuel.**

     ```
     # ifconfig vsw0 dhcp start
     ```

5. **Apportez les modifications requises au fichier de configuration pour rendre ce changement permanent.**

   ```
   # mv /etc/hostname.nxge0 /etc/hostname.vsw0
   # mv /etc/dhcp.nxge0 /etc/dhcp.vsw0
   ```

Activation du démon du serveur de terminal du réseau virtuel

Vous devez activer le démon du serveur de terminal du réseau virtuel (`vntsd`) pour fournir l'accès à la console virtuelle de chaque domaine logique. Reportez-vous à la page de manuel `vntsd`(1M) pour plus d'informations sur l'utilisation de ce démon.

▼ Procédure d'activation du démon du serveur de terminal du réseau virtuel

---

Remarque - Assurez-vous d'avoir créé le service par défaut `vconscon` (`vcc`) sur le domaine de contrôle avant d'activer `vntsd`. Voir la section "Création des services par défaut" à la page 38 pour plus d'informations.

---

1. **Activez le démon du serveur de terminal réseau virtuel, `vntsd`.**

   ```
   primary# svcadm enable vntsd
   ```

2. **Vérifiez que le démon `vntsd` est activé.**

   ```
   primary# svcs vntsd
   ```

```
STATE          STIME    FMRI
online         Oct_08   svc:/ldoms/vntsd:default
```

Vérification de l'activation de l'interconnexion ILOM

L'interconnexion ILOM est requise pour la communication entre le démon `ldmd` et le processeur de service (SP) sur les serveurs SPARC des séries T7, M7 et S7 et ne doit pas être désactivée. Pour plus d'informations, reportez-vous à la page de manuel `ilomconfig`(1M).

Remarque - Evitez de désactiver l'interconnexion ILOM sur les autres serveurs SPARC de série T, M5 et M6. Cependant, si vous le faites, le démon `ldmd` peut toujours communiquer avec le SP.

Si une tentative d'exécution de la commande `ldm` pour gérer les configurations de domaine sur des serveurs SPARC de série T7, M7 ou S7 échoue en raison d'une erreur de communication avec le processeur de service, vérifiez l'état de l'interconnexion ILOM et réactivez le service `ilomconfig-interconnect` si nécessaire. Reportez-vous à "Procédure de vérification de la configuration d'interconnexion ILOM" à la page 45 et "Procédure de réactivation du service d'interconnexion ILOM" à la page 46.

▼ Procédure de vérification de la configuration d'interconnexion ILOM

1. **Vérifiez que le service `ilomconfig-interconnect` est activé.**

```
primary# svcs ilomconfig-interconnect
STATE          STIME    FMRI
online          9:53:28 svc:/network/ilomconfig-interconnect:default
```

2. **Vérifiez que l'interconnexion ILOM est correctement configurée.**

   Dans une configuration appropriée, la valeur d'état est `enabled` et la valeur d'adresse IP de l'interconnexion hôte est une adresse IP et n'est pas `none`.

```
primary# ilomconfig list interconnect
Interconnect
============
State: enabled
Type: USB Ethernet
SP Interconnect IP Address: 169.254.182.76
Host Interconnect IP Address: 169.254.182.77
Interconnect Netmask: 255.255.255.0
SP Interconnect MAC Address: 02:21:28:57:47:16
Host Interconnect MAC Address: 02:21:28:57:47:17
```

3. **Vérifiez que le démon `ldmd` peut communiquer avec le SP.**

```
primary# ldm list-spconfig
```

▼ Procédure de réactivation du service d'interconnexion ILOM

Le service `ilomconfig-interconnect` est activé par défaut. Utilisez cette procédure si vous avez besoin de réactiver ce service manuellement.

1. **Activez le service d'interconnexion ILOM.**

```
primary# svcadm enable ilomconfig-interconnect
```

2. **Vérifiez que le service `ilomconfig-interconnect` est activé.**

```
primary# svcs ilomconfig-interconnect
STATE          STIME    FMRI
online          9:53:28 svc:/network/ilomconfig-interconnect:default
```

3. **Vérifiez que l'interconnexion ILOM est correctement configurée.**

   Dans une configuration appropriée, la valeur d'état est `enabled` et la valeur d'adresse IP de l'interconnexion hôte est une adresse IP et n'est pas `none`.

```
primary# ilomconfig list interconnect
Interconnect
============
State: enabled
Type: USB Ethernet
SP Interconnect IP Address: 169.254.182.76
Host Interconnect IP Address: 169.254.182.77
Interconnect Netmask: 255.255.255.0
SP Interconnect MAC Address: 02:21:28:57:47:16
Host Interconnect MAC Address: 02:21:28:57:47:17
```

4. **Vérifiez que le démon `ldmd` peut communiquer avec le SP.**

```
primary# ldm list-spconfig
```

♦ ♦ ♦ CHAPITRE 4

Configuration des domaines invités

Ce chapitre décrit les procédures nécessaires à la configuration des domaines invités.

Ce chapitre aborde les sujets suivants :

Création et démarrage d'un domaine invité

Le domaine invité doit exécuter le système d'exploitation qui est compatible avec la plate-forme `sun4v` et les périphériques virtuels présentés par l'hyperviseur. Actuellement, cette contrainte signifie que vous devez exécuter au moins le SE Oracle Solaris 10 11/06. L'exécution du SE Oracle Solaris 10 1/13 vous fournit toutes les fonctions d'Oracle VM Server for SPARC 3.4. Voir *Guide d'installation d'Oracle VM Server for SPARC 3.4* pour obtenir tous les patchs spécifiques dont vous pouvez avoir besoin. Une fois les services par défaut créés et les ressources réallouées à partir du domaine de contrôle, vous pouvez créer et démarrer un domaine invité.

Remarque - Un domaine invité auquel plus de 1 024 CPU ont été affectées ou qui comporte un ID CPU égal ou supérieur à 1 024 ne peut pas exécuter le SE Oracle Solaris 10. Vous ne pouvez pas utiliser la reconfiguration dynamique de CPU pour définir un nombre de CPU ou d'ID CPU inférieur à 1 024 et exécuter le SE Oracle Solaris 10.

▼ Procédure de création et de démarrage d'un domaine invité

1. **Créez un domaine logique.**

   La commande suivante crée un domaine invité nommé `ldg1`.

```
primary# ldm add-domain ldg1
```

2. **Ajoutez des CPU au domaine invité.**

   Effectuez l'une des opérations suivantes :

   - **Ajoutez des CPU virtuelles.**

     La commande suivante ajoute huit CPU virtuelles au domaine invité `ldg1`.

     ```
     primary# ldm add-vcpu 8 ldg1
     ```

   - **Ajouter des coeurs complets.**

     La commande suivante ajoute deux coeurs complets au domaine invité `ldg1`.

     ```
     primary# ldm add-core 2 ldg1
     ```

3. **Ajoutez de la mémoire au domaine invité.**

   La commande suivante ajoute 2 Go de mémoire au domaine invité `ldg1`.

   ```
   primary# ldm add-memory 2G ldg1
   ```

4. **Ajoutez un périphérique réseau virtuel au domaine invité.**

   La commande suivante ajoute un périphérique réseau virtuel avec ces paramètres au domaine invité `ldg1`.

   ```
   primary# ldm add-vnet vnet1 primary-vsw0 ldg1
   ```

   où :

   - `vnet1` est un nom d'interface unique vers le domaine logique, assigné à cette instance de périphérique réseau virtuel pour référence sur les sous-commandes ultérieures `set-vnet` ou `remove-vnet`.
   - `primary-vsw0` est le nom du service réseau existant (commutateur virtuel) auquel se connecter.

   Remarque - Les étapes 5 et 6 sont des instructions simplifiées pour ajouter un périphérique de serveur de disque virtuel (`vdsdev`) au domaine primary et un disque virtuel (`vdisk`) au domaine invité. Pour plus d'informations sur l'utilisation des volumes et des systèmes de fichiers ZFS en tant que disques virtuels, reportez-vous aux sections "Procédure d'exportation d'un volume ZFS en tant que disque à tranche unique" à la page 198 et "Utilisation de ZFS avec les disques virtuels" à la page 211.

5. **Indiquez le périphérique que le serveur de disque virtuel doit exporter en tant que disque virtuel sur le domaine invité.**

Vous pouvez exporter un disque physique, une tranche de disque ou un fichier en tant que périphérique en mode bloc. Les exemples suivants présentent un disque physique et un fichier.

- **Exemple d'un disque physique.** Dans cet exemple, un disque physique est ajouté à l'aide des informations suivantes :

  ```
  primary# ldm add-vdsdev /dev/dsk/c2t1d0s2 vol1@primary-vds0
  ```

  où :

  - `/dev/dsk/c2t1d0s2` est le nom du chemin d'accès du périphérique physique réel. Lors de l'ajout d'un périphérique, le nom du chemin d'accès doit être associé au nom du périphérique.
  - `vol1` est un nom unique que vous définissez pour le périphérique ajouté au serveur de disque virtuel. Le nom de volume doit être unique pour cette instance de serveur de disque virtuel, car ce nom est exporté par le serveur de disque virtuel sur les clients pour l'ajout. Lors de l'ajout d'un périphérique, le nom de volume doit être associé au nom du chemin d'accès du périphérique réel.
  - `primary-vds0` est le nom du serveur de disque virtuel auquel ajouter ce périphérique.
- **Exemple d'un fichier.** Dans cet exemple, un fichier est exporté en tant que périphérique en mode bloc.

  ```
  primary# ldm add-vdsdev backend vol1@primary-vds0
  ```

  où :

  - *backend* est le nom du chemin d'accès du fichier réel exporté en tant que périphérique en mode bloc. Lors de l'ajout d'un périphérique, le moteur de traitement doit être associé au nom du périphérique.
  - `vol1` est un nom unique que vous définissez pour le périphérique ajouté au serveur de disque virtuel. Le nom de volume doit être unique pour cette instance de serveur de disque virtuel, car ce nom est exporté par le serveur de disque virtuel sur les clients pour l'ajout. Lors de l'ajout d'un périphérique, le nom de volume doit être associé au nom du chemin d'accès du périphérique réel.
  - `primary-vds0` est le nom du serveur de disque virtuel auquel ajouter ce périphérique.

6. Ajoutez un disque virtuel au domaine invité.

L'exemple suivant ajout un disque virtuel au domaine invité `ldg1`.

```
primary# ldm add-vdisk vdisk1 vol1@primary-vds0 ldg1
```

où :

- `vdisk1` est le nom du disque virtuel.
- `vol1` est le nom du volume existant auquel se connecter.
- `primary-vds0` est le nom du serveur de disque virtuel existant auquel se connecter.

Remarque - Les disques virtuels sont des blocs génériques associés à différents types de périphériques physiques, volumes ou fichiers. Un disque virtuel n'est pas synonyme de disque SCSI et, par conséquent, exclut l'ID cible dans l'étiquette de disque. Les disques virtuels dans un domaine logique ont le format suivant : `cNdNsN`, où `cN` est le contrôleur virtuel, `dN` est le numéro du disque virtuel et `sN` est la tranche.

7. **Définissez les variables `auto-boot?` et `boot-device` pour le domaine invité.**

Remarque - Lorsque vous définissez la valeur de la propriété `boot-device`, utilisez uniquement des minuscules même si le nom du disque virtuel comprend des majuscules.

L'exemple de commande suivant définit `auto-boot?` sur `true` pour le domaine invité `ldg1`.

```
primary# ldm set-var auto-boot\?=true ldg1
```

L'exemple de commande suivant définit `boot-device` sur `vdisk1` pour le domaine invité `ldg1`.

```
primary# ldm set-var boot-device=vdisk1 ldg1
```

8. **Liez les ressources au domaine invité `ldg1`, puis répertoriez le domaine pour vérifier qu'il est associé.**

```
primary# ldm bind-domain ldg1
primary# ldm list-domain ldg1
NAME          STATE    FLAGS  CONS   VCPU MEMORY   UTIL  UPTIME
ldg1          bound    -----  5000   8    2G
```

9. **Pour trouver le port de la console du domaine invité, vous pouvez consulter la sortie de la sous-commande précédente `list-domain`.**

Vous pouvez voir sous le titre `CONS` que le domaine logique invité 1 (`ldg1`) a sa sortie de console liée au port `5000`.

10. **Connectez-vous à la console d'un domaine invité à partir d'un autre terminal en vous connectant au domaine de contrôle et en vous connectant directement au port de la console sur l'hôte local.**

```
$ ssh hostname.domain-name
$ telnet localhost 5000
```

11. **Démarrez le domaine invité `ldg1`.**

```
primary# ldm start-domain ldg1
```

Installation du SE Oracle Solaris sur un domaine invité

Cette section fournit des instructions sur différentes méthodes d'installation du SE Oracle Solaris sur un domaine invité.

Attention - Ne vous déconnectez *pas* de la console virtuelle au cours de l'installation du SE Oracle Solaris.

Pour les domaines Oracle Solaris 11, utilisez le profil de configuration réseau (NCP) `DefaultFixed`. Vous pouvez activer ce profil pendant ou après l'installation.

Pendant l'installation d'Oracle Solaris 11, sélectionnez la configuration de gestion des réseaux Manuelle. Après l'installation d'Oracle Solaris 11, assurez-vous que le NCP `DefaultFixed` est activé à l'aide de la commande `netadm list`. Reportez-vous aux chapitres 2, 3, 5 et 6 du manuel *Configuring and Managing Network Components in Oracle Solaris 11.3*.

Taille de mémoire requise

Le logiciel Oracle VM Server for SPARC n'impose aucune limitation de taille pour la mémoire lors de la création d'un domaine. La taille de la mémoire requise est une caractéristique du système d'exploitation hôte. Certaines fonctions d'Oracle VM Server for SPARC risquent de ne pas fonctionner si la quantité de mémoire présente est inférieure à la taille recommandée. Pour connaître la taille recommandée et minimale de la mémoire pour le SE Oracle Solaris 10, reportez-vous à la section "System Requirements and Recommendations" du manuel *Oracle Solaris 10 1/13 Installation Guide: Planning for Installation and Upgrade*. Pour connaître la taille recommandée et minimale de la mémoire pour le SE Oracle Solaris 11, reportez-vous aux sections *Oracle Solaris 11 Release Notes*, *Oracle Solaris 11.1 Release Notes*, *Oracle Solaris 11.2 Release Notes*, et *Oracle Solaris 11.3 Release Notes*.

La PROM OpenBoot a une contrainte de taille minimale pour un domaine. Actuellement, cette limite est de 12 méga-octets. Si vous avez un domaine dont la taille est inférieure, Logical Domains Manager étend automatiquement celui-ci à 12 Mo. La taille minimale d'un Serveur Fujitsu M10 est limitée à 256 Mo. Reportez-vous aux notes de version de votre microprogramme système pour connaître la taille de la mémoire requise

La fonction de reconfiguration dynamique (DR) de la mémoire applique un alignement de 256 Mo sur l'adresse et la taille de la mémoire impliquées dans une opération donnée. Voir la section "Alignement de la mémoire" à la page 367.

▼ Procédure d'installation du SE Oracle Solaris à l'aide d'un DVD sur un domaine invité

1. **Insérez le DVD du SE Oracle Solaris 10 ou Oracle Solaris 11 dans l'unité de DVD.**

2. **Arrêtez le démon de gestion du volume, `vold`(1M), sur le domaine `primary`.**

```
primary# svcadm disable volfs
```

3. **Arrêtez et dissociez le domaine invité (`ldg1`).**

```
primary# ldm stop ldg1
primary# ldm unbind ldg1
```

4. **Ajoutez le DVD avec le média DVD-ROM en tant que volume secondaire et disque virtuel.**

   L'exemple suivant utilise `c0t0d0s2` comme unité de DVD dans laquelle le média d'Oracle Solaris réside, `dvd_vol@primary-vds0` comme volume secondaire et `vdisk_cd_media` comme disque virtuel.

```
primary# ldm add-vdsdev options=ro /dev/dsk/c0t0d0s2 dvd_vol@primary-vds0
primary# ldm add-vdisk vdisk_cd_media dvd_vol@primary-vds0 ldg1
```

5. **Vérifiez que le DVD est ajouté en tant que volume secondaire et disque virtuel.**

```
primary# ldm list-bindings
NAME             STATE    FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active   -n-cv   SP      8     8G       0.2%  22h 45m
...
VDS
   NAME             VOLUME         OPTIONS          DEVICE
   primary-vds0     vol1                            /dev/dsk/c2t1d0s2
   dvd_vol                                          /dev/dsk/c0t0d0s2
....
------------------------------------------------------------------------------
NAME             STATE    FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
ldg1             inactive -----           60    6G
...
DISK
   NAME             VOLUME                      TOUT DEVICE  SERVER
   vdisk1           vol1@primary-vds0
   vdisk_cd_media   dvd_vol@primary-vds0
....
```

6. **Associez et démarrez le domaine invité (`ldg1`).**

```
primary# ldm bind ldg1
```

```
primary# ldm start ldg1
LDom ldg1 started
primary# telnet localhost 5000
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.

Connecting to console "ldg1" in group "ldg1" ....
Press ~? for control options ..
```

7. **Affichez les alias du périphérique dans le client PROM OpenBoot.**

   Dans cet exemple, recherchez les alias du périphérique pour `vdisk_cd_media`, qui est le DVD d'Oracle Solaris et `vdisk1`, qui est un disque virtuel sur lequel vous pouvez installer le SE Oracle Solaris.

```
ok devalias
vdisk_cd_media  /virtual-devices@100/channel-devices@200/disk@1
vdisk1          /virtual-devices@100/channel-devices@200/disk@0
vnet1           /virtual-devices@100/channel-devices@200/network@0
virtual-console /virtual-devices/console@1
name            aliases
```

8. **Sur la console du domaine invité, initialisez à partir de `vdisk-cd_media` (`disk@1`) sur la tranche `f`.**

```
ok boot vdisk_cd_media:f
Boot device: /virtual-devices@100/channel-devices@200/disk@1:f  File and args: -s
SunOS Release 5.10 Version Generic_139555-08 64-bit
Copyright (c), 1983-2010, Oracle and/or its affiliates.  All rights reserved.
```

9. **Poursuivez l'installation du SE Oracle Solaris.**

▼ Procédure d'installation du SE Oracle Solaris sur un domaine invité à l'aide d'un fichier ISO Oracle Solaris

1. **Arrêtez et dissociez le domaine invité (`ldg1`).**

```
primary# ldm stop ldg1
primary# ldm unbind ldg1
```

2. **Ajoutez le fichier ISO Oracle Solaris en tant que volume secondaire et disque virtuel.**

   L'exemple suivant utilise `solarisdvd.iso` en tant que fichier ISO Oracle Solaris, `iso_vol@primary-vds0` en tant que volume secondaire et `vdisk_iso` en tant que disque virtuel :

```
primary# ldm add-vdsdev /export/solarisdvd.iso  iso_vol@primary-vds0
```

```
primary# ldm add-vdisk vdisk_iso iso_vol@primary-vds0 ldg1
```

3. **Vérifiez que le fichier ISO Oracle Solaris est ajouté en tant que volume secondaire et disque virtuel.**

```
primary# ldm list-bindings
NAME             STATE    FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active   -n-cv   SP      8     8G       0.2%  22h 45m
...
VDS
   NAME             VOLUME         OPTIONS          DEVICE
   primary-vds0     vol1                            /dev/dsk/c2t1d0s2
   iso_vol                                          /export/solarisdvd.iso
....
------------------------------------------------------------------------------
NAME             STATE    FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
ldg1             inactive -----           60    6G
...
DISK
   NAME       VOLUME               TOUT ID DEVICE  SERVER  MPGROUP
   vdisk1     vol1@primary-vds0
   vdisk_iso  iso_vol@primary-vds0
....
```

4. **Associez et démarrez le domaine invité (`ldg1`).**

```
primary# ldm bind ldg1
primary# ldm start ldg1
LDom ldg1 started
primary# telnet localhost 5000
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.

Connecting to console "ldg1" in group "ldg1" ....
Press ~? for control options ..
```

5. **Affichez les alias du périphérique dans le client PROM OpenBoot.**

   Dans cet exemple, recherchez les alias de périphérique pour `vdisk_iso`, qui est l'image ISO Oracle Solaris et `vdisk_install` qui est l'espace disque.

```
ok devalias
vdisk_iso       /virtual-devices@100/channel-devices@200/disk@1
vdisk1          /virtual-devices@100/channel-devices@200/disk@0
vnet1           /virtual-devices@100/channel-devices@200/network@0
virtual-console /virtual-devices/console@1
name            aliases
```

6. **Sur la console du domaine invité, initialisez à partir de `vdisk_iso` (`disk@1`) sur la tranche `f`.**

```
ok boot vdisk_iso:f
Boot device: /virtual-devices@100/channel-devices@200/disk@1:f  File and args: -s
SunOS Release 5.10 Version Generic_139555-08 64-bit
Copyright (c) 1983-2010, Oracle and/or its affiliates.  All rights reserved.
```

7. **Poursuivez l'installation du SE Oracle Solaris.**

▼ Procédure d'utilisation de la fonction JumpStart d'Oracle Solaris sur un domaine invité Oracle Solaris 10

Remarque - La fonction JumpStart d'Oracle Solaris est uniquement disponible pour le système d'exploitation Oracle Solaris 10. Voir *Oracle Solaris 10 1/13 Installation Guide: JumpStart Installations*.

Pour effectuer une installation automatisée du système d'exploitation Oracle Solaris 11, vous pouvez utiliser le programme d'installation automatisée (AI). Voir *Transitioning From Oracle Solaris 10 to Oracle Solaris 11.3*.

- **Modifiez votre profil JumpStart afin de refléter le format de nom de périphérique de disque différent pour le domaine invité.**

  Les noms de périphérique de disque virtuel dans un domaine logique diffèrent des noms de périphériques de disque physique. Les noms de périphérique de disque virtuel ne contiennent pas d'ID cible (`t`*N*). Au lieu du format usuel`c`*N*`t`*N*`d`*N*`s`*N*, les noms de périphérique de disque virtuel utilise le format `c`*N*`d`*N*`s`*N*. `c`*N* est le contrôleur virtuel, `d`*N* est le numéro de disque virtuel et `s`*N* est le numéro de tranche.

  Remarque - Un disque virtuel peut apparaître comme un disque complet ou un disque à tranche unique. Le SE Oracle Solaris peut être installé sur un disque complet l'aide d'un profil JumpStart normal qui définit plusieurs partitions. Un disque à tranche unique n'a qu'une seule partition, `s0`, qui utilise tout le disque. Pour installer le SE Oracle Solaris sur un seul disque, vous devez utiliser un profil ayant une seule partition (`/`) qui utilise tout le disque. Vous ne pouvez pas définir d'autres partitions, notamment un swap. Pour plus d'informations sur les disques complets et les disques à tranche unique, reportez-vous à la section "Apparence d'un disque virtuel" à la page 189.

  - **Profil JumpStart d'installation d'un système de fichiers root UFS.**

    Voir *Oracle Solaris 10 1/13 Installation Guide: JumpStart Installations*.

    Profil UFS normal

    ```
    filesys c1t1d0s0 free /
    filesys c1t1d0s1 2048 swap
    filesys c1t1d0s5 120 /spare1
    filesys c1t1d0s6 120 /spare2
    ```

    Profil UFS réel pour installer un domaine sur un disque complet

    ```
    filesys c0d0s0 free /
    filesys c0d0s1 2048 swap
    ```

```
filesys c0d0s5 120 /spare1
filesys c0d0s6 120 /spare2
```

Profil UFS réel pour installer un domaine sur un disque à tranche unique

```
filesys c0d0s0 free /
```

- **Profil JumpStart d'installation d'un système de fichiers root ZFS.**

  Voir Chapitre 9, "Installing a ZFS Root Pool With JumpStart" du manuel *Oracle Solaris 10 1/13 Installation Guide: JumpStart Installations*.

  Profil ZFS normal

  ```
  pool rpool auto 2G 2G c1t1d0s0
  ```

  Profil ZFS réel pour l'installation d'un domaine

  ```
  pool rpool auto 2G 2G c0d0s0
  ```

♦ ♦ ♦ CHAPITRE 5

Utilisation des consoles de domaine

Ce chapitre décrit les fonctions de console de domaine que vous pouvez activer sur votre système Oracle VM Server for SPARC.

Ce chapitre aborde les sujets suivants :

Remarque - Les exemples de ce manuel sont présentés comme s'ils étaient réalisés par un superutilisateur. Cependant, les profils d'utilisateur permettent aux utilisateurs d'obtenir des autorisations plus détaillées pour effectuer les tâches de gestion.

Contrôle de l'accès à une console de domaine à l'aide de droits utilisateur

Par défaut, tous les utilisateurs peuvent accéder à toutes les consoles de domaines. Pour contrôler l'accès à une console, configurez le démon `vntsd` de manière à ce qu'il procède à une vérification des autorisations. Le démon `vntsd` fournit la propriété SMF (Service Management Facility, utilitaire de gestion des services) appelée `vntsd/authorization`. Cette propriété peut être configurée de manière à activer la vérification des autorisations des utilisateurs et des rôles pour une console de domaine ou un groupe de consoles. Pour activer le contrôle d'autorisation, utilisez la commande `svccfg` pour définir la valeur de cette propriété sur `true`. Lorsque cette option est activée, `vntsd` écoute et accepte les connexions uniquement sur `localhost`. Si la propriété `listen_addr` indique une autre adresse IP lorsque `vntsd/authorization` est activé, `vntsd` ignore l'autre adresse IP et continue d'écouter uniquement sur `localhost`.

Attention - Ne configurez *pas* le service `vntsd` pour utiliser un hôte autre que `localhost`.

Si vous spécifiez un hôte différent de `localhost`, vous n'êtes plus empêché de vous connecter aux consoles de domaine invitées à partir du domaine de contrôle. Si vous vous connectez à distance à un domaine invité à l'aide de la commande `telnet`, les informations d'identification de connexion sont transmises en texte clair sur le réseau.

Par défaut, une autorisation d'accès à toutes les consoles invitées est présente dans la base de données de description de l'autorisation locale.

```
solaris.vntsd.consoles:::Access All LDoms Guest Consoles::
```

Servez-vous de la commande `usermod` pour affecter les autorisations requises à des utilisateurs ou des rôles dans les fichiers locaux. Avec cette commande, seul l'utilisateur ou le rôle qui possède les autorisations requises peut accéder à une console de domaine ou un groupe de consoles donnés. Pour affecter des autorisations à des utilisateurs ou des rôles dans un service de noms, reportez-vous au manuel *System Administration Guide: Naming and Directory Services (DNS, NIS, and LDAP)*.

Vous pouvez contrôler l'accès à toutes les consoles de domaines ou à une seule console de domaine.

- Pour contrôler l'accès à toutes les consoles de domaines, reportez-vous aux sections "Procédure de contrôle de l'accès à toutes les consoles de domaines par le biais de rôles" à la page 58 et "Procédure de contrôle de l'accès à toutes les consoles de domaines par le biais de profils de droits" à la page 60
- Pour contrôler l'accès à une seule console de domaine, reportez-vous aux sections "Procédure de contrôle de l'accès à une console unique par le biais de rôles" à la page 61 et "Procédure de contrôle de l'accès à une console unique par le biais de profils de droits" à la page 62.

▼ Procédure de contrôle de l'accès à toutes les consoles de domaines par le biais de rôles

1. **Restreignez l'accès à une console de domaine en activant la vérification des autorisations de la console.**

```
primary# svccfg -s vntsd setprop vntsd/authorization = true
primary# svcadm refresh vntsd
primary# svcadm restart vntsd
```

2. **Créez un rôle possédant l'autorisation `solaris.vntsd.consoles`, qui permet d'accéder à toutes les consoles de domaines.**

```
primary# roleadd -A solaris.vntsd.consoles role-name
```

```
primary# passwd role-name
```

3. **Assignez le nouveau rôle à un utilisateur.**

```
primary# usermod -R role-name username
```

Exemple 2 Contrôle de l'accès à toutes les consoles de domaines par le biais de rôles

Activez tout d'abord la vérification des autorisations de la console pour restreindre l'accès à une console de domaine.

```
primary# svccfg -s vntsd setprop vntsd/authorization = true
primary# svcadm refresh vntsd
primary# svcadm restart vntsd
primary# ldm ls
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART    8     16G      0.2%  47m
ldg1             active     -n--v-  5000    2     1G       0.1%  17h 50m
ldg2             active     -t----  5001    4     2G        25%  11s
```

L'exemple suivant décrit la création du rôle `all_cons` avec l'autorisation `solaris.vntsd.consoles`, laquelle permet d'accéder à toutes les consoles de domaines.

```
primary# roleadd -A solaris.vntsd.consoles all_cons
primary# passwd all_cons
New Password:
Re-enter new Password:
passwd: password successfully changed for all_cons
```

Cette commande affecte le rôle `all_cons` à l'utilisateur `sam`.

```
primary# usermod -R all_cons sam
```

L'utilisateur `sam` prend le rôle `all_cons` et peut accéder à n'importe quelle console. Par exemple :

```
$ id
uid=700299(sam) gid=1(other)
$ su all_cons
Password:
$ telnet localhost 5000
Trying 0.0.0.0...
Connected to 0.
Escape character is '^]'.

Connecting to console "ldg1" in group "ldg1" ....
Press ~? for control options ..

$ telnet localhost 5001
Trying 0.0.0.0...
Connected to 0.
Escape character is '^]'.

Connecting to console "ldg2" in group "ldg2" ....
Press ~? for control options ..
```

Cet exemple montre ce qui se produit lorsqu'un utilisateur non autorisé, `dana`, tente d'accéder à une console de domaine :

```
$ id
```

```
uid=702048(dana) gid=1(other)
$ telnet localhost 5000
Trying 0.0.0.0...
Connected to 0.
Escape character is '^]'.
Connection to 0 closed by foreign host.
```

▼ Procédure de contrôle de l'accès à toutes les consoles de domaines par le biais de profils de droits

1. **Restreignez l'accès à une console de domaine en activant la vérification des autorisations de la console.**

   ```
   primary# svccfg -s vntsd setprop vntsd/authorization = true
   primary# svcadm refresh vntsd
   primary# svcadm restart vntsd
   ```

2. **Créez un profil de droits comportant l'autorisation `solaris.vntsd.consoles`.**
   Utilisez la commande `profiles` pour créer un profil.

   ```
   primary# profiles -p "LDoms Consoles" \
   'set desc="Access LDoms Consoles"; set auths=solaris.vntsd.consoles'
   ```

3. **Affectez le profil de droits à un utilisateur.**

   ```
   primary# usermod -P +"LDoms Consoles" username
   ```

4. **Connectez-vous à la console du domaine sous le nom de d'utilisateur concerné.**

   ```
   $ telnet localhost 5000
   ```

Exemple 3 Contrôle de l'accès à toutes les consoles de domaines par le biais de profils de droits

Les exemples qui suivent illustrent l'utilisation de profils de droits pour contrôler l'accès à toutes les consoles de domaine. Utilisez la commande `profiles` pour créer un profil de droits présentant l'autorisation `solaris.vntsd.consoles` dans la base de données de description des profils de droits.

```
primary# profiles -p "LDoms Consoles" \
'set desc="Access LDoms Consoles"; set auths=solaris.vntsd.consoles'
```

Affectez le profil de droits à un utilisateur.

```
primary# usermod -P +"LDoms Consoles" sam
```

Les commandes suivantes permettent de vérifier que l'utilisateur est bien `sam` et que les profils de droits `All`, `Basic Solaris User` et `LDoms Consoles` sont effectifs. La commande `telnet` permet d'accéder à la console du domaine `ldg1`.

```
$ id
uid=702048(sam) gid=1(other)
$ profiles
All
Basic Solaris User
LDoms Consoles
$ telnet localhost 5000
Trying 0.0.0.0...
Connected to 0.
Escape character is '^]'.

Connecting to console "ldg1" in group "ldg1" ....
Press ~? for control options ..
```

▼ Procédure de contrôle de l'accès à une console unique par le biais de rôles

1. **Restreignez l'accès à une console de domaine en activant la vérification des autorisations de la console.**

   ```
   primary# svccfg -s vntsd setprop vntsd/authorization = true
   primary# svcadm refresh vntsd
   primary# svcadm restart vntsd
   ```

2. **Ajoutez une autorisation pour un seul domaine à la base de données de description des autorisations.**

   Le nom de l'autorisation est dérivé du nom du domaine et se présente sous la forme : `solaris.vntsd.console-`*domain* :

   ```
   solaris.vntsd.console-domain:::Access domain Console::
   ```

3. **Créez un rôle possédant la nouvelle autorisation pour n'autoriser l'accès qu'à la console du domaine.**

   ```
   primary# roleadd -A solaris.vntsd.console-domain role-name
   primary# passwd role-name
   New Password:
   Re-enter new Password:
   passwd: password successfully changed for role-name
   ```

4. **Assignez le rôle *role-name* à un utilisateur.**

   ```
   primary# usermod -R role-name username
   ```

Exemple 4 Accès à une console de domaine unique

Dans cet exemple, l'utilisateur `terry` prend le rôle `ldg1cons` et accède à la console du domaine `ldg1`.

Ajoutez tout d'abord une autorisation pour un domaine unique, `ldg1`, dans la base de données de description des autorisations.

```
solaris.vntsd.console-ldg1:::Access ldg1 Console::
```

Créez ensuite un rôle possédant la nouvelle autorisation pour n'autoriser l'accès qu'à la console du domaine.

```
primary# roleadd -A solaris.vntsd.console-ldg1 ldg1cons
primary# passwd ldg1cons
New Password:
Re-enter new Password:
passwd: password successfully changed for ldg1cons
```

Affectez le rôle `ldg1cons` à l'utilisateur `terry`, prenez le rôle `ldg1cons` et accédez à la console de domaine.

```
primary# usermod -R ldg1cons terry
primary# su terry
Password:
$ id
uid=700300(terry) gid=1(other)
$ su ldg1cons
Password:
$ id
uid=700303(ldg1cons) gid=1(other)
$ telnet localhost 5000
Trying 0.0.0.0...
Escape character is '^]'.

Connecting to console "ldg1" in group "ldg1" ....
Press ~? for control options ..
```

L'exemple ci-dessous indique que l'utilisateur `terry` ne peut pas accéder à la console de domaine `ldg2` :

```
$ telnet localhost 5001
Trying 0.0.0.0...
Connected to 0.
Escape character is '^]'.
Connection to 0 closed by foreign host.
```

▼ Procédure de contrôle de l'accès à une console unique par le biais de profils de droits

1. **Restreignez l'accès à une console de domaine en activant la vérification des autorisations de la console.**

```
primary# svccfg -s vntsd setprop vntsd/authorization = true
primary# svcadm refresh vntsd
primary# svcadm restart vntsd
```

2. **Ajoutez une autorisation pour un seul domaine à la base de données de description des autorisations.**

   L'exemple d'entrée suivant ajoute l'autorisation pour une console de domaine :

   ```
   solaris.vntsd.console-domain:::Access domain Console::
   ```

3. **Créez un profil de droits comportant une autorisation d'accès à une console de domaine particulière.**

   Utilisez la commande `profiles` pour créer un profil.

   ```
   primary# profiles -p "domain Console" \
   'set desc="Access domain Console";
   set auths=solaris.vntsd.console-domain'
   ```

4. **Assignez le profil de droits.**

   ```
   primary# usermod -P +"domain Console" username
   ```

Utilisation de la journalisation des consoles de domaines

Dans un environnement Oracle VM Server for SPARC, la console d'E/S du domaine `primary` est dirigée vers le processeur de service (SP). Les E/S des consoles de tous les autres domaines sont redirigées vers le domaine de service qui exécute le concentrateur de console virtuelle, `vcc`. Si le domaine de service exécute le SE Oracle Solaris 11, la sortie de la console du domaine invité peut être consignée dans un fichier.

Les domaines de service prennent en charge la journalisation des consoles des domaines logiques. Tandis que le domaine de service doit exécuter le SE Oracle Solaris 11, le domaine hôte consigné peut exécuter le SE Oracle Solaris 10 ou le SE Oracle Solaris 11.

Le journal des consoles de domaines est enregistré dans un fichier sur le domaine de service appelé `/var/log/vntsd/`*domain*`/console-log` fournissant le service `vcc`. Vous pouvez faire pivoter les fichiers journaux de la console à l'aide de la commande `logadm`. Reportez-vous aux pages de manuel `logadm`(1M) et `logadm.conf`(4).

Le logiciel Oracle VM Server for SPARC vous permet de désactiver et d'activer de façon sélective la journalisation des consoles pour chaque domaine logique. La journalisation des consoles est activée par défaut.

▼ Procédure d'activation et de désactivation de la journalisation des consoles

Vous devez activer ou désactiver la journalisation des consoles pour chaque domaine logique, même si les domaines appartiennent au même groupe de consoles.

1. **Répertoriez les paramètres de la console actuels pour le domaine.**

   ```
   primary# ldm list -o console domain
   ```

2. **Arrêtez et dissociez le domaine.**

   L'état du domaine doit être inactif et non lié avant que vous ne puissiez modifier les paramètres de la console.

   ```
   primary# ldm stop domain
   primary# ldm unbind domain
   ```

3. **Activez ou désactivez la journalisation des consoles.**

   - **Pour activer la journalisation des consoles.**

     ```
     primary# ldm set-vcons log=on domain
     ```

   - **Pour désactiver la journalisation des consoles.**

     ```
     primary# ldm set-vcons log=off domain
     ```

Configuration requise du domaine de service pour la journalisation des consoles de domaines

Un domaine relié à un domaine de service exécutant une version de SE antérieure à Oracle Solaris 11.1 *ne peut pas* être journalisé.

Remarque - Même si vous activez la journalisation des consoles pour un domaine, la console virtuelle du domaine n'est pas journalisée si la prise en charge requise n'est pas assurée sur le domaine de service.

Connexion à une console invitée sur le réseau

Vous pouvez vous connecter à une console invitée sur un réseau si la propriété `listen_addr` est définie sur l'adresse IP du domaine de contrôle dans le manifeste SMF `vntsd`(1M). Par exemple :

```
$ telnet hostname 5001
```

Remarque - L'activation de l'accès réseau à une console a des implications de sécurité. N'importe quel utilisateur peut se connecter à une console et pour cette raison, elle est désactivée par défaut.

Un manifeste d'utilitaire de gestion des services est un fichier XML qui décrit un service. Pour plus d'informations sur la création d'un manifeste SMF, reportez-vous à la documentation destinée aux administrateurs système d'Oracle Solaris 10 (`http://download.oracle.com/docs/cd/E18752_01/index.html`).

Remarque - Pour accéder à un SE non anglais dans un domaine invité via la console, le terminal de la console doit être configuré avec l'environnement linguistique requis par le SE.

Utilisation de groupes de consoles

Le démon du serveur de terminal du réseau virtuel, `vntsd`, vous permet de fournir l'accès à plusieurs consoles de domaines à l'aide d'un seul port TCP. Après la création du domaine, Logical Domains Manager assigne un port TCP unique à chaque console en créant un nouveau groupe par défaut pour cette console du domaine. Le port TCP est ensuite assigné au groupe de consoles au contraire de la console elle-même. La console peut être liée à un groupe existant à l'aide de la sous-commande `set-vcons`.

▼ Procédure d'association de plusieurs consoles en un groupe

1. **Associez les consoles pour les domaines en un groupe unique.**

   L'exemple suivant présente l'association de la console pour trois domaines différents (`ldg1`, `ldg2` et `ldg3`) au même groupe de consoles (`group1`).

   ```
   primary# ldm set-vcons group=group1 service=primary-vcc0 ldg1
   ```

```
primary# ldm set-vcons group=group1 service=primary-vcc0 ldg2
primary# ldm set-vcons group=group1 service=primary-vcc0 ldg3
```

2. **Connectez le port TCP associé (`localhost` sur le port `5000` dans cet exemple).**

```
# telnet localhost 5000
primary-vnts-group1: h, l, c{id}, n{name}, q:
```

Vous êtes invité à sélectionné une des consoles du domaine.

3. **Répertoriez les domaines dans le groupe en sélectionnant `l` (liste).**

```
primary-vnts-group1: h, l, c{id}, n{name}, q: l
DOMAIN ID           DOMAIN NAME                  DOMAIN STATE
0                   ldg1                         online
1                   ldg2                         online
2                   ldg3                         online
```

Remarque - Pour vous permettre de réassigner la console à un groupe différent ou à une instance `vcc`, le domaine doit être dissocié, c'est-à-dire qu'il doit être inactif. Reportez-vous à la page de manuel `vntsd`(1M) pour plus d'informations sur la configuration et l'utilisation de SMF pour gérer `vntsd` et sur l'utilisation de groupes de consoles.

♦ ♦ ♦ CHAPITRE 6

Configuration de domaines d'E/S

Ce chapitre décrit les domaines d'E/S et la procédure de configuration de ces derniers dans un environnement d'Oracle VM Server for SPARC.

Ce chapitre aborde les sujets suivants :

Présentation d'un domaine d'E/S

Un domaine d'E/S est détenteur direct de périphériques d'E/S physiques et dispose d'un accès direct à ces derniers. Il peut être créé en assignant un bus PCI Express (PCIe), un périphérique d'extrémité PCIe ou une fonction virtuelle SR-IOV PCIe à un domaine. Exécutez la commande `ldm add-io` pour assigner un bus, un périphérique ou une fonction virtuelle à un domaine.

Les domaines d'E/S présentent les avantages suivants :

- Un domaine d'E/S dispose d'un accès direct à un périphérique d'E/S physique, ce qui évite les baisses de performances associées aux E/S virtuelles. Par conséquent, les performances d'E/S sur un domaine d'E/S sont plus proches des performances d'E/S sur un système brut.
- Un domaine d'E/S peut héberger des services d'E/S virtuels pour une utilisation par des domaines invités.

Pour plus d'informations sur la configuration des domaines d'E/S, reportez-vous aux informations des chapitres suivants :

- Chapitre 7, Création d'un domaine root par assignation de bus PCIe
- Chapitre 9, Création d'un domaine d'E/S en utilisant les E/S directes
- Chapitre 8, Création d'un domaine d'E/S à l'aide des fonctions virtuelles SR-IOV PCIe
- Chapitre 10, Utilisation de domaines root différents du domaine `primary`

Remarque - Il est impossible de migrer un domaine comportant des bus PCIe, des périphériques d'extrémité PCIe ou des fonctions virtuelles SR-IOV. Pour plus d'informations sur les autres restrictions applicables à la migration, voir Chapitre 14, Migration des domaines.

Lignes directrices pour la création d'un domaine d'E/S

Un domaine d'E/S peut disposer d'un accès direct à un ou plusieurs périphériques d'E/S tels que des bus PCIe, des NIU (Network Interface Unit, unité d'interface réseau), des périphériques d'extrémité PCIe et des fonctions virtuelles SR-IOV (single root I/O Virtualization, virtualisation d'E/S à racine unique) PCIe.

Ce type d'accès direct aux périphériques d'E/S se traduit par une augmentation de la bande passante d'E/S disponible pour assurer les services suivants :

- Services aux applications dans le domaine d'E/S
- Services E/S virtuels fournis aux domaines invités

Les règles de base suivantes vous permettent d'exploiter au mieux la bande passante d'E/S :

- Affectez les ressources CPU avec une précision de l'ordre du coeur de CPU. Affectez un ou plusieurs coeurs de processeur en fonction du type de périphérique d'E/S et du nombre de périphériques d'E/S dans le domaine d'E/S.

  Par exemple, pour exploiter l'ensemble de la bande passante, un périphérique Ethernet 1 Gbit/s peut nécessiter un plus petit nombre de coeurs de processeur qu'un périphérique Ethernet 10 Gbit/s.
- Respectez la configuration requise en ce qui concerne la mémoire. La mémoire requise dépend du type de périphérique d'E/S assigné au domaine. Un minimum de 4 Go est recommandé par périphérique d'E/S. Plus vous affectez de périphériques d'E/S, plus la mémoire devant être allouée augmente.
- Lorsque vous utilisez la fonctionnalité SR-IOV PCIe, suivez pour chaque fonction virtuelle SR-IOV les mêmes règles que vous suivriez pour d'autres périphériques d'E/S. Assignez donc un ou plusieurs coeurs de CPU et de la mémoire (en Go) afin d'exploiter pleinement la bande passante disponible grâce à la fonction virtuelle.

Notez que la création et l'affectation d'un grand nombre de fonctions virtuelles à un domaine qui ne dispose pas d'un nombre suffisant de ressources CPU et de mémoire ont peu de chances d'aboutir à une configuration optimale.

Les systèmes SPARC, jusqu'aux plates-formes SPARC T5 et SPARC M6 incluses, indiquent un nombre limité d'interruptions afin qu'Oracle Solaris limite le nombre d'interruptions utilisables par chaque périphérique. La limite par défaut doit correspondre aux besoins d'une

configuration système type, sachant que la valeur devra peut-être être ajustée pour certaines configurations système. Pour plus d'informations, reportez-vous à la section "Ajustement de la limite d'interruption" à la page 427.

♦ ♦ ♦ CHAPITRE 7

Création d'un domaine root par assignation de bus PCIe

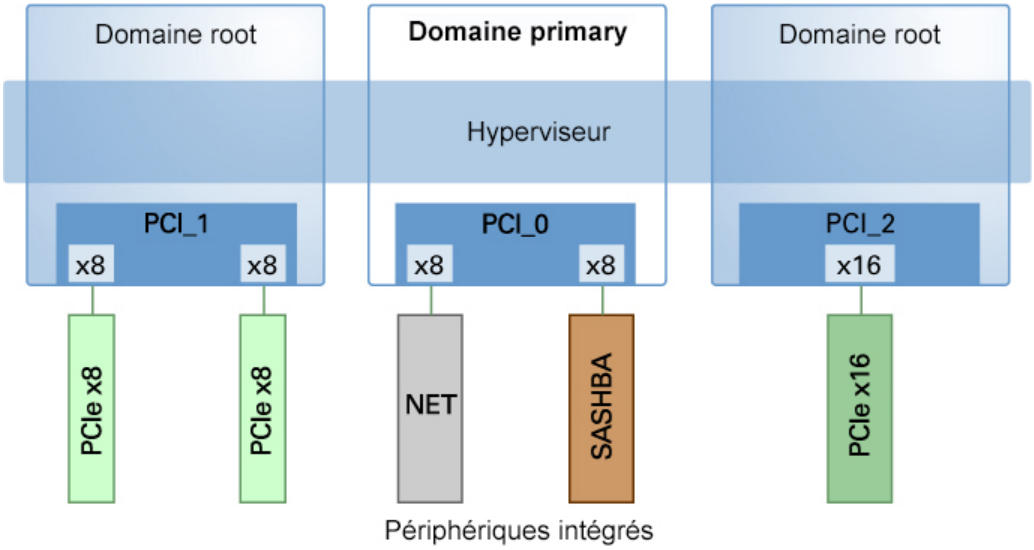
Ce chapitre indique comment créer un domaine root en affectant des bus PCIe.

Création d'un domaine root par assignation de bus PCIe

Vous pouvez utiliser le logiciel Oracle VM Server for SPARC pour assigner un bus PCIe entier (également appelé *root complex*) à un domaine. Un bus PCIe entier est composé du bus PCIe lui-même et de tous ses commutateurs et périphériques PCI. Les bus PCIe présents sur un serveur sont identifiés par des noms tels que `pci@400` (`pci_0`). Un domaine d'E/S configuré avec un bus PCIe entier est également appelé *root domain*.

Le schéma suivant représente un système avec trois complexes root `pci_0`, `pci_1` et `pci_2`.

FIGURE 2 Affectation d'un bus PCIe à un domaine root



Le nombre maximal de domaines root que vous pouvez créer avec des bus PCIe dépend du nombre de bus PCIe disponibles sur le serveur. Déterminez le nombre de bus PCIe disponibles sur votre système à l'aide de la commande `ldm list-io`

Lorsque vous affectez un bus PCIe à un domaine root, tous les périphériques sur ce bus sont détenus par le domaine root concerné. Vous pouvez assigner n'importe lequel des périphériques d'extrémité PCIe sur ce bus à d'autres domaines.

Si un serveur est configuré dès le départ dans un environnement Oracle VM Server for SPARC ou utilise la configuration `factory-default`, le domaine `primary` a accès à toutes les ressources du périphérique physique. Par conséquent, le domaine `primary` est le seul domaine root configuré sur le système et il est propriétaire de tous les bus PCIe.

Affectation de bus PCIe statiques

La méthode d'affectation de bus PCIe statique d'un domaine root nécessite l'initialisation d'une reconfiguration retardée sur le domaine root lors de l'affectation ou de la suppression d'un bus PCIe. Si vous avez l'intention d'utiliser cette méthode pour un domaine qui n'a pas encore de bus PCIe, vous devez arrêter le domaine avant de pouvoir affecter le bus PCIe. Après avoir

effectué les étapes de configuration sur le domaine root, vous devez le réinitialiser. Vous devez utiliser cette méthode statique lorsque le microprogramme Oracle VM Server for SPARC 3.2 n'est pas installé dans le système ou lorsque la version du SE installée dans le domaine concerné ne prend pas en charge affectation de bus PCIe dynamique.

Pendant que le domaine root est arrêté ou en reconfiguration retardée, vous pouvez exécuter au moins l'une des commandes `ldm add-io` ou `ldm remove-io` avant de réinitialiser le domaine root. Afin de minimiser le temps d'arrêt, prévoyez à l'avance toute affectation ou suppression de bus PCIe.

- Pour les domaines root, `primary` et non-`primary` utilisez la reconfiguration retardée. Après avoir ajouté ou supprimé les bus PCIe, réinitialisez le domaine root pour que les changements prennent effet.

  ```
  primary# ldm start-reconf root-domain
  Add or remove the PCIe bus by using the ldm add-io or ldm remove-io command
  primary# ldm stop -r domain-name
  ```

  Notez que vous pouvez utiliser la reconfiguration retardée uniquement si le domaine possède déjà un bus PCIe.
- Pour les domaines non-root, arrêtez le domaine, puis ajoutez ou supprimez le bus PCIe.

  ```
  primary# ldm stop domain-name
  Add or remove the PCIe bus by using the ldm add-io or ldm remove-io command
  primary# ldm start domain-name
  ```

Affectation de bus PCIe dynamique

La fonction affectation de bus PCIe dynamique permet d'affecter ou de supprimer dynamiquement un bus PCIe à partir d'un domaine root.

La fonction affectation de bus PCIe dynamique est activée lorsque votre système exécute le microprogramme et le logiciel requis. Voir "Contraintes liées à l'Affectation de bus PCIe dynamique" à la page 73. Si le système n'exécute pas les microprogrammes et logiciels requis, les commandes `ldm add-io` et `ldm remove-io` échouent.

Lorsque cette option est activée, vous pouvez exécuter les commandes `ldm add-io` et `ldm remove-io` sans qu'il soit nécessaire d'arrêter le domaine root ni de le mettre en reconfiguration retardée.

Contraintes liées à l'Affectation de bus PCIe dynamique

La fonctionnalité affectation de bus PCIe dynamique est prise en charge sur les serveurs SPARC M5, les serveurs SPARC M6, les serveurs des séries SPARC M7, SPARC T7 et

SPARC S7 et les Serveurs Fujitsu M10 qui exécutent le SE Oracle Solaris 11 dans le domaine root. Les serveurs SPARC M5 et SPARC M6 doivent au moins exécuter la version 9.4.2 du microprogramme système, les serveurs des séries SPARC T7 et SPARC M7 doivent exécuter au moins la version 9.4.3, les serveurs de série SPARC S7 peuvent exécuter n'importe quelle version publiée du microprogramme système et le Serveur Fujitsu M10 doit exécuter au moins XCP2240.

▼ Procédure de création d'un domaine root par affectation d'un bus PCIe

Cet exemple de procédure indique comment créer un nouveau domaine root à partir d'une configuration initiale où plusieurs bus appartiennent au domaine `primary`. Par défaut, le domaine `primary` détient tous les bus présents sur le système. L'exemple suivant concerne le serveur SPARC T4-2. Cette procédure peut également être utilisée sur d'autres serveurs. Les instructions pour les différents serveurs peuvent varier légèrement de celles-ci, mais cet exemple illustre les principes de base.

Assurez-vous de ne pas supprimer les bus PCIe qui hébergent le disque d'initialisation et l'interface réseau principale du domaine `primary`.

Attention - Tous les disques internes sur les serveurs pris en charge peuvent être connectés à un seul bus PCIe. Si un domaine est initialisé à partir d'un disque interne, ne supprimez pas le bus concerné de ce domaine.

Ne supprimez pas un bus dont les périphériques, tels que les ports réseau ou périphériques `usbecm` sont utilisés par un domaine. Si vous supprimez le mauvais bus, le domaine risque de ne pas pouvoir accéder aux périphériques requis et peut devenir inutilisable. Pour supprimer un bus qui a des périphériques utilisés par un domaine, reconfigurez ce domaine pour qu'il utilise des périphériques d'autres bus. Par exemple, vous devrez peut-être reconfigurer le domaine pour qu'il utilise un port réseau intégré différent ou une carte PCIe d'un emplacement PCIe différent.

Sur certains serveurs SPARC, vous pouvez supprimer un bus PCIe qui contient des contrôleurs de graphiques USB et d'autres périphériques. Toutefois, vous ne pouvez pas ajouter de bus PCIe à d'autres domaines. Ce type de bus PCIe ne peut être ajouté qu'au domaine `primary`.

Dans cet exemple, le domaine `primary` n'utilise qu'un pool ZFS (`rpool`) et une interface réseau (`igb0`). Si le domaine `primary` utilise plusieurs périphériques, répétez les étapes 2 à 4 pour chaque périphérique afin de vous assurer qu'aucun d'entre eux n'est situé sur le bus qui va être supprimé.

Vous pouvez ajouter un bus de manière à supprimer un bus d'un domaine à l'aide de son chemin de périphérique (`pci@`*nnn*) ou de son pseudonyme (`pci_`*n*). La commande `ldm list-bindings primary` ou `ldm list -l -o physio primary` affiche ce qui suit :

- `pci@400` correspond à `pci_0`
- `pci@500` correspond à `pci_1`
- `pci@600` correspond à `pci_2`
- `pci@700` correspond à `pci_3`

1. Vérifiez que le domaine `primary` détient plusieurs bus PCIe.

```
primary# ldm list-io
NAME                                      TYPE   BUS      DOMAIN   STATUS
----                                      ----   ---      ------   ------
pci_0                                     BUS    pci_0    primary
pci_1                                     BUS    pci_1    primary
pci_2                                     BUS    pci_2    primary
pci_3                                     BUS    pci_3    primary
/SYS/MB/PCIE1                             PCIE   pci_0    primary  EMP
/SYS/MB/SASHBA0                           PCIE   pci_0    primary  OCC
/SYS/MB/NET0                              PCIE   pci_0    primary  OCC
/SYS/MB/PCIE5                             PCIE   pci_1    primary  EMP
/SYS/MB/PCIE6                             PCIE   pci_1    primary  EMP
/SYS/MB/PCIE7                             PCIE   pci_1    primary  EMP
/SYS/MB/PCIE2                             PCIE   pci_2    primary  EMP
/SYS/MB/PCIE3                             PCIE   pci_2    primary  EMP
/SYS/MB/PCIE4                             PCIE   pci_2    primary  EMP
/SYS/MB/PCIE8                             PCIE   pci_3    primary  EMP
/SYS/MB/SASHBA1                           PCIE   pci_3    primary  OCC
/SYS/MB/NET2                              PCIE   pci_3    primary  OCC
/SYS/MB/NET0/IOVNET.PF0                   PF     pci_0    primary
/SYS/MB/NET0/IOVNET.PF1                   PF     pci_0    primary
/SYS/MB/NET2/IOVNET.PF0                   PF     pci_3    primary
/SYS/MB/NET2/IOVNET.PF1                   PF     pci_3    primary
```

2. Déterminez le chemin du périphérique du disque d'initialisation qui doit être conservé.

- **Pour les systèmes de fichiers UFS, exécutez la commande `df /` pour déterminer le chemin de périphérique du disque d'initialisation.**

```
primary# df /
/                  (/dev/dsk/c0t5000CCA03C138904d0s0):22755742 blocks  2225374 files
```

- **Pour les systèmes de fichiers ZFS, exécutez d'abord la commande `df /` pour déterminer le nom de pool. Puis exécutez la commande `zpool status` pour déterminer le chemin de périphérique du disque d'initialisation.**

```
primary# zpool status rpool
  pool: rpool
 state: ONLINE
  scan: none requested
```

```
config:

        NAME                       STATE     READ WRITE CKSUM
        rpool                      ONLINE       0     0     0
          c0t5000CCA03C138904d0s0  ONLINE       0     0     0
```

3. **Obtenez des informations sur le disque d'initialisation du système.**

   - **S'il s'agit d'un disque d'initialisation géré à l'aide la fonctionnalité de multipathing d'E/S d'Solaris, identifiez le bus PCIe sous lequel il est connecté en exécutant la commande `mpathadm`.**

     a. **Recherchez le port initiateur auquel le disque est connecté.**

     ```
     primary# mpathadm show lu /dev/rdsk/c0t5000CCA03C138904d0s0
     Logical Unit:  /dev/rdsk/c0t5000CCA03C138904d0s2
            mpath-support:  libmpscsi_vhci.so
            Vendor:  HITACHI
            Product:  H106030SDSUN300G
            Revision:  A2B0
            Name Type:  unknown type
            Name:  5000cca03c138904
            Asymmetric:  no
            Current Load Balance:  round-robin
            Logical Unit Group ID:  NA
            Auto Failback:  on
            Auto Probing:  NA

            Paths:
                    Initiator Port Name:  w50800200014100c8
                    Target Port Name:  w5000cca03c138905
                    Override Path:  NA
                    Path State:  OK
                    Disabled:  no

            Target Ports:
                    Name:  w5000cca03c138905
                    Relative ID:  0
     ```

     b. **Identifiez le bus PCIe sur lequel le port initiateur est présent.**

     ```
     primary# mpathadm show initiator-port w50800200014100c8
     Initiator Port:  w50800200014100c8
            Transport Type:  unknown
            OS Device File:  /devices/pci@400/pci@2/pci@0/pci@e/scsi@0/iport@1
     ```

   - **Pour un disque qui n'est pas géré avec la fonctionnalité multipathing d'E/S d'Solaris, déterminez le périphérique physique auquel le périphérique en mode bloc est connecté à l'aide de la commande `ls -l`.**

     L'exemple suivant utilise le périphérique en mode bloc `c1t0d0s0` :

     ```
     primary# ls -l /dev/dsk/c0t1d0s0
     lrwxrwxrwx   1 root     root          49 Oct  1 10:39 /dev/dsk/c0t1d0s0 ->
     ../../devices/pci@400/pci@0/pci@1/scsi@0/sd@1,0:a
     ```

Dans cet exemple, le périphérique physique du disque d'initialisation du domaine `primary` est connecté au bus `pci@400`.

4. Identifiez l'interface réseau utilisée par le système.

Identifiez l'interface du réseau principal qui est “raccordée” à l'aide de la commande `ifconfig`. Une interface raccordée contient des flux configurés de manière à ce que le protocole IP puisse utiliser le périphérique.

```
primary# ifconfig -a
lo0: flags=2001000849<UP,LOOPBACK,RUNNING,MULTICAST,IPv4,VIRTUAL> mtu 8232 index 1
        inet 127.0.0.1 netmask ff000000
net0: flags=1004843<UP,BROADCAST,RUNNING,MULTICAST,DHCP,IPv4> mtu 1500 index 3
        inet 10.129.241.135 netmask ffffff00 broadcast 10.129.241.255
        ether 0:10:e0:e:f1:78

primary# dladm show-phys net0
LINK              MEDIA               STATE      SPEED  DUPLEX    DEVICE
net0              Ethernet            up         1000   full      igb0
```

5. Identifiez le périphérique physique auquel l'interface réseau est connectée.

La commande suivante utilise l'interface réseau `igb0` :

```
primary# ls -l /dev/igb0
lrwxrwxrwx   1 root     root          46 Oct  1 10:39 /dev/igb0  ->
../devices/pci@500/pci@0/pci@c/network@0:igb0
```

Lancez également la commande `ls -l /dev/usbecm`.

Dans cet exemple, le périphérique physique utilisé par le domaine `primary` se trouve sous le bus `pci@500`, qui correspond à la liste précédente de `pci_1`. Par conséquent, les deux autres bus, `pci_2` (`pci@600`) et `pci_3` (`pci@700`), peuvent être assignés à d'autres domaines en toute sécurité, car ils ne sont pas utilisés par le domaine `primary`.

Si l'interface réseau utilisée par le domaine `primary` se trouve sur un bus que vous voulez affecter à un autre domaine, reconfigurez le domaine `primary` pour utiliser une interface réseau différente.

6. Supprimez du domaine `primary` les bus qui ne contiennent pas le disque d'initialisation ou l'interface réseau.

Dans cet exemple, le bus `pci_2` est supprimé du domaine `primary`.

- **Méthode dynamique :**

  Assurez-vous que les périphériques du bus `pci_2` ne sont pas en cours d'utilisation par le SE du domaine `primary`. Si tel est le cas, cette commande pourra ne pas parvenir à supprimer le bus. Pour forcer la suppression du bus `pci_2`, utilisez la méthode statique.

  ```
  primary# ldm remove-io pci_2 primary
  ```

- **Méthode statique :**

  Avant de supprimer le bus, vous devez démarrer une reconfiguration retardée.

```
primary# ldm start-reconf primary
primary# ldm remove-io pci_2 primary
primary# shutdown -y -g0 -i6
```

Le bus utilisé par le domaine `primary` pour le disque d'initialisation et le périphérique réseau ne peut pas être affecté à d'autres domaines. Vous pouvez affecter n'importe lequel des autres bus à un autre domaine. Dans cet exemple, `pci@600` n'est pas utilisé par le domaine `primary` et peut donc être réaffecté à un autre domaine.

7. **Ajoutez un bus à un domaine.**

   Dans cet exemple, vous ajoutez le bus `pci_2` au domaine `ldg1`.

   - **Méthode dynamique :**

```
primary# ldm add-io pci_2 ldg1
```

   - **Méthode statique :**

     Avant d'ajouter le bus, vous devez arrêter le domaine.

```
primary# ldm stop-domain ldg1
primary# ldm add-io pci_2 ldg1
primary# ldm start-domain ldg1
```

8. **Enregistrez cette configuration sur le processeur de service.**

   Dans cet exemple, la configuration est `io-domain`.

```
primary# ldm add-config io-domain
```

   Cette configuration, `io-domain`, est également définie comme la configuration suivante à utiliser après la réinitialisation.

9. **Vérifiez que le bus correct est toujours assigné au domaine `primary` et que le bus correct est assigné au domaine `ldg1`.**

```
primary# ldm list-io
NAME                                      TYPE   BUS      DOMAIN   STATUS
----                                      ----   ---      ------   ------
pci_0                                     BUS    pci_0    primary
pci_1                                     BUS    pci_1    primary
pci_2                                     BUS    pci_2    ldg1
pci_3                                     BUS    pci_3    primary
/SYS/MB/PCIE1                             PCIE   pci_0    primary  EMP
/SYS/MB/SASHBA0                           PCIE   pci_0    primary  OCC
/SYS/MB/NET0                              PCIE   pci_0    primary  OCC
/SYS/MB/PCIE5                             PCIE   pci_1    primary  EMP
/SYS/MB/PCIE6                             PCIE   pci_1    primary  EMP
/SYS/MB/PCIE7                             PCIE   pci_1    primary  EMP
```

```
/SYS/MB/PCIE2                          PCIE   pci_2    ldg1     EMP
/SYS/MB/PCIE3                          PCIE   pci_2    ldg1     EMP
/SYS/MB/PCIE4                          PCIE   pci_2    ldg1     EMP
/SYS/MB/PCIE8                          PCIE   pci_3    primary  EMP
/SYS/MB/SASHBA1                        PCIE   pci_3    primary  OCC
/SYS/MB/NET2                           PCIE   pci_3    primary  OCC
/SYS/MB/NET0/IOVNET.PF0                PF     pci_0    primary
/SYS/MB/NET0/IOVNET.PF1                PF     pci_0    primary
/SYS/MB/NET2/IOVNET.PF0                PF     pci_3    primary
/SYS/MB/NET2/IOVNET.PF1                PF     pci_3    primary
```

Cette sortie confirme que les bus PCIe `pci_0`, `pci_1` et `pci_3` ainsi que leurs périphériques sont assignés au domaine `primary`. Elle confirme également que le bus PCIe `pci_2` et ses périphériques sont assignés au domaine `ldg1`.

♦ ♦ ♦ CHAPITRE 8

Création d'un domaine d'E/S à l'aide des fonctions virtuelles SR-IOV PCIe

Ce chapitre contient les sujets relatifs à SR-IOV PCIe suivants :

Présentation de SR-IOV

Remarque - Etant donné que les domaines root ne peuvent pas présenter de dépendances à d'autres domaines root, les périphériques d'extrémité PCIe et les fonctions virtuelles SR-IOV présents sur un bus PCIe détenu par un domaine root ne peuvent pas être assignées à un autre domaine root. En revanche, un périphérique d'extrémité PCIe ou une fonction virtuelle de bus PCIe *peuvent* être affectés au domaine root propriétaire du bus.

La virtualisation d'E/S à racine unique PCIe (Peripheral Component Interconnect Express) est basée sur la version 1.1 de la norme telle que définie par le consortium PCI-SIG. La norme

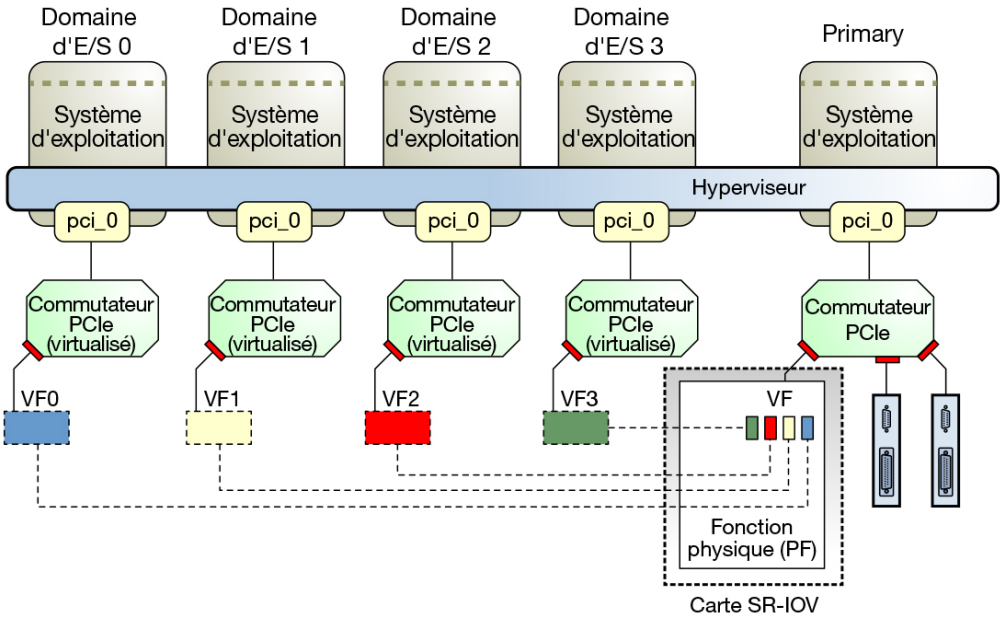
SR-IOV permet le partage efficace de périphériques PCIe entre des machines virtuelles et est implémentée dans le matériel afin d'atteindre des performances d'E/S comparables à des performances natives. La spécification SR-IOV définit une nouvelle norme dans laquelle les nouveaux périphériques créés permettent à la machine virtuelle d'être directement connectée au périphérique d'E/S.

Une ressource d'E/S unique, appelée *fonction physique*, peut être partagée par plusieurs machines virtuelles. Les périphériques partagés fournissent des ressources dédiées et utilisent également des ressources communes partagées. De cette manière, chaque machine virtuelle a accès à des ressources uniques. Par conséquent, sous réserve de prise en charge par le SE et de matériel approprié, un périphérique PCIe tel qu'un port Ethernet sur lequel la fonction SR-IOV est activée peut apparaître sous la forme de plusieurs périphériques physiques distincts possédant chacun son propre espace de configuration PCIe.

Pour plus d'informations sur SR-IOV, visitez le site Web de PCI-SIG (`http://www.pcisig.com/`).

La figure suivante illustre la relation entre les fonctions virtuelles et une fonction physique dans un domaine d'E/S.

FIGURE 3 Utilisation de fonctions virtuelles (VF) et d'une fonction physique (PF) dans un domaine d'E/S.



SR-IOV présente les types de fonctions suivants :

- **Fonction physique** : fonction PCI prenant en charge les fonctionnalités SR-IOV telles que définies dans la norme SR-IOV. Une fonction physique contient la structure des fonctions SR-IOV et gère la fonctionnalité SR-IOV. Les fonctions physiques sont des fonctions PCIe à part entière qui peuvent être découvertes, gérées et manipulées comme n'importe quel autre périphérique PCIe. Les fonctions physiques peuvent être utilisées pour configurer et contrôler un périphérique PCIe.
- **Fonction virtuelle** : fonction PCI associée à une fonction physique. Une fonction virtuelle est une fonction PCIe allégée qui partage une ou plusieurs ressources physiques avec la fonction physique et avec d'autres fonctions virtuelles associées à la même fonction physique. A la différence d'une fonction physique, une fonction virtuelle peut uniquement configurer son propre comportement.

Chaque périphérique SR-IOV peut avoir une fonction physique et chaque fonction physique peut se voir associer jusqu'à 256 fonctions virtuelles. Ce nombre varie en fonction du périphérique SR-IOV concerné. Les fonctions virtuelles sont créées par la fonction physique.

Une fois que SR-IOV est activé dans la fonction physique, l'espace de configuration PCI de chaque fonction virtuelle est accessible via le numéro de bus, de périphérique et de fonction de la fonction physique. Chaque fonction virtuelle dispose d'un espace mémoire PCI utilisé pour mapper son jeu de registres. Les pilotes de périphérique de la fonction virtuelle agissent sur le jeu de registres pour activer sa fonctionnalité et la fonction virtuelle apparaît sous la forme d'un véritable périphérique PCI. Après sa création, vous pouvez directement affecter une fonction virtuelle à un domaine d'E/S. Cette fonctionnalité permet à la fonction virtuelle de partager le périphérique physique et de réaliser des E/S sans surcharge de la CPU et du logiciel hyperviseur.

Il est recommandé d'utiliser la fonction SR-IOV dans votre environnement pour récolter les avantages suivants :

- **Performances accrues et latence réduite** : accès direct au matériel à partir d'un environnement de machines virtuelles
- **Réduction des coûts** : réduction des dépenses d'investissement et d'exploitation, notamment :
  - Economies d'énergie
  - Nombre réduit d'adaptateurs
  - Câblage moindre
  - Moins de ports de commutateur

L'implémentation SR-IOV d'Oracle VM Server for SPARC inclut à la fois des méthodes de configuration statiques et dynamiques. Pour plus d'informations, reportez-vous aux sections"SR-IOV statique" à la page 89 et "SR-IOV dynamique" à la page 90.

La fonction SR-IOV d'Oracle VM Server for SPARC permet de réaliser les opérations suivantes :

- Création d'une fonction virtuelle sur une fonction physique spécifiée
- Destruction d'une fonction virtuelle spécifiée sur une fonction physique
- Assignation d'une fonction virtuelle à un domaine
- Suppression d'une fonction virtuelle d'un domaine

Pour créer et supprimer des fonctions virtuelles dans les périphériques des fonctions virtuelles SR-IOV, vous devez commencer par activer la virtualisation d'E/S sur le bus PCIe. Vous pouvez utiliser la commande `ldm set-io` ou `ldm add-io` pour paramétrer la propriété `iov` sur `on`. Vous pouvez également utiliser la commande `ldm add-domain` ou `ldm set-domain` pour paramétrer la propriété `rc-add-policy` sur `iov`. Reportez-vous à la page de manuel `ldm`(1M).

Remarque - Sur les serveurs de série SPARC M7, SPARC T7 et les Serveurs Fujitsu M10, les bus PCIe sont activés par défaut pour la virtualisation d'E/S.

L'affectation d'une fonction virtuelle SR-IOV à un domaine crée une dépendance implicite sur le domaine fournissant le service de la fonction physique SR-IOV. Vous pouvez afficher ces dépendances ou domaines dépendant de la fonction physique SR-IOV à l'aide de la commande `ldm list-dependencies`. Voir "Liste des dépendances d'E/S de domaine" à la page 430.

Configuration matérielle et logicielle requise pour SR-IOV

Les fonctionnalités SR-IOV PCIe dynamique et statique sont prises en charge sur les serveurs SPARC T4, SPARC T5, les serveurs de série SPARC T7, les serveurs SPARC M5, SPARC M6 et les serveurs de série SPARC M7. La fonction dynamique est prise en charge par les Plates-formes Fujitsu M10 pour les périphériques Ethernet uniquement, alors que les autres types de périphériques exigent que vous utilisiez la méthode statique. La plate-forme SPARC T3 prend uniquement en charge la fonctionnalité SR-IOV PCIe statique.

- **Configuration matérielle.**

  Reportez-vous à la documentation matérielle de votre plate-forme pour vérifier les cartes que vous pouvez utiliser. Pour consulter une liste plus récente des cartes PCIe prises en charge, consultez la page `https://support.oracle.com/CSP/main/article?cmd=show&type=NOT&doctype=REFERENCE&id=1325454.1`.

  - **SR-IOV Ethernet.** Pour utiliser la fonctionnalité SR-IOV, vous pouvez utiliser n'importe quel périphérique SR-IOV PCIe intégré ainsi que des cartes d'extension SR-IOV PCIe. Tous les périphériques SR-IOV intégrés à une plate-forme donnée sont pris en charge, sauf mention contraire dans la documentation de la plate-forme.

- **SR-IOV InfiniBand.** Les périphériques InfiniBand sont pris en charge sur les serveurs SPARC T4, SPARC T5, les serveurs de série SPARC T7, les serveurs SPARC M5, SPARC M6, les serveurs de série SPARC M7 et les Serveurs Fujitsu M10.
- **SR-IOV Fibre Channel.** Les périphériques Fibre Channel sont pris en charge sur les serveurs SPARC T4, SPARC T5, les serveurs de série SPARC T7, les serveurs SPARC M5, SPARC M6, les serveurs de série SPARC M7 et les Serveurs Fujitsu M10.

  Pour obtenir la liste actuelle des périphériques pris en charge sur les Plates-formes Fujitsu M10, reportez-vous au *Fujitsu M10/SPARC M10 Systems PCI Card Installation Guide* des notes de produit de votre modèle à l'adresse suivante : `http://www.fujitsu.com/global/services/computing/server/sparc/downloads/manual/`

- **Configuration de microprogramme requise.**
  - **SR-IOV Ethernet.** Pour utiliser la fonctionnalité SR-IOV dynamique, le serveur SPARC T4 doit au moins exécuter la version 8.4.0.a du microprogramme du système. Les serveurs SPARC T5, SPARC M5 et SPARC M6 doivent au moins exécuter la version 9.1.0.a du microprogramme du système. Les serveurs de série SPARC T7 et SPARC M7 doivent au moins exécuter la version 9.4.3 du microprogramme du système, tandis que les Serveurs Fujitsu M10 doivent au moins exécuter la version XCP2210 du microprogramme du système. Le serveur SPARC T3 prend uniquement en charge la fonctionnalité SR-IOV statique.

    Pour utiliser la fonction SR-IOV, les périphériques SR-IOV PCIe doivent exécuter au minimum la version 3.01 du microprogramme du périphérique. Effectuez les étapes suivantes pour mettre à jour le microprogramme pour les adaptateurs réseau Sun Dual 10-Gigabit Ethernet SFP+ PCIe 2.0

    1. Déterminez si la mise à niveau de la version FCode est nécessaire sur le périphérique.

       Exécutez ces commandes à partir de l'invite `ok` :

       ```
       {0} ok cd path-to-device
       {0} ok .properties
       ```

       La valeur `version` dans la sortie doit être au choix :

`PL`	Sun Dual 10GbE SFP+ PCIe 2.0 LP FCode 3.01 4/2/2012
`PEM`	Sun Dual 10GbE SFP+ PCIe 2.0 EM FCode 3.01 4/2/2012
`FEM`	Sun Dual 10GbE SFP+ PCIe 2.0 FEM FCode 3.01 4/2/2012

    2. Téléchargez le patch13932765 sur le site Web My Oracle Support (`https://support.oracle.com/CSP/ui/flash.html#tab=PatchHomePage(page=PatchHomePage&id=h0wvdxy6())`).
    3. Installez le patch.

Le package du patch inclut un document décrivant la manière d'utiliser l'outil afin de mettre à niveau.

- **SR-IOV InfiniBand.** Pour utiliser cette fonctionnalité, le système doit au moins exécuter la version suivante du microprogramme système :
  - **Serveurs SPARC T4 –** 8.4
  - **Serveurs SPARC T5 –** 9.1.0.*x*
  - **Serveurs de série SPARC T7 –** 9.4.3
  - **Serveurs SPARC M5 et SPARC M6 –** 9.1.0.*x*
  - **Serveurs de série SPARC M7 –** 9.4.3
  - **Serveur Fujitsu M10 –** XCP2210

  Pour prendre en charge l'adaptateur de canal hôte Dual 40-Gigabit (4x) InfiniBand HBA M2 en tant que périphérique SR-IOV InfiniBand, le module de carte ou l'Express Module doit au moins exécuter la version 2.11.2010 du microprogramme. Vous pouvez obtenir cette version du microprogramme en installant les patchs suivants :
  - **Profil bas (X4242A) –** ID de patch 16340059
  - **Express Module (X4243A) –** ID de patch 16340042

  Utilisez la commande `fwflash` Oracle Solaris 11.1 pour répertorier et mettre à jour les microprogrammes dans le domaine `primary`. Pour répertorier la version actuelle du microprogramme, utilisez la commande IB `fwflash -lc IB`. Pour mettre à jour le microprogramme, utilisez la commande `fwflash -f` *firmware-file* `-d` *device*. Voir la page de manuel `fwflash`(1M).

  Pour utiliser SR-IOV InfiniBand, assurez-vous que les commutateurs InfiniBand sont équipés au moins de la version 2.1.2 du microprogramme. Vous pouvez obtenir cette version du microprogramme en installant les patchs suivants :
  - **Commutateur Sun Datacenter InfiniBand Switch 36 (X2821A-Z) –** ID de patch 16221424
  - **Commutateur de passerelle Sun Network QDR GatewaySwitch (X2826A-Z) –** ID de patch 16221538

  Pour plus d'informations sur la procédure de mise à jour du microprogramme, reportez-vous à la documentation de votre commutateur InfiniBand.
- **SR-IOV Fibre Channel.** Pour utiliser cette fonctionnalité, le système doit au moins exécuter la version suivante du microprogramme système :
  - **Serveur SPARC T4 –** 8.4.2.c
  - **Serveur SPARC T5 –** 9.1.2.d
  - **Serveur de série SPARC T7 –** 9.4.3
  - **Serveur SPARC M5 –** 9.1.2.d
  - **Serveur SPARC M6 –** 9.1.2.d

- **Serveur de série SPARC M7 –** 9.4.3
- **Serveur Fujitsu M10 –** XCP2210

Le microprogramme de Sun Storage 16 Gb Fibre Channel Universal HBA, Emulex doit être au moins à la révision 1.1.60.1 pour permettre la fonctionnalité SR-IOV Fibre Channel. Le microprogramme est fourni avec des instructions d'installation.

Remarque - Si vous prévoyez d'utiliser la fonction SR-IOV, vous devez mettre à jour le microprogramme pour correspondre au niveau minimal requis.

- **Configuration logicielle requise.**
  - **SR-IOV Ethernet.** Pour utiliser la fonctionnalité SR-IOV, tous les domaines doivent au moins exécuter le SE Oracle Solaris 11.1 SRU 10.
  - **SR-IOV InfiniBand.** Les domaines suivants doivent exécuter le SE Oracle Solaris pris en charge :
    - Le domaine `primary` ou un domaine root non `primary` doivent exécuter au moins le SE Oracle Solaris 11.1 SRU 10.
    - Les domaines d'E/S doivent au moins exécuter le SE Oracle Solaris 11.1 SRU 10.
    - Mettez à jour le fichier `/etc/system` sur un tout domaine root détenant une fonction physique SR-IOV InfiniBand à partir duquel vous envisagez de configurer des fonctions virtuelles.

      ```
      set ldc:ldc_maptable_entries = 0x20000
      ```

      Pour plus d'informations sur la création ou la mise à jour des valeurs de propriété `/etc/system`, reportez-vous à la section "Mise à jour des valeurs de propriété dans le fichier `/etc/system`" à la page 409.

      Mettez à jour le fichier `/etc/system` sur le domaine d'E/S auquel vous ajoutez une fonction virtuelle.

      ```
      set rdsv3:rdsv3_fmr_pool_size = 16384
      ```
  - **SR-IOV Fibre Channel.** Pour utiliser la fonctionnalité SR-IOV, tous les domaines doivent au moins exécuter le SE Oracle Solaris 11.1 SRU 17.

Consultez les sections suivantes pour plus d'informations sur la configuration logicielle requise pour les fonctions SR-IOV statiques et dynamiques :

- "Configuration logicielle requise pour la méthode SR-IOV statique" à la page 90
- "Configuration logicielle requise pour la fonction SR-IOV dynamique" à la page 91

Consultez les sections suivantes pour plus d'informations sur la configuration matérielle requise pour SR-IOV en fonction de la classe :

- "Configuration matérielle requise pour SR-IOV Ethernet" à la page 96
- "Configuration matérielle requise pour SR-IOV InfiniBand" à la page 117
- "Configuration matérielle requise pour SR-IOV Fibre Channel" à la page 131

Restrictions actuelles de la fonction SR-IOV

La fonction SR-IOV présente les restrictions suivantes :

- Un domaine d'E/S ne peut pas démarrer si le domaine root qui lui est associé n'est pas en cours d'exécution.
- La migration est désactivée pour tout domaine auquel des fonctions physiques SR-IOV ou des fonctions virtuelles SR-IOV sont affectées.
- Vous pouvez uniquement détruire la dernière fonction virtuelle créée pour une fonction physique. Par conséquent, si vous créez trois fonctions virtuelles, la première fonction virtuelle que vous pouvez détruire doit être la troisième.
- Si une carte SR-IOV est assignée à un domaine à l'aide de la fonction d'E/S directes (DIO), la fonction SR-IOV n'est pas activée pour cette carte.
- Les périphériques d'extrémité PCIe et les fonctions virtuelles SR-IOV d'un bus PCIe donné peuvent être affectés à 15 domaines au maximum sur les serveurs SPARC de série T et de série M. Sur les serveurs de série SPARC T7 et SPARC M7, vous pouvez associer des périphériques d'extrémité PCIe et des fonctions virtuelles SR-IOV d'un bus PCIe donné à 31 domaines au maximum. Sur un Serveur Fujitsu M10, vous pouvez associer des périphériques d'extrémité PCIe et des fonctions virtuelles SR-IOV d'un bus PCIe donné à un maximum de 24 domaines. Les ressources PCIe telles que les vecteurs d'interruption de chaque bus PCIe sont réparties entre le domaine root et les domaines d'E/S. De ce fait, le nombre de périphériques pouvant être affectés à un domaine d'E/S particulier est également limité. Soyez donc attentif à ne pas assigner un grand nombre de fonctions virtuelles à un même domaine d'E/S. Il n'existe pas de limitation d'interruption pour les serveurs de série SPARC T7 et SPARC M7. Pour une description des problèmes liés à SR-IOV, reportez-vous au manuel *Notes de version d'Oracle VM Server for SPARC 3.4*.
- Le domaine root est le propriétaire du bus PCIe et est responsable de l'initialisation et de la gestion de ce bus. Le domaine root doit être actif et exécuter une version du SE SE Oracle Solaris prenant en charge la fonction SR-IOV. L'arrêt, l'interruption et la réinitialisation du domaine root interrompent l'accès au bus PCIe. Lorsque le bus PCIe est indisponible, les périphériques PCIe sur ce bus sont affectés et risquent de devenir indisponibles.

  Le comportement de domaines d'E/S comportant des fonctions virtuelles SR-IOV PCIe est imprévisible si le domaine root est réinitialisé alors que ces domaines d'E/S sont en cours d'exécution. Par exemple, les domaines d'E/S avec des périphériques d'extrémité PCIe

peuvent paniquer au cours ou après une réinitialisation. Dans cas, il faut arrêter et démarrer manuellement chaque domaine après la réinitialisation du domaine root.

Si le domaine d'E/S est résilient, il peut continuer à fonctionner même si le domaine root qui est le propriétaire du bus PCIe devient indisponible. Voir "Résilience de domaine d'E/S" à la page 145.

- Les systèmes SPARC, jusqu'aux plates-formes SPARC T5 et SPARC M6 incluses, indiquent un nombre limité d'interruptions afin qu'Oracle Solaris limite le nombre d'interruptions utilisables par chaque périphérique. La limite par défaut doit correspondre aux besoins d'une configuration système type, sachant que la valeur devra peut-être être ajustée pour certaines configurations système. Pour plus d'informations, reportez-vous à la section "Ajustement de la limite d'interruption" à la page 427.

SR-IOV statique

La méthode SR-IOV statique nécessite que le domaine root soit en mode reconfiguration retardée ou que le domaine d'E/S soit arrêté pendant l'exécution des opérations SR-IOV. Après avoir effectué les étapes de configuration sur le domaine root, vous devez le réinitialiser. Vous devez utiliser cette méthode lorsque le microprogramme Oracle VM Server for SPARC 3.1 n'est pas installé dans le système ou lorsque la version du SE installée dans le domaine concerné ne prend pas en charge le SR-IOV dynamique.

Pour créer ou détruire une fonction virtuelle SR-IOV, vous devez lancer une reconfiguration retardée sur le domaine root. Vous pouvez ensuite exécuter une ou plusieurs commandes`ldm create-vf` et `ldm destroy-vf` pour configurer les fonctions virtuelles. Pour finir, vous devez réinitialiser le domaine root. Les commandes suivantes indiquent comment créer une fonction virtuelle sur un domaine root différent de `primary` :

```
primary# ldm start-reconf root-domain-name
primary# ldm create-vf pf-name
primary# ldm stop-domain -r root-domain-name

primary# shutdown -i6 -g0 -y
```

Pour ajouter une fonction virtuelle de façon statique à un domaine invité ou pour l'en supprimer, vous devez d'abord arrêter le domaine invité. Exécutez ensuite les commandes `ldm add-io` et `ldm remove-io` pour configurer les fonctions virtuelles. Une fois les modifications effectuées, démarrez le domaine. Les commandes suivantes indiquent comment assigner une fonction virtuelle de cette manière :

```
primary# ldm stop guest-domain
primary# ldm add-io vf-name guest-domain
primary# ldm start guest-domain
```

Au lieu d'un domaine invité, vous pouvez également ajouter une fonction virtuelle à un domaine root ou l'en supprimer. Pour ajouter ou supprimer une fonction virtuelle SR-IOV d'un domaine

root, lancez d'abord une reconfiguration retardée sur le domaine root. Vous pouvez ensuite exécuter une ou plusieurs des commandes `ldm add-io` et `ldm remove-io`. Pour finir, vous devez réinitialiser le domaine root.

Planifiez la configuration des fonctions virtuelles à l'avance afin de réduire l'indisponibilité du domaine.

Remarque - Les périphériques SR-IOV InfiniBand sont uniquement pris en charge avec la méthode SR-IOV statique.

Configuration logicielle requise pour la méthode SR-IOV statique

Pour plus d'informations sur la configuration SR-IOV matérielle et logicielle requise, voir `https://support.oracle.com/CSP/main/article?cmd=show&type=NOT&doctype=REFERENCE&id=1325454.1`.

Vous pouvez utiliser la commande `ldm set-io` ou `ldm add-io` pour paramétrer la propriété `iov` sur `on`. Vous pouvez également utiliser la commande `ldm add-domain` ou `ldm set-domain` pour paramétrer la propriété `rc-add-policy` sur `iov`. Reportez-vous à la page de manuel `ldm`(1M).

La réinitialisation du domaine root affecte la fonction SR-IOV. En conséquence, planifiez correctement vos modifications de configuration d'E/S directes pour optimiser les modifications relatives à la fonction SR-IOV sur le domaine root et pour réduire les réinitialisations du domaine root.

SR-IOV dynamique

La fonction SR-IOV dynamique dispense des conditions requises pour la fonction SR-IOV statique suivantes :

- **Domaine root.** Lancement d'une reconfiguration retardée sur le domaine root, création ou suppression d'une fonction virtuelle et réinitialisation du domaine root
- **Domaine d'E/S.** Arrêt du domaine d'E/S, ajout ou suppression d'une fonction virtuelle et démarrage du domaine d'E/S

Grâce à la fonction SR-IOV dynamique, vous pouvez créer ou détruire une fonction virtuelle de façon dynamique sans avoir à lancer une reconfiguration retardée sur le domaine root.

Vous pouvez également ajouter une fonction virtuelle à un domaine d'E/S ou l'en supprimer sans avoir à arrêter le domaine. Logical Domains Manager communique avec l'agent Logical Domains et la structure de virtualisation d'E/S d'Oracle Solaris pour appliquer ces modifications de façon dynamique.

Configuration logicielle requise pour la fonction SR-IOV dynamique

Pour plus d'informations sur les versions logicielles et matérielles SR-IOV PCIe requises, reportez-vous à la section "Configuration matérielle et logicielle requise pour SR-IOV" à la page 84.

Remarque - Si votre système n'est pas équipé des logiciels et microprogrammes requis pour la fonction SR-IOV dynamique, utilisez la méthode SR-IOV statique pour effectuer les tâches liées à SR-IOV. Voir la section "SR-IOV statique" à la page 89.

Configuration requise pour la fonction SR-IOV dynamique

Si vous voulez créer ou détruire une fonction virtuelle de façon dynamique, assurez-vous que les conditions suivantes sont remplies :

- La virtualisation d'E/S a été activée pour un bus PCIe avant que vous ne commenciez à configurer des fonctions virtuelles.
- Le système d'exploitation qui s'exécute sur le domaine root et les domaines d'E/S doit être au moins le SE Oracle Solaris 11.1 SRU 10.
- Le périphérique de fonction physique n'est pas configuré dans le SE ou fait partie d'une configuration de multipathing. Par exemple, vous pouvez démonter un périphérique SR-IOV Ethernet ou l'ajouter à une configuration IPMP (IP Network Multipathing) ou un groupement pour pouvoir créer ou détruire une fonction virtuelle.

  Une opération de création ou de destruction d'une fonction virtuelle exige que le pilote de périphérique de la fonction physique bascule entre les états en ligne et hors ligne. Une configuration multipathing permet au pilote de périphérique de basculer entre ces états.
- Lorsque vous supprimez une fonction virtuelle d'un domaine d'E/S, la fonction virtuelle n'est pas utilisée ou fait partie d'une configuration multipathing. Par exemple, vous pouvez soit démonter une fonction virtuelle SR-IOV Ethernet soit ne pas l'utiliser dans une configuration IPMP.

Remarque - Vous ne pouvez pas utiliser le groupement pour les fonctions virtuelles SR-IOV Ethernet car l'implémentation actuelle de la fonctionnalité multipathing ne prend pas en charge les fonctions virtuelles.

Destruction de l'ensemble des fonctions virtuelles et renvoi des emplacements vers le domaine root n'entraînant aucune restauration des ressources de complexe root

Remarque - Cette section s'applique aux serveurs antérieurs ou identiques aux séries SPARC M6 et SPARC T5.

Les ressources situées sur le complexe root ne sont pas restaurées après la destruction de l'ensemble des fonctions virtuelles et le renvoi des emplacements vers le domaine root.

Récupération : renvoyez l'ensemble des ressources d'E/S virtuelles associées au complexe root à leur domaine root.

Placez d'abord le domaine de contrôle en mode de reconfiguration retardée.

```
primary# ldm start-reconf primary
```

Renvoyez l'ensemble des emplacements PCIe enfants au domaine root propriétaire du bus `pci_0`. Supprimez ensuite l'ensemble des fonctions virtuelles enfants sur le bus `pci_0` et détruisez-les.

Enfin, définissez `iov=off` pour le bus `pci_0` et réinitialisez le domaine root.

```
primary# ldm set-io iov=off pci_0
primary# shutdown -y -g 10
```

Solution de contournement : définissez l'option `iov` sur `off` pour le bus PCIe spécifique.

```
primary# ldm start-reconf primary
primary# ldm set-io iov=off pci_0
```

Activation de la virtualisation des E/S

Avant de pouvoir configurer des fonctions virtuelles SR-IOV, vous devez activer la virtualisation d'E/S pour le bus PCIe et lancer une reconfiguration retardée sur le domaine root. Réinitialisez le domaine pour appliquer la modification.

Remarque - Sur les serveurs de série SPARC M7, SPARC T7 et les Serveurs Fujitsu M10, les bus PCIe sont activés par défaut pour la virtualisation d'E/S.

▼ Procédure d'activation de la virtualisation d'E/S pour un bus PCIe

Cette procédure ne doit être effectuée qu'une seule fois par complexe root. Le complexe root doit être en cours d'exécution dans le cadre de la même configuration de SP.

1. **Lancez une reconfiguration retardée sur le domaine root.**

   ```
   primary# ldm start-reconf root-domain-name
   ```

2. **Activez les opérations de virtualisation d'E/S pour un bus PCIe.**

   Effectuez uniquement cette étape si la virtualisation d'E/S n'est pas déjà activée pour le bus qui a la fonction physique.

   Exécutez l'une des commandes suivantes :

   - **Activation de la virtualisation d'E/S si le bus PCIe spécifié est déjà assigné à un domaine root.**

     ```
     primary# ldm set-io iov=on bus
     ```

   - **Activation de la virtualisation d'E/S pendant l'ajout d'un bus PCIe à un domaine root.**

     ```
     primary# ldm add-io iov=on bus
     ```

3. **Réinitialisez le domaine root.**

   Exécutez l'une des commandes suivantes :

   - **Réinitialisation du domaine root s'il est différent de `primary`.**

     ```
     primary# ldm stop-domain -r root-domain
     ```

   - **Réinitialisation du domaine root `primary`.**

     ```
     primary# shutdown -i6 -g0 -y
     ```

Planification de l'utilisation de fonctions virtuelles SR-IOV PCIe

Planifiez à l'avance la manière dont vous souhaitez utiliser les fonctions virtuelles dans votre configuration. Déterminez quelles fonctions virtuelles des périphériques SR-IOV vont répondre à vos besoins de configuration actuels et à venir.

Si vous n'avez pas encore activé la virtualisation d'E/S, qui requiert l'utilisation de la méthode statique, associez cette étape aux étapes de création des fonctions virtuelles. En combinant ces étapes, vous ne devrez réinitialiser qu'une seule fois le domaine root.

Même lorsque la fonction SR-IOV dynamique est disponible, il est recommandé de créer toutes les fonctions virtuelles en une seule opération car vous risquez de ne pas pouvoir les créer de façon dynamique après leur assignation aux domaines d'E/S.

Dans le cas de la méthode SR-IOV statique, une bonne planification vous permet d'éviter des réinitialisations multiples du domaine root, lesquelles peuvent avoir une incidence négative sur les domaines d'E/S.

Pour plus d'informations sur les domaines d'E/S, reportez-vous à la section "Lignes directrices pour la création d'un domaine d'E/S" à la page 68.

Procédez aux étapes générales suivantes pour planifier et réaliser la configuration et l'affectation des fonctions virtuelles SR-IOV :

1. Faites l'inventaire des fonctions physiques SR-IOV PCIe disponibles sur votre système et déterminez celles qui répondent le mieux à vos besoins.

   Utilisez les commandes suivantes pour obtenir les informations requises :

`ldm list-io`	Identifie les périphériques de fonction physique SR-IOV disponibles.
`prtdiag -v`	Identifie les cartes SR-IOV PCIe et les périphériques intégrés disponibles.
`ldm list-io -l` *pf-name*	Identifie des informations supplémentaires à propos d'une fonction physique donnée, par exemple le nombre maximal de fonctions virtuelles prises en charge par le périphérique.
`ldm list-io -d` *pf-name*	Identifie les propriétés spécifiques au périphérique prises en charge par le périphérique. Voir la section "Rubriques SR-IOV avancées : SR-IOV Ethernet" à la page 110.

2. Activez les opérations de virtualisation d'E/S pour un bus PCIe.

Voir la section "Procédure d'activation de la virtualisation d'E/S pour un bus PCIe" à la page 93.

3. Créez le nombre de fonctions virtuelles requises sur la fonction physique SR-IOV spécifiée.

   Utilisez la commande suivante pour créer les fonctions virtuelles de la fonction physique :

   ```
   primary# ldm create-vf -n max pf-name
   ```

   Pour plus d'informations, reportez-vous aux sections "Procédure de création d'une fonction virtuelle SR-IOV Ethernet" à la page 98, "Procédure de création d'une fonction virtuelle InfiniBand" à la page 117 et "Procédure de création d'une fonction virtuelle SR-IOV Fibre Channel" à la page 134.

4. Utilisez la commande `ldm add-config` pour enregistrer la configuration sur le processeur de service.

   Pour plus d'informations, reportez-vous aux sections "Procédure d'ajout d'une fonction virtuelle SR-IOV Ethernet à un domaine d'E/S" à la page 107, "Procédure d'ajout d'une fonction virtuelle InfiniBand à un domaine d'E/S" à la page 122 et "Procédure d'ajout d'une fonction virtuelle SR-IOV Fibre Channel à un domaine d'E/S" à la page 142.

Utilisation des fonctions virtuelles SR-IOV Ethernet

Vous pouvez utiliser les méthodes SR-IOV statiques et dynamiques pour gérer les périphériques SR-IOV Ethernet.

Attention - Lors de l'utilisation de certains adaptateurs réseau Intel prenant en charge SR-IOV, une fonction virtuelle peut être la cible d'un comportement malveillant. Des trames inattendues générées par le logiciel peuvent ralentir le trafic entre l'hôte et le commutateur virtuel, ce qui risque d'avoir un impact négatif sur les performances.

Configurez tous les ports compatibles SR-IOV afin qu'ils utilisent le balisage VLAN pour supprimer les trames inattendues et potentiellement malveillantes.

- Pour configurer le balisage VLAN sur une fonction physique et les fonctions virtuelles qui lui sont associées, utilisez la commande suivante :

  `ldm create-vf [pvid=`*pvid*`] [vid=`*vid1*`,`*vid2*`,...>]` *net-pf-name*

- Pour configurer le balisage VLAN sur une fonction virtuelle existante, utilisez la commande suivante :

  `ldm set-io [pvid=[`*pvid*`]] [vid=[`*vid1*`,`*vid2*`,...]]` *net-vf-name*

Pour plus d'informations sur la création de l'interface VLAN dans le domaine d'E/S, reportez-vous à la section "Utilisation du balisage VLAN" du manuel *Guide d'administration d'Oracle VM Server for SPARC 3.4*.

Configuration matérielle requise pour SR-IOV Ethernet

Pour plus d'informations sur le matériel Ethernet SR-IOV PCIe requis, reportez-vous à la section "Configuration matérielle et logicielle requise pour SR-IOV" à la page 84.

Restrictions applicables à SR-IOV Ethernet

Dans cette version, la fonction Ethernet SR-IOV présente les restrictions suivantes :

- Vous pouvez activer les configurations VLAN de fonctions virtuelles en définissant la propriété `pvid` ou la propriété `vid`. Ces deux propriétés de fonctions virtuelles ne peuvent pas être définies simultanément.
- Vous ne pouvez pas utiliser une fonction virtuelle SR-IOV en tant que périphérique backend pour un commutateur virtuel.

Planification de l'utilisation des fonctions virtuelles SR-IOV Ethernet

Lorsque vous créez des fonctions virtuelles de façon dynamique, assurez-vous que les fonctions physiques utilisent la fonctionnalité multipathing ou qu'elles ne sont pas raccordées.

Si vous ne pouvez pas utiliser la fonctionnalité multipathing ou que vous devez raccorder la fonction physique, servez-vous de la méthode statique pour créer les fonctions virtuelles. Voir la section "SR-IOV statique" à la page 89.

Propriétés spécifiques aux périphériques et au réseau pour Ethernet

Utilisez la commande `ldm create-vf` pour définir les propriétés spécifiques au périphérique et au réseau d'une fonction virtuelle. La propriété `unicast-slots` est spécifique au périphérique. Les propriétés `mac-addr`, `alt-mac-addrs`, `mtu`, `pvid` et `vid` sont spécifiques au réseau.

Notez que dans le cadre d'une configuration retardée, les propriétés spécifiques au réseau `mac-addr`, `alt-mac-addrs` et `mtu` peuvent être uniquement modifiées lorsque la fonction virtuelle est assignée au domaine `primary`.

Les tentatives de modification de ces propriétés échouent lorsque la fonction virtuelle est assignée comme suit :

- Lorsque la fonction virtuelle est affectée à un domaine d'E/S actif : une demande de modification de propriété est rejetée car la modification doit être effectuée lorsque le domaine propriétaire est dans l'état inactif ou lié.
- Lorsque la fonction virtuelle est assignée à un domaine non `primary` et qu'une reconfiguration retardée est en cours : une demande de modification de propriété échoue et un message d'erreur s'affiche.

Les propriétés spécifiques au réseau `pvid` et `vid` peuvent être modifiées sans restrictions.

Création de fonctions Ethernet virtuelles

Cette section décrit la création et la destruction dynamiques de fonctions virtuelles. Si vous ne pouvez pas utiliser les méthodes dynamiques pour exécuter ces actions, lancez une

reconfiguration retardée sur le domaine root avant de créer ou de détruire des fonctions virtuelles.

▼ Procédure de création d'une fonction virtuelle SR-IOV Ethernet

Si vous ne pouvez pas utiliser cette méthode dynamique, utilisez plutôt la méthode statique. Voir la section "SR-IOV statique" à la page 89.

1. **Identifiez le périphérique de fonction physique.**

   ```
   primary# ldm list-io
   ```

   Notez que le nom de la fonction physique inclut les informations d'emplacement de la carte SR-IOV PCIe ou du périphérique intégré.

2. **Effectuez uniquement cette étape si la virtualisation d'E/S n'est pas déjà activée pour le bus qui a la fonction physique.**

   Effectuez uniquement cette étape si la virtualisation d'E/S n'est pas déjà activée pour le bus qui a la fonction physique.

   Voir la section "Procédure d'activation de la virtualisation d'E/S pour un bus PCIe" à la page 93.

3. **Vous pouvez créer une fonction virtuelle unique ou plusieurs fonctions virtuelles à partir d'une fonction Ethernet physique, de manière dynamique ou statique.**

   Après avoir créé une ou plusieurs fonctions virtuelles, vous pouvez les attribuer à un domaine invité.

   - **Méthode dynamique :**

     - **Pour créer plusieurs fonctions virtuelles à partir d'une fonction physique en une seule fois, utilisez la commande suivante :**

       ```
       primary# ldm create-vf -n number | max pf-name
       ```

       Utilisez la commande `ldm create-vf -n max` pour créer en une fois toutes les fonctions virtuelles de cette fonction physique.

Attention - Lorsque votre système utilise une carte Ethernet Intel 10 Gbits, vous pouvez optimiser les performances en ne créant pas plus de 31 fonctions virtuelles à partir de chaque fonction physique.

Vous pouvez spécifier des fonctions virtuelles soit par le biais de leur nom de chemin d'accès, soit par leur pseudonyme. Toutefois, la pratique recommandée est d'utiliser le pseudonyme.

- **Pour créer une fonction virtuelle à partir d'une fonction physique, utilisez la commande suivante :**

```
ldm create-vf [mac-addr=num] [alt-mac-addrs=[auto|num1,[auto|num2,...]]]
  [pvid=pvid] [vid=vid1,vid2,...] [mtu=size] [name=value...] pf-name
```

Remarque - Si elle n'est pas explicitement assignée, l'adresse MAC est automatiquement allouée aux périphériques réseau.

Utilisez cette commande pour créer une fonction virtuelle pour cette fonction physique. Vous pouvez également indiquer manuellement les valeurs de propriété Ethernet en fonction de la classe.

Remarque - Parfois, une fonction virtuelle nouvellement créée n'est pas disponible pour une utilisation immédiate, car le SE recherche toujours les périphériques IOV. Utilisez la commande `ldm list-io` pour déterminer si la fonction physique parent et ses fonctions virtuelles enfant ont la valeur `INV` dans la colonne Statut. Si cette valeur s'affiche, attendez que la sortie`ldm list-io` n'affiche plus la valeur `INV` dans la colonne Statut (environ 45 secondes) avant d'utiliser cette fonction physique ou l'une de ses fonctions enfant. Si ce statut persiste, cela signifie qu'il y a un problème avec le périphérique.

Le statut de l'appareil pourra être `INV` immédiatement après une réinitialisation du domaine root (y compris celle du domaine `primary`) ou immédiatement après avoir utilisé la commande `ldm create-vf` ou `ldm destroy-vf`.

- **Méthode statique :**

  a. **Lancez une reconfiguration retardée.**

  ```
  primary# ldm start-reconf root-domain-name
  ```

  b. **Vous pouvez créer une fonction virtuelle unique ou plusieurs fonctions virtuelles à partir d'une fonction Ethernet physique.**

  Utilisez les mêmes commandes affichées précédemment pour créer les fonctions virtuelles de manière dynamique.

  c. **Réinitialisez le domaine root.**

- **Procédure de réinitialisation du domaine root non-`primary` :**

  ```
  primary# ldm stop-domain -r root-domain
  ```

- **Procédure de réinitialisation du domaine root `primary` :**

  ```
  primary# shutdown -i6 -g0 -y
  ```

Exemple 5 Affichage des informations relatives à la fonction Ethernet physique

Cet exemple affiche des informations sur la fonction physique `/SYS/MB/NET0/IOVNET.PF0` :

- Cette fonction physique est issue d'un périphérique réseau `NET0` intégré.
- La chaîne `IOVNET` indique que la fonction physique est un périphérique réseau SR-IOV.

```
primary# ldm list-io
NAME                                    TYPE   BUS      DOMAIN   STATUS
----                                    ----   ---      ------   ------
niu_0                                   NIU    niu_0    primary
niu_1                                   NIU    niu_1    primary
pci_0                                   BUS    pci_0    primary
pci_1                                   BUS    pci_1    primary
/SYS/MB/PCIE0                           PCIE   pci_0    primary  OCC
/SYS/MB/PCIE2                           PCIE   pci_0    primary  OCC
/SYS/MB/PCIE4                           PCIE   pci_0    primary  OCC
/SYS/MB/PCIE6                           PCIE   pci_0    primary  EMP
/SYS/MB/PCIE8                           PCIE   pci_0    primary  EMP
/SYS/MB/SASHBA                          PCIE   pci_0    primary  OCC
/SYS/MB/NET0                            PCIE   pci_0    primary  OCC
/SYS/MB/PCIE1                           PCIE   pci_1    primary  OCC
/SYS/MB/PCIE3                           PCIE   pci_1    primary  OCC
/SYS/MB/PCIE5                           PCIE   pci_1    primary  OCC
/SYS/MB/PCIE7                           PCIE   pci_1    primary  EMP
/SYS/MB/PCIE9                           PCIE   pci_1    primary  EMP
/SYS/MB/NET2                            PCIE   pci_1    primary  OCC
/SYS/MB/NET0/IOVNET.PF0                 PF     pci_0    primary
/SYS/MB/NET0/IOVNET.PF1                 PF     pci_0    primary
/SYS/MB/PCIE5/IOVNET.PF0                PF     pci_1    primary
/SYS/MB/PCIE5/IOVNET.PF1                PF     pci_1    primary
/SYS/MB/NET2/IOVNET.PF0                 PF     pci_1    primary
/SYS/MB/NET2/IOVNET.PF1                 PF     pci_1    primary
```

La commande suivante fournit plus d'informations sur la fonction physique spécifiée. La valeur `maxvfs` indique le nombre maximal de fonctions virtuelles pris en charge par le périphérique.

```
primary# ldm list-io -l /SYS/MB/NET0/IOVNET.PF0
NAME                                    TYPE   BUS      DOMAIN   STATUS
----                                    ----   ---      ------   ------
/SYS/MB/NET0/IOVNET.PF0                 PF     pci_0    primary
[pci@400/pci@1/pci@0/pci@4/network@0]
    maxvfs = 7
```

Exemple 6 Création dynamique d'une fonction Ethernet virtuelle sans paramétrage des propriétés facultatives

Cet exemple crée une fonction virtuelle de façon dynamique sans définir de propriété facultative. Dans ce cas, l'adresse MAC des fonctions virtuelles de classe réseau est automatiquement attribuée.

Assurez-vous que la virtualisation d'E/S est activée sur le bus PCIe `pci_0`. Voir la section "Procédure d'activation de la virtualisation d'E/S pour un bus PCIe" à la page 93.

Vous pouvez ensuite utiliser la commande `ldm create-vf` pour créer la fonction virtuelle à partir de la fonction physique `/SYS/MB/NET0/IOVNET.PF0`.

```
primary# ldm create-vf /SYS/MB/NET0/IOVNET.PF0
Created new vf: /SYS/MB/NET0/IOVNET.PF0.VF0
```

Exemple 7 Création dynamique d'une fonction Ethernet virtuelle avec paramétrage des propriétés

Cet exemple crée une fonction virtuelle de façon dynamique tout en définissant la propriété `mac-addr` sur `00:14:2f:f9:14:c0` et la propriété `vid` sur les ID de VLAN `2` et `3`.

```
primary# ldm create-vf mac-addr=00:14:2f:f9:14:c0 vid=2,3 /SYS/MB/NET0/IOVNET.PF0
```

Exemple 8 Création dynamique d'une fonction Ethernet virtuelle possédant deux adresses MAC

Cet exemple crée de façon dynamique une fonction virtuelle possédant deux adresses MAC. Une adresse MAC est automatiquement attribuée, tandis que l'autre est explicitement spécifiée comme `00:14:2f:f9:14:c2`.

```
primary# ldm create-vf alt-mac-addrs=auto,00:14:2f:f9:14:c2 /SYS/MB/NET0/IOVNET.PF0
```

Exemple 9 Création statique d'une fonction virtuelle sans paramétrage des propriétés facultatives

Cet exemple crée une fonction virtuelle de façon statique sans définir de propriété facultative. Dans ce cas, l'adresse MAC des fonctions virtuelles de classe réseau est automatiquement attribuée.

Commencez par démarrer une reconfiguration retardée sur le domaine `primary` et activez la virtualisation d'E/S sur le bus PCIe `pci_0`. Le bus `pci_0` ayant déjà été assigné au domaine root `primary`, vous devez utiliser la commande `ldm set-io` pour activer la virtualisation d'E/S.

```
primary# ldm start-reconf primary
Initiating a delayed reconfiguration operation on the primary domain.
All configuration changes for other domains are disabled until the primary
domain reboots, at which time the new configuration for the primary domain
will also take effect.
```

```
primary# ldm set-io iov=on pci_0
```

Vous pouvez ensuite utiliser la commande `ldm create-vf` pour créer la fonction virtuelle à partir de la fonction physique `/SYS/MB/NET0/IOVNET.PF0`.

```
primary# ldm create-vf /SYS/MB/NET0/IOVNET.PF0

------------------------------------------------------------------------------
Notice: The primary domain is in the process of a delayed reconfiguration.
Any changes made to the primary domain will only take effect after it reboots.
------------------------------------------------------------------------------

Created new vf: /SYS/MB/NET0/IOVNET.PF0.VF0
```

Pour finir, réinitialisez le domaine root `primary` pour appliquer les modifications.

```
primary# shutdown -i6 -g0 -y
```

Exemple 10 Création de plusieurs fonctions virtuelles SR-IOV Ethernet

La commande suivant indique la procédure à suivre pour créer quatre fonctions virtuelles à partir de la fonction physique `/SYS/MB/NET2/IOVNET.PF1` :

```
primary# ldm create-vf -n 31 /SYS/MB/NET2/IOVNET.PF1
Created new vf: /SYS/MB/NET2/IOVNET.PF1.VF0
Created new vf: /SYS/MB/NET2/IOVNET.PF1.VF1
Created new vf: /SYS/MB/NET2/IOVNET.PF1.VF2
...
Created new vf: /SYS/MB/NET2/IOVNET.PF1.VF30
```

Notez que la commande `ldm create-vf -n` crée plusieurs fonctions virtuelles définies avec des valeurs de propriété par défaut, le cas échéant. Vous pouvez indiquer plus tard les valeurs de propriété qui ne sont pas définies par défaut à l'aide de la commande `ldm set-io`.

Destruction de fonctions Ethernet virtuelles

Vous pouvez détruire une fonction virtuelle si elle n'est pas affectée à un domaine. Vous pouvez uniquement détruire une fonction virtuelle dans l'ordre inverse de la création en d'autres termes, seule la dernière fonction virtuelle créée peut être détruite. La configuration résultante est validée par le pilote de la fonction physique.

▼ Procédure de destruction d'une fonction virtuelle SR-IOV Ethernet

Si vous ne pouvez pas utiliser cette méthode dynamique, utilisez plutôt la méthode statique. Voir la section "SR-IOV statique" à la page 89.

1. **Identifiez le périphérique de fonction physique.**

   ```
   primary# ldm list-io
   ```

2. **Vous pouvez détruire une fonction virtuelle unique ou plusieurs fonctions virtuelles de manière dynamique ou statique.**

   - **Méthode dynamique :**

     - **Pour détruire une ou plusieurs fonctions virtuelles à partir d'une fonction physique en une seule fois, utilisez la commande suivante :**

       ```
       primary# ldm destroy-vf -n number | max pf-name
       ```

       Utilisez la commande `ldm destroy-vf -n max` pour détruire en une fois toutes les fonctions virtuelles de cette fonction physique.

       Si vous indiquez un *nombre* comme argument pour l'option `-n`, le dernier *nombre* des fonctions virtuelles est détruit. Utilisez cette méthode car elle exécute l'opération en une seule transition d'état de pilote de périphérique de fonction physique.

       Vous pouvez spécifier des fonctions virtuelles soit par le biais de leur nom de chemin d'accès, soit par leur pseudonyme. Toutefois, la pratique recommandée est d'utiliser le pseudonyme.

     - **Procédure de destruction d'une fonction virtuelle spécifiée :**

       ```
       primary# ldm destroy-vf vf-name
       ```

     En raison de retards du périphérique matériel affecté au SE, la fonction physique affectée et toutes les autres fonctions virtuelles enfant risquent de ne plus être disponibles pour une utilisation immédiate. Utilisez la commande `ldm list-io` pour déterminer si la fonction physique parent et ses fonctions virtuelles enfant ont la valeur `INV` dans la colonne Statut. Si cette valeur s'affiche, attendez que la sortie`ldm list-io` n'affiche plus `INV` dans la colonne Statut (environ 45 secondes). A ce moment, vous pouvez utiliser cette fonction physique en toute sécurité, ou n'importe quelle fonction virtuelle enfant. Si ce statut persiste, cela signifie qu'il y a un problème avec le périphérique.

     Le statut de l'appareil pourra être `INV` immédiatement après une réinitialisation du domaine root (y compris celle du domaine `primary`) ou immédiatement après avoir utilisé la commande `ldm create-vf` ou `ldm destroy-vf`.

   - **Méthode statique :**

     a. **Lancez une reconfiguration retardée.**

        ```
        primary# ldm start-reconf root-domain-name
        ```

b. Vous pouvez détruire une fonction virtuelle unique ou plusieurs fonctions virtuelles.

- **Pour détruire toutes les fonctions virtuelles à partir d'une fonction physique spécifique en une seule fois, utilisez la commande suivante :**

  ```
  primary# ldm destroy-vf -n number | max pf-name
  ```

  Vous pouvez spécifier des fonctions virtuelles soit par le biais de leur nom de chemin d'accès, soit par leur pseudonyme. Toutefois, la pratique recommandée est d'utiliser le pseudonyme.

- **Procédure de destruction d'une fonction virtuelle spécifiée :**

  ```
  primary# ldm destroy-vf vf-name
  ```

c. Réinitialisez le domaine root.

- **Procédure de réinitialisation du domaine root non-`primary` :**

  ```
  primary# ldm stop-domain -r root-domain
  ```

- **Procédure de réinitialisation du domaine root `primary` :**

  ```
  primary# shutdown -i6 -g0 -y
  ```

Exemple 11 Destruction d'une fonction virtuelle Ethernet

Cet exemple illustre la destruction dynamique de la fonction virtuelle `/SYS/MB/NET0/IOVNET.PF0.VF0`.

```
primary# ldm destroy-vf /SYS/MB/NET0/IOVNET.PF0.VF0
```

L'exemple suivant illustre la destruction statique de la fonction virtuelle `/SYS/MB/NET0/IOVNET.PF0.VF0` :

```
primary# ldm start-reconf primary
Initiating a delayed reconfiguration operation on the primary domain.
All configuration changes for other domains are disabled until the primary
domain reboots, at which time the new configuration for the primary domain
will also take effect.

primary# ldm destroy-vf /SYS/MB/NET0/IOVNET.PF0.VF0
primary# shutdown -i6 -g0 -y
```

Exemple 12 Destruction de plusieurs fonctions virtuelles SR-IOV Ethernet

Cet exemple illustre les résultats de la destruction de toutes les fonctions virtuelles de la fonction physique `/SYS/MB/NET2/IOVNET.PF1`. La sortie `ldm list-io` indique que la fonction physique possède sept fonctions virtuelles. La commande `ldm destroy-vf` détruit toutes les fonctions virtuelles et la dernière sortie `ldm list-io` indique qu'il ne reste aucune fonction virtuelle.

```
primary# ldm list-io
...
/SYS/MB/NET2/IOVNET.PF1                    PF     pci_1
/SYS/MB/NET2/IOVNET.PF1.VF0                VF     pci_1
/SYS/MB/NET2/IOVNET.PF1.VF1                VF     pci_1
/SYS/MB/NET2/IOVNET.PF1.VF2                VF     pci_1
/SYS/MB/NET2/IOVNET.PF1.VF3                VF     pci_1
/SYS/MB/NET2/IOVNET.PF1.VF4                VF     pci_1
/SYS/MB/NET2/IOVNET.PF1.VF5                VF     pci_1
/SYS/MB/NET2/IOVNET.PF1.VF6                VF     pci_1
primary# ldm destroy-vf -n max /SYS/MB/NET2/IOVNET.PF1
primary# ldm list-io
...
/SYS/MB/NET2/IOVNET.PF1                    PF     pci_1    ldg1
```

Modification des fonctions virtuelles SR-IOV Ethernet

La commande `ldm set-io` *vf-name* modifie la configuration en cours d'une fonction virtuelle en modifiant les valeurs des propriétés ou en définissant de nouvelles propriétés. Cette commande permet de modifier à la fois les propriétés propres au réseau et les propriétés spécifiques au périphérique. Pour plus d'informations sur les propriétés spécifiques aux périphériques, reportez-vous à la section "Rubriques SR-IOV avancées : SR-IOV Ethernet" à la page 110.

Si vous ne pouvez pas utiliser cette méthode dynamique, utilisez à sa place la méthode statique. Voir la section "SR-IOV statique" à la page 89.

La commande `ldm set-io` permet de modifier les propriétés suivantes :

- `mac-addr`, `alt-mac-addrs` et `mtu`

  Pour modifier ces propriétés de fonction virtuelle, arrêtez le domaine propriétaire de la fonction virtuelle, utilisez la commande `ldm set-io` pour modifier les valeurs de propriétés et démarrez le domaine.

- `pvid` et `vid`

  Vous pouvez modifier ces propriétés de façon dynamique alors que les fonctions virtuelles sont assignées à un domaine. Notez qu'une telle modification peut entraîner un changement du trafic réseau d'une fonction virtuelle active ; la définition de la propriété `pvid` permet un

VLAN transparent. La configuration de la propriété `vid` de manière à ce qu'elle spécifie des ID de VLAN autorise le trafic VLAN vers les VLAN spécifiés.

- **Propriétés spécifiques aux périphériques**

  Utilisez la commande `ldm list-io -d` *pf-name* pour afficher la liste des propriétés spécifiques au périphérique valides. Vous pouvez modifier ces propriétés pour la fonction physique aussi bien que pour la fonction virtuelle. Vous devez utiliser la méthode statique pour modifier les propriétés spécifiques aux périphériques Voir la section "SR-IOV statique" à la page 89. Pour plus d'informations sur les propriétés spécifiques aux périphériques, reportez-vous à la section "Rubriques SR-IOV avancées : SR-IOV Ethernet" à la page 110.

▼ Procédure de modification des propriétés de la fonction virtuelle Ethernet SR-IOV

1. Identifiez le périphérique de fonction physique.

```
primary# ldm list-io
```

Notez que le nom de la fonction physique inclut les informations d'emplacement de la carte SR-IOV PCIe ou du périphérique intégré.

2. Modifiez une propriété de fonction virtuelle.

`ldm set-io` *name=value* [*name=value...*] *vf-name*

Exemple 13 Modification des propriétés de la fonction virtuelle Ethernet

Ces exemples illustrent l'utilisation de la commande `ldm set-io` pour définir les propriétés d'une fonction virtuelle Ethernet.

- L'exemple suivant modifie les propriétés de la fonction virtuelle spécifiée, `/SYS/MB/NET0/IOVNET.PF0.VF0` et indique qu'elle fait partie des ID de VLAN `2`, `3` et `4`.

  ```
  primary# ldm set-io vid=2,3,4 /SYS/MB/NET0/IOVNET.PF0.VF0
  ```

  Notez que cette commande modifie de façon dynamique l'association à un VLAN d'une fonction virtuelle. Pour permettre l'utilisation de ces VLAN, les interfaces VLAN dans les domaines d'E/S doivent être configurées à l'aide des commandes de mise en réseau du SE Oracle Solaris appropriées.
- Dans l'exemple suivant, la valeur de la propriété `pvid` est définie sur `2` pour la fonction virtuelle `/SYS/MB/NET0/IOVNET.PF0.VF0`, ce qui attribue de façon transparente la fonction virtuelle au VLAN `2`. C'est-à-dire que la fonction virtuelle ne visualisera aucun trafic VLAN étiqueté.

  ```
  primary# ldm set-io pvid=2 /SYS/MB/NET0/IOVNET.PF0.VF0
  ```

- L'exemple suivant affecte trois adresses MAC automatiquement attribuées à une fonction virtuelle. Les adresses alternatives permettent la création de cartes d'interface réseau virtuelles (VNIC) Oracle Solaris 11 venant s'ajouter à une fonction virtuelle. Notez que pour utiliser des VNIC, vous devez exécuter le SE Oracle Solaris 11 dans le domaine.

  Remarque - Avant d'exécuter cette commande, arrêtez le domaine propriétaire de la fonction virtuelle.

```
primary# ldm set-io alt-mac-addrs=auto,auto,auto /SYS/MB/NET0/IOVNET.PF0.VF0
```

- Dans l'exemple suivant, la propriété spécifique au périphérique `unicast-slots` est définie sur `12` pour la fonction virtuelle spécifiée. Pour identifier les propriétés spécifiques au périphérique valides pour une fonction physique, utilisez la commande `ldm list-io -d` *pf-name*.

```
primary# ldm set-io unicast-slots=12 /SYS/MB/NET0/IOVNET.PF0.VF0

All configuration changes for other domains are disabled until the primary
domain reboots, at which time the new configuration for the primary domain
will also take effect.
```

Ajout et suppression de fonctions virtuelles SR-IOV Ethernet sur des domaines d'E/S

▼ Procédure d'ajout d'une fonction virtuelle SR-IOV Ethernet à un domaine d'E/S

Si vous ne pouvez pas supprimer la fonction virtuelle de façon dynamique, utilisez la méthode statique. Voir la section "SR-IOV statique" à la page 89.

1. **Identifiez la fonction virtuelle que vous voulez ajouter à un domaine d'E/S.**

```
primary# ldm list-io
```

2. **Ajoutez une fonction virtuelle de manière dynamique ou statique.**

   - **Procédure d'ajout dynamique d'une fonction virtuelle :**

```
primary# ldm add-io vf-name domain-name
```

vf-name est le pseudonyme ou le nom du chemin d'accès de la fonction virtuelle. La pratique recommandée est d'utiliser le pseudonyme. *domain-name* représente le nom du domaine auquel vous ajoutez la fonction virtuelle.

Le nom du chemin d'accès du périphérique de la fonction virtuelle dans le domaine est le chemin indiqué dans la sortie `list-io -l`.

- **Procédure d'ajout statique d'une fonction virtuelle :**

  a. **Lancez une reconfiguration retardée, puis ajoutez la fonction virtuelle.**

  ```
  primary# ldm start-reconf root-domain-name
  primary# ldm add-io vf-name domain-name
  ```

  vf-name est le pseudonyme ou le nom du chemin d'accès de la fonction virtuelle. La pratique recommandée est d'utiliser le pseudonyme. *domain-name* représente le nom du domaine auquel vous ajoutez la fonction virtuelle. L'invité spécifié doit avoir l'état inactif ou lié.

  Le nom du chemin d'accès du périphérique de la fonction virtuelle dans le domaine est le chemin indiqué dans la sortie `list-io -l`.

  b. **Réinitialisez le domaine root.**

  - **Procédure de réinitialisation du domaine root non-`primary` :**

    ```
    primary# ldm stop-domain -r root-domain
    ```

  - **Procédure de réinitialisation du domaine root `primary` :**

    ```
    primary# shutdown -i6 -g0 -y
    ```

Exemple 14 Ajout d'une fonction virtuelle Ethernet

Cet exemple illustre l'ajout dynamique de la fonction virtuelle `/SYS/MB/NET0/IOVNET.PF0.VF0` au domaine `ldg1`.

```
primary# ldm add-io /SYS/MB/NET0/IOVNET.PF0.VF0 ldg1
```

Si vous ne pouvez pas ajouter la fonction virtuelle de manière dynamique, utilisez la méthode statique :

```
primary# ldm stop-domain ldg1
primary# ldm add-io /SYS/MB/NET0/IOVNET.PF0.VF0 ldg1
primary# ldm start-domain ldg1
```

▼ Procédure de suppression d'une fonction virtuelle SR-IOV Ethernet d'un domaine d'E/S

Si vous ne pouvez pas supprimer la fonction virtuelle de façon dynamique, utilisez la méthode statique. Voir la section "SR-IOV statique" à la page 89.

Attention - Avant de supprimer la fonction virtuelle du domaine, assurez-vous qu'elle n'est pas indispensable à l'initialisation du domaine.

1. **Identifiez la fonction virtuelle que vous souhaitez supprimer d'un domaine d'E/S.**

   ```
   primary# ldm list-io
   ```

2. **Supprimez une fonction virtuelle de manière dynamique ou statique.**

   - **Procédure de suppression dynamique d'une fonction virtuelle :**

     ```
     primary# ldm remove-io vf-name domain-name
     ```

     vf-name est le pseudonyme ou le nom du chemin d'accès de la fonction virtuelle. La pratique recommandée est d'utiliser le pseudonyme du périphérique. *domain-name* représente le nom du domaine dans lequel vous souhaitez supprimer la fonction virtuelle.

   - **Procédure de suppression statique d'une fonction virtuelle :**

     a. **Arrêtez le domaine d'E/S.**

        ```
        primary# ldm stop-domain domain-name
        ```

     b. **Supprimez la fonction virtuelle.**

        ```
        primary# ldm remove-io vf-name domain-name
        ```

        vf-name est le pseudonyme ou le nom du chemin d'accès de la fonction virtuelle. La pratique recommandée est d'utiliser le pseudonyme du périphérique. *domain-name* représente le nom du domaine dans lequel vous souhaitez supprimer la fonction virtuelle. L'invité spécifié doit avoir l'état inactif ou lié.

     c. **Démarrez le domaine d'E/S.**

        ```
        primary# ldm start-domain domain-name
        ```

Exemple 15 Suppression dynamique d'une fonction virtuelle Ethernet

Ce exemple illustre la suppression dynamique de la fonction virtuelle `/SYS/MB/NET0/IOVNET.PF0.VF0` du domaine `ldg1`.

```
primary# ldm remove-io /SYS/MB/NET0/IOVNET.PF0.VF0 ldg1
```

Si la commande s'exécute correctement, la fonction virtuelle est supprimée du domaine `ldg1`. Lorsque `ldg1` est redémarré, la fonction virtuelle spécifiée n'apparaît plus dans le domaine.

Si vous ne pouvez pas retirer la fonction virtuelle de manière dynamique, utilisez la méthode statique :

```
primary# ldm stop-domain ldg1
primary# ldm remove-io /SYS/MB/NET0/IOVNET.PF0.VF0 ldg1
primary# ldm start-domain ldg1
```

Rubriques SR-IOV avancées : SR-IOV Ethernet

Cette section décrit quelques-unes des rubriques avancées relatives à l'utilisation des fonctions virtuelles SR-IOV.

Configuration réseau avancée pour les fonctions virtuelles

Lorsque vous utilisez des fonctions virtuelles SR-IOV, tenez compte des problèmes suivants :

- Les fonctions virtuelles SR-IOV peuvent uniquement utiliser les adresses MAC affectées par Logical Domains Manager. Si vous utilisez d'autres commandes réseau du SE Oracle Solaris pour modifier l'adresse MAC du domaine d'E/S, les commandes risquent d'échouer ou de ne pas fonctionner correctement.
- A l'heure actuelle, le groupement de liaisons des fonctions virtuelles réseau SR-IOV dans le domaine d'E/S n'est pas prise en charge. Si vous tentez de créer un groupement de liaisons, il risque de ne pas fonctionner comme prévu.
- Vous pouvez créer des services d'E/S virtuels et les affecter à des domaines d'E/S. Ces services d'E/S virtuels peuvent être créés sur la fonction physique même à partir de laquelle les fonctions virtuelles sont créées. Par exemple, vous pouvez utiliser un périphérique réseau 1 Gbit/s intégré (`net0` ou `igb0`) en tant que périphérique backend réseau pour un commutateur virtuel et créer statistiquement des fonctions virtuelles à partir de ce même périphérique de fonction physique.

Initialisation d'un domaine d'E/S à l'aide d'une fonction virtuelle SR-IOV

Une fonction virtuelle SR-IOV offre le même type de fonctionnalité que n'importe quel autre périphérique PCIe, notamment la possibilité d'utiliser une fonction virtuelle en tant que

périphérique d'initialisation d'un domaine logique. Une fonction virtuelle de réseau par exemple peut être utilisée pour lancer l'initialisation via le réseau afin d'installer le SE Oracle Solaris dans un domaine d'E/S.

Remarque - Lorsque vous initialisez le SE Oracle Solaris à partir d'un périphérique de fonction virtuelle, assurez-vous que le SE Oracle Solaris en cours de chargement prend en charge les périphériques de fonction virtuelle. Si c'est le cas, vous pouvez poursuivre l'installation comme prévu.

Propriétés SR-IOV spécifiques aux périphériques

Les pilotes de périphérique d'une fonction physique SR-IOV peuvent exporter des propriétés spécifiques aux périphériques Celles-ci peuvent être utilisées pour régler l'allocation des ressources de la fonction physique ainsi que celle de ses fonctions virtuelles. Pour plus d'informations sur les propriétés, reportez-vous à la page de manuel relative au pilote de la fonction physique, comme par exemple les pages de manuel `igb`(7D) et `ixgbe`(7D).

La commande `ldm list-io -d` affiche les propriétés spécifiques au périphérique exportées par le pilote de périphérique de la fonction physique spécifiée. Les informations de chaque propriété incluent un nom, une description succincte, une valeur par défaut, des valeurs maximales et un ou plusieurs des indicateurs suivants :

`P`	Concerne une fonction physique
`V`	Concerne une fonction virtuelle
`R`	Paramètre en lecture seule ou à caractère informatif uniquement

```
primary# ldm list-io -d pf-name
```

Utilisez la commande `ldm create-vf` ou la commande `ldm set-io` pour définir les propriétés de lecture-écriture d'une fonction physique ou virtuelle. Notez que pour définir une propriété spécifique au périphérique, vous devez utiliser la méthode statique. Voir la section "SR-IOV statique" à la page 89.

L'exemple suivant montre les propriétés spécifiques au périphérique exportées par le périphérique SR-IOV Intel 1Gbit/s intégré :

```
primary# ldm list-io -d /SYS/MB/NET0/IOVNET.PF0
Device-specific Parameters
--------------------------
max-config-vfs
    Flags = PR
    Default = 7
    Descr = Max number of configurable VFs
max-vf-mtu
```

```
    Flags = VR
    Default = 9216
    Descr = Max MTU supported for a VF
max-vlans
    Flags = VR
    Default = 32
    Descr = Max number of VLAN filters supported
pvid-exclusive
    Flags = VR
    Default = 1
    Descr = Exclusive configuration of pvid required
unicast-slots
    Flags = PV
    Default = 0 Min = 0 Max = 24
    Descr = Number of unicast mac-address slots
```

Dans l'exemple suivant, la propriété `unicast-slots` est définie sur `8` :

```
primary# ldm create-vf unicast-slots=8 /SYS/MB/NET0/IOVNET.PF0
```

Création de cartes d'interface réseau virtuelles (VNIC) sur des fonctions virtuelles SR-IOV

La création de VNIC Oracle Solaris 11 est prise en charge sur les fonctions virtuelles SR-IOV. Cependant, le nombre de VNIC pris en charge est limité au nombre d'adresses MAC alternatives (propriété `alt-mac-addrs`) attribuées à la fonction virtuelle. Veillez à affecter un nombre suffisant d'adresses MAC alternatives lorsque vous utilisez des VNIC sur la fonction virtuelle. Utilisez la commande `ldm create-vf` ou la commande `ldm set-io` pour définir les adresses MAC alternatives à l'aide de la propriété `alt-mac-addrs`.

L'exemple suivant illustre la création de quatre cartes d'interface réseau virtuelles sur une fonction virtuelle SR-IOV. La première commande affecte des adresses MAC alternatives au périphérique de fonction virtuelle. Cette commande utilise la méthode d'allocation automatique pour allouer quatre adresses MAC alternatives au périphérique de fonction virtuelle `/SYS/MB/NET0/IOVNET.PF0.VF0` :

```
primary# ldm set-io alt-mac-addrs=auto,auto,auto,auto /SYS/MB/NET0/IOVNET.PF0.VF0
```

La commande suivante démarre le domaine d'E/S `ldg1`. Etant donné que la propriété `auto-boot?` a pour valeur `true` dans cet exemple, le SE Oracle Solaris 11 est également initialisé dans le domaine d'E/S.

```
primary# ldm start ldg1
```

La commande suivante utilise la commande `dladm` d'Oracle Solaris 11 dans le domaine invité pour afficher la fonction virtuelle possédant des adresses MAC alternatives. Cette sortie indique que la fonction virtuelle `net30` possède quatre adresses MAC alternatives.

```
guest# dladm show-phys -m
LINK              SLOT     ADDRESS            INUSE CLIENT
```

```
net0              primary  0:14:4f:fa:b4:d1   yes   net0
net25             primary  0:14:4f:fa:c9:eb   no    --
net30             primary  0:14:4f:fb:de:4c   no    --
                   1       0:14:4f:f9:e8:73   no    --
                   2       0:14:4f:f8:21:58   no    --
                   3       0:14:4f:fa:9d:92   no    --
                   4       0:14:4f:f9:8f:1d   no    --
```

Les commandes suivantes créent quatre cartes VNIC. Notez que toute tentative de créer plus de carte d'interface réseau virtuelles que spécifié en ayant recours à des adresses MAC alternatives est vouée à l'échec.

```
guest# dladm create-vnic -l net30 vnic0
guest# dladm create-vnic -l net30 vnic1
guest# dladm create-vnic -l net30 vnic2
guest# dladm create-vnic -l net30 vnic3
guest# dladm show-link
LINK                CLASS     MTU    STATE    OVER
net0                phys      1500   up       --
net25               phys      1500   up       --
net30               phys      1500   up       --
vnic0               vnic      1500   up       net30
vnic1               vnic      1500   up       net30
vnic2               vnic      1500   up       net30
vnic3               vnic      1500   up       net30
```

Utilisation d'une fonction virtuelle SR-IOV pour créer un domaine d'E/S

La procédure suivante décrit la création d'un domaine d'E/S incluant des fonctions virtuelles SR-IOV PCIe.

▼ Procédure de création d'un domaine d'E/S par affectation d'une fonction virtuelle SR-IOV

Planifiez à l'avance le nombre de réinitialisations du domaine root pour réduire les indisponibilités.

Avant de commencer

Avant de commencer, assurez-vous que vous avez activé la virtualisation d'E/S pour le bus PCIe qui est le parent de la fonction physique à partir de laquelle vous créez des fonctions virtuelles. Voir la section "Procédure d'activation de la virtualisation d'E/S pour un bus PCIe" à la page 93.

1. **Identifiez une fonction physique SR-IOV à partager avec un domaine d'E/S utilisant la fonctionnalité SR-IOV.**

```
primary# ldm list-io
```

2. **Créez une ou plusieurs fonctions virtuelles pour la fonction physique.**

   ```
   primary# ldm create-vf pf-name
   ```

   Vous pouvez exécuter cette commande pour chaque fonction virtuelle que vous souhaitez créer. Vous pouvez également utiliser l'option `-n` pour créer plusieurs fonctions virtuelles à partir de la même fonction physique d'une seule commande. Reportez-vous à la section Exemple 10, "Création de plusieurs fonctions virtuelles SR-IOV Ethernet" et à la page de manuel `ldm`(1M).

   Remarque - Cette commande échoue si d'autres fonctions virtuelles ont déjà été créées à partir de la fonction physique associée et si l'une de ces fonctions virtuelles est liée à un autre domaine.

3. **Affichez la liste des fonctions virtuelles disponibles sur le domaine root.**

   ```
   primary# ldm list-io
   ```

4. **Assignez la fonction virtuelle créée dans Étape 2 à son domaine d'E/S cible.**

   ```
   primary# ldm add-io vf-name domain-name
   ```

   Remarque - Si le SE du domaine d'E/S cible ne prend pas en charge la méthode SR-IOV dynamique, utilisez la méthode statique. Voir la section "SR-IOV statique" à la page 89.

5. **Vérifiez que la fonction virtuelle est disponible sur le domaine d'E/S.**

   La commande Oracle Solaris 11 suivante affiche la disponibilité de la fonction virtuelle :

   ```
   guest# dladm show-phys
   ```

Exemple 16 Création dynamique d'un domaine d'E/S par assignation d'une fonction virtuelle SR-IOV

L'exemple suivant illustre la création dynamique d'une fonction virtuelle, `/SYS/MB/NET0/IOVNET.PF0.VF0`, pour une fonction physique, `/SYS/MB/NET0/IOVNET.PF0`, et l'assignation de la fonction virtuelle au domaine d'E/S `ldg1`.

Cet exemple suppose les circonstances suivantes :

- Le SE du domaine `primary` prend en charge les opérations SR-IOV dynamiques
- Le bus `pci_0` est assigné au domaine `primary` et a été initialisé pour les opérations de virtualisation d'E/S
- La fonction physique `/SYS/MB/NET0/IOVNET.PF0` appartient au bus `pci_0`
- La fonction physique `/SYS/MB/NET0/IOVNET.PF0` ne comporte pas de fonctions virtuelles assignées à des domaines
- Le domaine `ldg1` est actif et initialisé et son SE prend en charge les opérations SR-IOV dynamiques

Créez la fonction virtuelle à partir de la fonction physique `/SYS/MB/NET0/IOVNET.PF0`.

```
primary# ldm create-vf /SYS/MB/NET0/IOVNET.PF0
Created new vf: /SYS/MB/NET0/IOVNET.PF0.VF0
```

Ajoutez la fonction virtuelle `/SYS/MB/NET0/IOVNET.PF0.VF0` au domaine `ldg1`.

```
primary# ldm add-io /SYS/MB/NET0/IOVNET.PF0.VF0 ldg1
```

La commande suivante indique que la fonction virtuelle a été ajoutée au domaine `ldg1`.

```
primary# ldm list-io
NAME                          TYPE  BUS      DOMAIN    STATUS
----                          ----  ---      ------    ------
niu_0                         NIU   niu_0    primary
niu_1                         NIU   niu_1    primary
pci_0                         BUS   pci_0    primary   IOV
pci_1                         BUS   pci_1    primary
/SYS/MB/PCIE0                 PCIE  pci_0    primary   OCC
/SYS/MB/PCIE2                 PCIE  pci_0    primary   OCC
/SYS/MB/PCIE4                 PCIE  pci_0    primary   OCC
/SYS/MB/PCIE6                 PCIE  pci_0    primary   EMP
/SYS/MB/PCIE8                 PCIE  pci_0    primary   EMP
/SYS/MB/SASHBA                PCIE  pci_0    primary   OCC
/SYS/MB/NET0                  PCIE  pci_0    primary   OCC
/SYS/MB/PCIE1                 PCIE  pci_1    primary   OCC
/SYS/MB/PCIE3                 PCIE  pci_1    primary   OCC
/SYS/MB/PCIE5                 PCIE  pci_1    primary   OCC
/SYS/MB/PCIE7                 PCIE  pci_1    primary   EMP
/SYS/MB/PCIE9                 PCIE  pci_1    primary   EMP
/SYS/MB/NET2                  PCIE  pci_1    primary   OCC
/SYS/MB/NET0/IOVNET.PF0       PF    pci_0    primary
/SYS/MB/NET0/IOVNET.PF1       PF    pci_0    primary
/SYS/MB/PCIE5/IOVNET.PF0      PF    pci_1    primary
/SYS/MB/PCIE5/IOVNET.PF1      PF    pci_1    primary
/SYS/MB/NET2/IOVNET.PF0       PF    pci_1    primary
/SYS/MB/NET2/IOVNET.PF1       PF    pci_1    primary
/SYS/MB/NET0/IOVNET.PF0.VF0   VF    pci_0    ldg1
```

Exemple 17 Création statique d'un domaine d'E/S par assignation d'une fonction virtuelle SR-IOV

L'exemple statique suivant illustre la création d'une fonction virtuelle, `/SYS/MB/NET0/IOVNET.PF0.VF0`, pour une fonction physique, `/SYS/MB/NET0/IOVNET.PF0`, et l'assignation de la fonction virtuelle au domaine `ldg1`.

Cet exemple suppose les circonstances suivantes :

- Le SE du domaine `primary` ne prend pas en charge les opérations SR-IOV dynamiques
- Le bus `pci_0` est assigné au domaine `primary` et n'a pas été initialisé pour les opérations de virtualisation d'E/S.
- La fonction physique `/SYS/MB/NET0/IOVNET.PF0` appartient au bus `pci_0`
- La fonction physique `/SYS/MB/NET0/IOVNET.PF0` ne comporte pas de fonctions virtuelles assignées à des domaines
- Le domaine `ldg1` est actif et initialisé et son SE ne prend pas en charge les opérations SR-IOV dynamiques

- La propriété `auto-boot?` du domaine `ldg1` est définie sur `true`, de sorte que le domaine est automatiquement initialisé lors du démarrage du domaine

Démarrez d'abord une reconfiguration retardée sur le domaine `primary`, activez la virtualisation d'E/S et créez la fonction virtuelle à partir de la fonction physique `/SYS/MB/NET0/IOVNET.PF0`.

```
primary# ldm start-reconf primary
Initiating a delayed reconfiguration operation on the primary domain.
All configuration changes for other domains are disabled until the primary
domain reboots, at which time the new configuration for the primary domain
will also take effect.

primary# ldm set-io iov=on pci_0
primary# ldm create-vf /SYS/MB/NET0/IOVNET.PF0

--------------------------------------------------------------------------------
Notice: The primary domain is in the process of a delayed reconfiguration.
Any changes made to the primary domain will only take effect after it reboots.
--------------------------------------------------------------------------------
Created new vf: /SYS/MB/NET0/IOVNET.PF0.VF0
```

Ensuite, arrêtez le domaine `primary`.

```
primary# shutdown -i6 -g0 -y
```

Arrêtez le domaine `ldg1`, ajoutez la fonction virtuelle et démarrez le domaine.

```
primary# ldm stop ldg1
primary# ldm add-io /SYS/MB/NET0/IOVNET.PF0.VF0 ldg1
primary# ldm start ldg1
```

La commande suivante indique que la fonction virtuelle a été ajoutée au domaine `ldg1`.

```
primary# ldm list-io
NAME                          TYPE  BUS      DOMAIN    STATUS
----                          ----  ---      ------    ------
niu_0                         NIU   niu_0    primary
niu_1                         NIU   niu_1    primary
pci_0                         BUS   pci_0    primary   IOV
pci_1                         BUS   pci_1    primary
/SYS/MB/PCIE0                 PCIE  pci_0    primary   OCC
/SYS/MB/PCIE2                 PCIE  pci_0    primary   OCC
/SYS/MB/PCIE4                 PCIE  pci_0    primary   OCC
/SYS/MB/PCIE6                 PCIE  pci_0    primary   EMP
/SYS/MB/PCIE8                 PCIE  pci_0    primary   EMP
/SYS/MB/SASHBA                PCIE  pci_0    primary   OCC
/SYS/MB/NET0                  PCIE  pci_0    primary   OCC
/SYS/MB/PCIE1                 PCIE  pci_1    primary   OCC
/SYS/MB/PCIE3                 PCIE  pci_1    primary   OCC
/SYS/MB/PCIE5                 PCIE  pci_1    primary   OCC
/SYS/MB/PCIE7                 PCIE  pci_1    primary   EMP
/SYS/MB/PCIE9                 PCIE  pci_1    primary   EMP
/SYS/MB/NET2                  PCIE  pci_1    primary   OCC
/SYS/MB/NET0/IOVNET.PF0       PF    pci_0    primary
/SYS/MB/NET0/IOVNET.PF1       PF    pci_0    primary
/SYS/MB/PCIE5/IOVNET.PF0      PF    pci_1    primary
/SYS/MB/PCIE5/IOVNET.PF1      PF    pci_1    primary
/SYS/MB/NET2/IOVNET.PF0       PF    pci_1    primary
/SYS/MB/NET2/IOVNET.PF1       PF    pci_1    primary
/SYS/MB/NET0/IOVNET.PF0.VF0   VF    pci_0    ldg1
```

Utilisation des fonctions virtuelles SR-IOV InfiniBand

Remarque - Vous pouvez utiliser la méthode SR-IOV statique pour les périphériques SR-IOV InfiniBand.

Pour réduire les indisponibilités, exécutez l'ensemble des commandes SR-IOV en tant que groupe pendant que le domaine root est en reconfiguration retardée ou qu'un domaine invité est arrêté. Les commandes SR-IOV restreintes de cette manière sont les suivantes : `ldm create-vf`, `ldm destroy-vf`, `ldm add-io` et `ldm remove-io`.

En règle générale, les fonctions virtuelles sont assignées à plusieurs domaines invités. Une réinitialisation du domaine root affecte tous les domaines invités auxquels les fonctions virtuelles du domaine root ont été assignées.

Etant donné qu'une fonction virtuelle InfiniBand inutilisée mobilise peu de temps système, vous pouvez éviter les périodes d'indisponibilité en créant les fonctions virtuelles nécessaires au préalable, même si elles ne sont pas immédiatement utilisées.

Configuration matérielle requise pour SR-IOV InfiniBand

Pour plus d'informations sur le matériel InfiniBand SR-IOV PCIe requis, reportez-vous à la section "Configuration matérielle et logicielle requise pour SR-IOV" à la page 84.

Pour la prise en charge de SR-IOV InfiniBand, le domaine root doit exécuter au moins le système d'exploitation Oracle Solaris 11.1 SRU 10. Les domaines d'E/S peuvent exécuter au moins le SE Oracle Solaris 11.1 SRU 10.

Création et destruction de fonctions virtuelles InfiniBand

▼ Procédure de création d'une fonction virtuelle InfiniBand

Cette procédure indique comment créer une fonction virtuelle SR-IOV InfiniBand.

1. **Lancez une reconfiguration retardée sur le domaine root.**

   ```
   primary# ldm start-reconf root-domain-name
   ```

2. **Activez la virtualisation d'E/S en définissant `iov=on`.**

   Effectuez uniquement cette étape si la virtualisation d'E/S n'est pas déjà activée pour le bus qui a la fonction physique.

   ```
   primary# ldm set-io iov=on bus
   ```

3. **Créez une ou plusieurs fonctions virtuelles associées aux fonctions physiques issues de ce domaine root.**

   ```
   primary# ldm create-vf pf-name
   ```

   Vous pouvez exécuter cette commande pour chaque fonction virtuelle que vous souhaitez créer. Vous pouvez également utiliser l'option `-n` pour créer plusieurs fonctions virtuelles à partir de la même fonction physique d'une seule commande. Reportez-vous à la section Exemple 10, "Création de plusieurs fonctions virtuelles SR-IOV Ethernet" et à la page de manuel `ldm`(1M).

4. **Réinitialisez le domaine root.**

   Exécutez l'une des commandes suivantes :

   - **Réinitialisation du domaine root s'il est différent de `primary`.**

     ```
     primary# ldm stop-domain -r root-domain
     ```

   - **Réinitialisation du domaine root `primary`.**

     ```
     primary# shutdown -i6 -g0 -y
     ```

Exemple 18 Création d'une fonction virtuelle InfiniBand

L'exemple suivant affiche des informations sur la fonction physique `/SYS/MB/RISER1/PCIE4/IOVIB.PF0` :

- Cette fonction physique se trouve dans l'emplacement PCIe 4.
- La chaîne `IOVIB` indique que la fonction physique est un périphérique SR-IOV InfiniBand.

```
primary# ldm list-io
NAME                                      TYPE   BUS      DOMAIN   STATUS
----                                      ----   ---      ------   ------
pci_0                                     BUS    pci_0    primary
niu_0                                     NIU    niu_0    primary
/SYS/MB/RISER0/PCIE0                      PCIE   pci_0    primary  EMP
/SYS/MB/RISER1/PCIE1                      PCIE   pci_0    primary  EMP
/SYS/MB/RISER2/PCIE2                      PCIE   pci_0    primary  EMP
/SYS/MB/RISER0/PCIE3                      PCIE   pci_0    primary  OCC
/SYS/MB/RISER1/PCIE4                      PCIE   pci_0    primary  OCC
/SYS/MB/RISER2/PCIE5                      PCIE   pci_0    primary  EMP
```

```
/SYS/MB/SASHBA0                          PCIE   pci_0    primary  OCC
/SYS/MB/SASHBA1                          PCIE   pci_0    primary  OCC
/SYS/MB/NET0                             PCIE   pci_0    primary  OCC
/SYS/MB/NET2                             PCIE   pci_0    primary  OCC
/SYS/MB/RISER0/PCIE3/IOVIB.PF0           PF     pci_0    primary
/SYS/MB/RISER1/PCIE4/IOVIB.PF0           PF     pci_0    primary
/SYS/MB/NET0/IOVNET.PF0                  PF     pci_0    primary
/SYS/MB/NET0/IOVNET.PF1                  PF     pci_0    primary
/SYS/MB/NET2/IOVNET.PF0                  PF     pci_0    primary
/SYS/MB/NET2/IOVNET.PF1                  PF     pci_0    primary
```

La commande suivante fournit plus d'informations sur la fonction physique spécifiée. La valeur `maxvfs` indique le nombre maximal de fonctions virtuelles prises en charge par le périphérique.

```
primary# ldm list-io -l /SYS/MB/RISER1/PCIE4/IOVIB.PF0
NAME                                     TYPE   BUS      DOMAIN   STATUS
----                                     ----   ---      ------   ------
/SYS/MB/RISER1/PCIE4/IOVIB.PF0           PF     pci_0    primary
[pci@400/pci@1/pci@0/pci@0/pciex15b3,673c@0]
    maxvfs = 64
```

L'exemple suivant indique comment créer une fonction virtuelle statique. Lancez tout d'abord une reconfiguration retardée sur le domaine `primary` et activez la virtualisation d'E/S sur le bus PCIe `pci_0` . Etant donné que le bus `pci_0` a déjà été assigné au domaine root `primary`, vous devez utiliser la commande `ldm set-io` pour activer la virtualisation d'E/S.

```
primary# ldm start-reconf primary
Initiating a delayed reconfiguration operation on the primary domain.
All configuration changes for other domains are disabled until the primary
domain reboots, at which time the new configuration for the primary domain
will also take effect.

primary# ldm set-io iov=on pci_0
------------------------------------------------------------------------
Notice: The primary domain is in the process of a delayed reconfiguration.
Any changes made to the primary domain will only take effect after it reboots.
------------------------------------------------------------------------
```

Vous pouvez ensuite utiliser la commande `ldm create-vf` pour créer une fonction virtuelle à partir de la fonction physique `/SYS/MB/RISER1/PCIE4/IOVIB.PF0`.

```
primary# ldm create-vf /SYS/MB/RISER1/PCIE4/IOVIB.PF0
------------------------------------------------------------------------
Notice: The primary domain is in the process of a delayed reconfiguration.
Any changes made to the primary domain will only take effect after it reboots.
------------------------------------------------------------------------
Created new vf: /SYS/MB/RISER1/PCIE4/IOVIB.PF0.VF0
```

Notez que vous pouvez créer plusieurs fonctions virtuelles au cours d'une même reconfiguration retardée. La commande suivante crée une deuxième fonction virtuelle :

```
primary# ldm create-vf /SYS/MB/RISER1/PCIE4/IOVIB.PF0
------------------------------------------------------------------------
Notice: The primary domain is in the process of a delayed reconfiguration.
Any changes made to the primary domain will only take effect after it reboots.
------------------------------------------------------------------------
Created new vf: /SYS/MB/RISER1/PCIE4/IOVIB.PF0.VF1
```

Pour finir, réinitialisez le domaine root `primary` pour appliquer les modifications.

```
primary# shutdown -i6 -g0 -y
Shutdown started.

Changing to init state 6 - please wait
...
```

▼ Procédure de destruction d'une fonction virtuelle InfiniBand

Cette procédure indique comment détruire une fonction virtuelle SR-IOV InfiniBand.

Vous pouvez détruire une fonction virtuelle si elle n'est pas affectée à un domaine. Vous pouvez uniquement détruire une fonction virtuelle dans l'ordre inverse de la création en d'autres termes, seule la dernière fonction virtuelle créée peut être détruite. La configuration résultante est validée par le pilote de la fonction physique.

1. **Lancez une reconfiguration retardée sur le domaine root.**

   ```
   primary# ldm start-reconf root-domain-name
   ```

2. **Détruisez une ou plusieurs des fonctions virtuelles associées aux fonctions physiques issues de ce domaine root.**

   ```
   primary# ldm destroy-vf vf-name
   ```

   Vous pouvez exécuter cette commande pour chaque fonction virtuelle que vous souhaitez détruire. Vous pouvez également utiliser l'option `-n` pour détruire plusieurs fonctions virtuelles à partir de la même fonction physique d'une seule commande. Reportez-vous à la section Exemple 12, "Destruction de plusieurs fonctions virtuelles SR-IOV Ethernet" et à la page de manuel `ldm`(1M).

3. **Réinitialisez le domaine root.**

   Exécutez l'une des commandes suivantes :

   - **Réinitialisation du domaine root s'il est différent de `primary`.**

     ```
     primary# ldm stop-domain -r root-domain
     ```

   - **Réinitialisation du domaine root `primary`.**

     ```
     primary# shutdown -i6 -g0 -y
     ```

Exemple 19 Destruction d'une fonction virtuelle InfiniBand

L'exemple suivant illustre la destruction de la fonction virtuelle InfiniBand statique `/SYS/MB/RISER1/PCIE4/IOVIB.PF0.VF1`.

La commande `ldm list-io` affiche des informations sur les bus, les fonctions physiques et les fonctions virtuelles.

```
primary# ldm list-io
NAME                                      TYPE   BUS      DOMAIN STATUS
----                                      ----   ---      ------ ------
pci_0                                     BUS    pci_0    primary  IOV
...
/SYS/MB/RISER1/PCIE4/IOVIB.PF0            PF     pci_0    primary
...
/SYS/MB/RISER1/PCIE4/IOVIB.PF0.VF0        VF     pci_0
/SYS/MB/RISER1/PCIE4/IOVIB.PF0.VF1        VF     pci_0
```

La commande `ldm list-io -l` permet d'obtenir des informations supplémentaires sur la fonction physique et les fonctions virtuelles associées.

```
primary# ldm list-io -l /SYS/MB/RISER1/PCIE4/IOVIB.PF0
NAME                                      TYPE   BUS      DOMAIN STATUS
----                                      ----   ---      ------ ------
/SYS/MB/RISER1/PCIE4/IOVIB.PF0            PF     pci_0    primary
[pci@400/pci@1/pci@0/pci@0/pciex15b3,673c@0]
    maxvfs = 64
/SYS/MB/RISER1/PCIE4/IOVIB.PF0.VF0        VF     pci_0
[pci@400/pci@1/pci@0/pci@0/pciex15b3,673c@0,1]
/SYS/MB/RISER1/PCIE4/IOVIB.PF0.VF1        VF     pci_0
[pci@400/pci@1/pci@0/pci@0/pciex15b3,673c@0,2]
```

Vous pouvez uniquement détruire une fonction virtuelle si elle n'est pas assignée à un domaine. La colonne DOMAIN de la sortie `ldm list-io -l` affiche le nom de tous les domaines auxquels une fonction virtuelle est affectée. De plus, les fonctions virtuelles doivent être détruites dans l'ordre inverse de leur création. Par conséquent, dans cet exemple, vous devez détruire la fonction virtuelle `/SYS/MB/RISER1/PCIE4/IOVIB.PF0.VF1` avant de pouvoir détruire la fonction virtuelle `/SYS/MB/RISER1/PCIE4/IOVIB.PF0.VF0`.

Après avoir identifié la fonction virtuelle appropriée, vous pouvez la détruire. Commencez par lancer une reconfiguration retardée.

```
primary# ldm start-reconf primary
Initiating a delayed reconfiguration operation on the primary domain.
All configuration changes for other domains are disabled until the primary
domain reboots, at which time the new configuration for the primary domain
will also take effect.

primary# ldm destroy-vf /SYS/MB/RISER1/PCIE4/IOVIB.PF0.VF1
------------------------------------------------------------------------------
Notice: The primary domain is in the process of a delayed reconfiguration.
Any changes made to the primary domain will only take effect after it reboots.
------------------------------------------------------------------------------
```

Vous pouvez exécuter plusieurs commandes `ldm destroy-vf` dans le cadre d'une reconfiguration retardée. Par conséquent, vous pourriez également détruire la fonction virtuelle `/SYS/MB/RISER1/PCIE4/IOVIB.PF0.VF0`.

Pour finir, réinitialisez le domaine root `primary` pour appliquer les modifications.

```
primary# shutdown -i6 -g0 -y
```

```
Shutdown started.

Changing to init state 6 - please wait
...
```

Ajout et suppression de fonctions virtuelles InfiniBand sur des domaines d'E/S

▼ Procédure d'ajout d'une fonction virtuelle InfiniBand à un domaine d'E/S

Cette procédure indique comment ajouter une fonction virtuelle SR-IOV InfiniBand à un domaine d'E/S.

1. **Arrêtez le domaine d'E/S.**

   ```
   primary# ldm stop-domain domain-name
   ```

2. **Ajoutez une ou plusieurs fonctions virtuelles au domaine d'E/S.**

   vf-name est le pseudonyme ou le nom du chemin d'accès de la fonction virtuelle. La pratique recommandée est d'utiliser le pseudonyme. *domain-name* représente le nom du domaine auquel vous ajoutez la fonction virtuelle. Le domaine d'E/S spécifié doit avoir l'état inactif ou lié.

   ```
   primary# ldm add-io vf-name domain-name
   ```

3. **Démarrez le domaine d'E/S.**

   ```
   primary# ldm start-domain domain-name
   ```

Exemple 20 Ajout d'une fonction virtuelle InfiniBand

L'exemple suivant indique comment ajouter la fonction virtuelle `/SYS/MB/RISER1/PCIE4/IOVIB.PF0.VF2` au domaine d'E/S `iodom1`.

Commencez par identifier la fonction virtuelle à assigner.

```
primary# ldm list-io
NAME                                    TYPE   BUS      DOMAIN STATUS
----                                    ----   ---      ------ ------
pci_0                                   BUS    pci_0    primary  IOV
...
/SYS/MB/RISER1/PCIE4/IOVIB.PF0          PF     pci_0    primary
...
/SYS/MB/RISER1/PCIE4/IOVIB.PF0.VF0      VF     pci_0
/SYS/MB/RISER1/PCIE4/IOVIB.PF0.VF1      VF     pci_0
/SYS/MB/RISER1/PCIE4/IOVIB.PF0.VF2      VF     pci_0
/SYS/MB/RISER1/PCIE4/IOVIB.PF0.VF3      VF     pci_0
```

Pour ajouter une fonction virtuelle à un domaine d'E/S, celle-ci ne doit pas être assignée. La colonne DOMAIN indique le nom du domaine auquel la fonction virtuelle est assignée. Dans ce cas, `/SYS/MB/RISER1/PCIE4/IOVIB.PF0.VF2` n'est assigné à aucun domaine.

Pour ajouter une fonction virtuelle à un domaine, le domaine doit être inactif ou lié.

```
primary# ldm list-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  NORM  UPTIME
primary          active     -n-cv-  UART    32    64G      0.2%  0.2%  56m
iodom1           active     -n----  5000    8     8G        33%   33%  25m
```

La sortie `ldm list-domain` montre que le domaine d'E/S `iodom1` est actif et qu'il doit donc être arrêté.

```
primary# ldm stop iodom1
LDom iodom1 stopped
primary# ldm list-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  NORM  UPTIME
primary          active     -n-cv-  UART    32    64G      0.0%  0.0%  57m
iodom1           bound      ------  5000    8     8G
```

Vous pouvez à présent ajouter la fonction virtuelle au domaine d'E/S.

```
primary# ldm add-io /SYS/MB/RISER1/PCIE4/IOVIB.PF0.VF2 iodom1
primary# ldm list-io
...
/SYS/MB/RISER1/PCIE4/IOVIB.PF0.VF2        VF     pci_0    iodom1
```

Notez que vous pouvez ajouter plusieurs fonctions virtuelles pendant l'arrêt d'un domaine d'E/S. Par exemple, vous pouvez ajouter d'autres fonctions virtuelles non assignées telles que `/SYS/MB/RISER1/PCIE4/IOVIB.PF0.VF3` à `iodom1`. Une fois que vous avez ajouté les fonctions virtuelles, vous pouvez redémarrer le domaine d'E/S.

```
primary# ldm start iodom1
LDom iodom1 started
primary# ldm list-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  NORM  UPTIME
primary          active     -n-cv-  UART    32    64G      1.0%  1.0%  1h 18m
iodom1           active     -n----  5000    8     8G        36%   36%  1m
```

▼ Procédure de suppression d'une fonction virtuelle InfiniBand d'un domaine d'E/S

Cette procédure indique comment supprimer une fonction virtuelle SR-IOV InfiniBand d'un domaine d'E/S.

1. **Arrêtez le domaine d'E/S.**

```
primary# ldm stop-domain domain-name
```

2. **Retirez une ou plusieurs fonctions virtuelles du domaine d'E/S.**

vf-name est le pseudonyme ou le nom du chemin d'accès de la fonction virtuelle. La pratique recommandée est d'utiliser le pseudonyme du périphérique. *domain-name* représente le nom du domaine dans lequel vous souhaitez supprimer la fonction virtuelle. Le domaine d'E/S spécifié doit avoir l'état inactif ou lié.

Remarque - Avant de supprimer la fonction virtuelle du domaine d'E/S, assurez-vous qu'elle n'est pas indispensable à l'initialisation du domaine.

```
primary# ldm remove-io vf-name  domain-name
```

3. Démarrez le domaine d'E/S.

```
primary# ldm start-domain domain-name
```

Exemple 21 Suppression d'une fonction virtuelle InfiniBand

L'exemple suivant illustre la suppression de la fonction virtuelle `/SYS/MB/RISER1/PCIE4/IOVIB.PF0.VF2` du domaine d'E/S `iodom1`.

Commencez par identifier les fonctions virtuelles à supprimer.

```
primary# ldm list-io
NAME                                    TYPE   BUS      DOMAIN STATUS
----                                    ----   ---      ------ ------
pci_0                                   BUS    pci_0    primary  IOV
...
/SYS/MB/RISER1/PCIE4/IOVIB.PF0          PF     pci_0    primary
...
/SYS/MB/RISER1/PCIE4/IOVIB.PF0.VF0      VF     pci_0
/SYS/MB/RISER1/PCIE4/IOVIB.PF0.VF1      VF     pci_0
/SYS/MB/RISER1/PCIE4/IOVIB.PF0.VF2      VF     pci_0    iodom1
/SYS/MB/RISER1/PCIE4/IOVIB.PF0.VF3      VF     pci_0    iodom1
```

La colonne DOMAIN indique le nom du domaine auquel la fonction virtuelle est assignée. La fonction virtuelle `/SYS/MB/RISER1/PCIE4/IOVIB.PF0.VF2` est assignée à `iodom1`.

Vous pouvez uniquement supprimer une fonction virtuelle d'un domaine d'E/S si celui-ci présente l'état inactif ou lié. Utilisez la commande `ldm list-domain` pour déterminer l'état du domaine.

```
primary# ldm list-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  NORM  UPTIME
primary          active     -n-cv-  UART    32    64G      0.3%  0.3%  29m
iodom1           active     -n----  5000    8     8G        17%   17%  11m
```

Dans ce cas, le domaine `iodom1` est actif et doit donc être arrêté.

```
primary# ldm stop iodom1
LDOM iodom1 stopped
primary# ldm list-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  NORM  UPTIME
```

```
primary          active     -n-cv-  UART    32    64G      0.0%  0.0%  31m
iodom1           bound      ------  5000    8     8G
```

Vous pouvez à présent supprimer la fonction virtuelle `/SYS/MB/RISER1/PCIE4/IOVIB.PF0.VF2` de `iodom1`.

```
primary# ldm remove-io /SYS/MB/RISER1/PCIE4/IOVIB.PF0.VF2 iodom1
primary# ldm list-io
NAME                                    TYPE   BUS      DOMAIN STATUS
----                                    ----   ---      ------ ------
...
/SYS/MB/RISER1/PCIE4/IOVIB.PF0.VF2      VF     pci_0
...
```

Notez que la colonne DOMAIN de la fonction virtuelle est maintenant vide.

Vous pouvez supprimer plusieurs fonctions virtuelles lorsqu'un domaine d'E/S est arrêté. Dans cet exemple, vous pourriez également supprimer la fonction virtuelle `/SYS/MB/RISER1/PCIE4/IOVIB.PF0.VF3`. Une fois que vous avez supprimé les fonctions virtuelles, vous pouvez redémarrer le domaine d'E/S.

```
primary# ldm start iodom1
LDom iodom1 started
primary# ldm list-domain
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  NORM  UPTIME
primary          active     -n-cv-  UART    32    64G      0.3%  0.3%  39m
iodom1           active     -n----  5000    8     8G       9.4%  9.4%  5s
```

Ajout et suppression de fonctions virtuelles InfiniBand de domaines root

▼ Procédure d'ajout d'une fonction virtuelle InfiniBand à un domaine root

Cette procédure indique comment ajouter une fonction virtuelle SR-IOV InfiniBand à un domaine root.

1. Lancez une reconfiguration retardée.

```
primary# ldm start-reconf root-domain
```

2. Ajoutez une ou plusieurs fonctions virtuelles au domaine root.

vf-name est le pseudonyme ou le nom du chemin d'accès de la fonction virtuelle. La pratique recommandée est d'utiliser le pseudonyme. *root-domain-name* spécifie le nom du domaine root duquel vous supprimez la fonction virtuelle.

```
primary# ldm add-io vf-name root-domain-name
```

3. **Réinitialisez le domaine root.**
   Exécutez l'une des commandes suivantes :

   - **Réinitialisation du domaine root s'il est différent de `primary`.**

     ```
     primary# ldm stop-domain -r root-domain-name
     ```

   - **Réinitialisation du domaine root `primary`.**

     ```
     primary# shutdown -i6 -g0 -y
     ```

▼ Procédure de suppression d'une fonction virtuelle InfiniBand d'un domaine root

Cette procédure indique comment supprimer une fonction virtuelle SR-IOV InfiniBand d'un domaine root.

1. **Lancez une reconfiguration retardée.**

   ```
   primary# ldm start-reconf root-domain
   ```

2. **Supprimez une ou plusieurs fonctions virtuelles du domaine root.**
   vf-name est le pseudonyme ou le nom du chemin d'accès de la fonction virtuelle. La pratique recommandée est d'utiliser le pseudonyme. *root-domain-name* spécifie le nom du domaine root duquel vous supprimez la fonction virtuelle.

   ```
   primary# ldm remove-io vf-name root-domain-name
   ```

3. **Réinitialisez le domaine root.**
   Exécutez l'une des commandes suivantes :

   - **Réinitialisation du domaine root s'il est différent de `primary`.**

     ```
     primary# ldm stop-domain -r root-domain-name
     ```

   - **Réinitialisation du domaine root `primary`.**

     ```
     primary# shutdown -i6 -g0 -y
     ```

Rubriques SR-IOV avancées : SR-IOV InfiniBand

Cette section indique comment identifier les fonctions physiques et virtuelles InfiniBand et mettre en corrélation les vues Oracle Solaris et Logical Domains Manager des fonctions physiques et virtuelles InfiniBand.

Liste des fonctions virtuelles SR-IOV InfiniBand

L'exemple suivant illustre différentes manières d'afficher les informations relatives à la fonction physique `/SYS/MB/RISER1/PCIE4/IOVIB.PF0`. La présence de la chaîne `IOVIB` dans le nom de fonction physique indique qu'il s'agit d'un périphérique SR-IOV InfiniBand.

```
primary# ldm list-io
NAME                                    TYPE   BUS      DOMAIN   STATUS
----                                    ----   ---      ------   ------
pci_0                                   BUS    pci_0    primary  IOV
niu_0                                   NIU    niu_0    primary
/SYS/MB/RISER0/PCIE0                    PCIE   pci_0    primary  EMP
/SYS/MB/RISER1/PCIE1                    PCIE   pci_0    primary  EMP
/SYS/MB/RISER2/PCIE2                    PCIE   pci_0    primary  EMP
/SYS/MB/RISER0/PCIE3                    PCIE   pci_0    primary  OCC
/SYS/MB/RISER1/PCIE4                    PCIE   pci_0    primary  OCC
/SYS/MB/RISER2/PCIE5                    PCIE   pci_0    primary  EMP
/SYS/MB/SASHBA0                         PCIE   pci_0    primary  OCC
/SYS/MB/SASHBA1                         PCIE   pci_0    primary  OCC
/SYS/MB/NET0                            PCIE   pci_0    primary  OCC
/SYS/MB/NET2                            PCIE   pci_0    primary  OCC
/SYS/MB/RISER0/PCIE3/IOVIB.PF0          PF     pci_0    primary
/SYS/MB/RISER1/PCIE4/IOVIB.PF0          PF     pci_0    primary
/SYS/MB/NET0/IOVNET.PF0                 PF     pci_0    primary
/SYS/MB/NET0/IOVNET.PF1                 PF     pci_0    primary
/SYS/MB/NET2/IOVNET.PF0                 PF     pci_0    primary
/SYS/MB/NET2/IOVNET.PF1                 PF     pci_0    primary
/SYS/MB/RISER0/PCIE3/IOVIB.PF0.VF0      VF     pci_0    primary
/SYS/MB/RISER0/PCIE3/IOVIB.PF0.VF1      VF     pci_0    primary
/SYS/MB/RISER0/PCIE3/IOVIB.PF0.VF2      VF     pci_0    iodom1
/SYS/MB/RISER0/PCIE3/IOVIB.PF0.VF3      VF     pci_0    iodom1
/SYS/MB/RISER1/PCIE4/IOVIB.PF0.VF0      VF     pci_0    primary
/SYS/MB/RISER1/PCIE4/IOVIB.PF0.VF1      VF     pci_0    primary
/SYS/MB/RISER1/PCIE4/IOVIB.PF0.VF2      VF     pci_0    iodom1
/SYS/MB/RISER1/PCIE4/IOVIB.PF0.VF3      VF     pci_0    iodom1
```

La commande `ldm list-io -l` fournit des informations plus détaillées sur le périphérique de fonction physique spécifié, `/SYS/MB/RISER1/PCIE4/IOVIB.PF0`. La valeur `maxvfs` indique que le nombre maximal de fonctions virtuelles pris en charge par le périphérique physique est 64. Pour chaque fonction virtuelle associée à la fonction physique, la sortie affiche ce qui suit :

- Nom de fonction
- Type de fonction
- Nom de bus
- Nom de domaine
- Etat facultatif de la fonction
- Chemin de périphérique

Cette sortie, `ldm list-io -l`, montre que `VF0` et `VF1` sont assignés au domaine `primary` et que `VF2` et `VF3` sont assignés au domaine d'E/S `iodom1`.

```
primary# ldm list-io -l /SYS/MB/RISER1/PCIE4/IOVIB.PF0
```

```
NAME                                      TYPE   BUS      DOMAIN   STATUS
----                                      ----   ---      ------   ------
/SYS/MB/RISER1/PCIE4/IOVIB.PF0            PF     pci_0    primary
[pci@400/pci@1/pci@0/pci@0/pciex15b3,673c@0]
    maxvfs = 64
/SYS/MB/RISER1/PCIE4/IOVIB.PF0.VF0        VF     pci_0    primary
[pci@400/pci@1/pci@0/pci@0/pciex15b3,673c@0,1]
/SYS/MB/RISER1/PCIE4/IOVIB.PF0.VF1        VF     pci_0    primary
[pci@400/pci@1/pci@0/pci@0/pciex15b3,673c@0,2]
/SYS/MB/RISER1/PCIE4/IOVIB.PF0.VF2        VF     pci_0    iodom1
[pci@400/pci@1/pci@0/pci@0/pciex15b3,673c@0,3]
/SYS/MB/RISER1/PCIE4/IOVIB.PF0.VF3        VF     pci_0    iodom1
[pci@400/pci@1/pci@0/pci@0/pciex15b3,673c@0,4]
```

Identification des fonctions SR-IOV InfiniBand

Cette section indique comment identifier les périphériques SR-IOV InfiniBand.

Servez-vous de la commande `ldm list-io -l` pour afficher le nom du chemin d'accès de périphérique Oracle Solaris associé à chaque fonction physique et virtuelle.

```
primary# ldm list-io -l /SYS/MB/RISER1/PCIE4/IOVIB.PF0
NAME                                      TYPE   BUS      DOMAIN   STATUS
----                                      ----   ---      ------   ------
/SYS/MB/RISER1/PCIE4/IOVIB.PF0            PF     pci_0    primary
[pci@400/pci@1/pci@0/pci@0/pciex15b3,673c@0]
    maxvfs = 64
/SYS/MB/RISER1/PCIE4/IOVIB.PF0.VF0        VF     pci_0    primary
[pci@400/pci@1/pci@0/pci@0/pciex15b3,673c@0,1]
/SYS/MB/RISER1/PCIE4/IOVIB.PF0.VF1        VF     pci_0    primary
[pci@400/pci@1/pci@0/pci@0/pciex15b3,673c@0,2]
/SYS/MB/RISER1/PCIE4/IOVIB.PF0.VF2        VF     pci_0    iodom1
[pci@400/pci@1/pci@0/pci@0/pciex15b3,673c@0,3]
/SYS/MB/RISER1/PCIE4/IOVIB.PF0.VF3        VF     pci_0    iodom1
[pci@400/pci@1/pci@0/pci@0/pciex15b3,673c@0,4]
```

Exécutez la commande `dladm show-phys -L` pour faire correspondre chaque instance IPoIB (IP over InfiniBand) à sa carte physique. Par exemple, la commande suivante met en évidence les instances IPoIB qui utilisent la carte se trouvant dans l'emplacement `PCIE4`, qui est la même carte que celle de l'exemple `ldm list-io -l` ci-dessus.

```
primary# dladm show-phys -L | grep PCIE4
net5              ibp0         PCIE4/PORT1
net6              ibp1         PCIE4/PORT2
net19             ibp8         PCIE4/PORT1
net9              ibp9         PCIE4/PORT2
net18             ibp4         PCIE4/PORT1
net11             ibp5         PCIE4/PORT2
```

Chaque adaptateur de canal hôte (HCA, Host Channel Adapter) InfiniBand possède un ID unique, appelé GUID. Chaque port possède également un GUID (en général, un HCA est doté de deux ports). Un GUID de HCA InfiniBand identifie l'adaptateur de façon unique. Les GUID des ports identifient chaque port de HCA de façon unique et jouent un rôle semblable à celui de

l'adresse MAC d'un périphérique réseau. Ces GUID à 16 chiffres sont utilisés par les outils de gestion et de diagnostic InfiniBand.

Exécutez la commande `dladm show-ib` pour obtenir les informations de GUID des périphériques SR-IOV InfiniBand. Les fonctions physiques et virtuelles d'un même périphérique ont des valeurs de GUID de HCA proches. Le 11e chiffre hexadécimal du GUID de HCA indique la relation entre une fonction physique et ses fonctions virtuelles. Notez que les zéros de tête sont supprimés dans les colonnes HCAGUID et PORTGUID.

Par exemple, la fonction physique `PF0` est accompagnée de deux fonctions virtuelles, `VF0` et `VF1`, qui sont affectées au domaine `primary`. Le 11e chiffre hexadécimal de chaque fonction virtuelle est incrémenté de un par rapport à celui de la fonction physique correspondante. Ainsi, si le GUID de `PF0` est `8`, les GUID de `VF0` et de `VF1` sont respectivement `9` et `A`.

La sortie de la commande `dladm show-ib` ci-après indique que les liens `net5` et `net6` appartiennent à la fonction physique `PF0`. Les liens `net19` et `net9` appartiennent à `VF0` du même périphérique tandis que `net18` et `net11` appartiennent à `VF1`.

```
primary# dladm show-ib
LINK         HCAGUID         PORTGUID        PORT STATE  PKEYS
net6         2128**0**001A17F56  21280001A17F58  2    up     FFFF
net5         2128**0**001A17F56  21280001A17F57  1    up     FFFF
net19        2129**0**001A17F56  14050000000001  1    up     FFFF
net9         2129**0**001A17F56  14050000000008  2    up     FFFF
net18        212**A**0001A17F56  14050000000002  1    up     FFFF
net11        212**A**0001A17F56  14050000000009  2    up     FFFF
```

Le périphérique de la sortie `dladm show-phys` suivante indique la relation entre les liens et les périphériques de port InfiniBand sous-jacents (`ibp`*X*).

```
primary# dladm show-phys
LINK              MEDIA                STATE      SPEED  DUPLEX    DEVICE
...
net6              Infiniband           up         32000  unknown   ibp1
net5              Infiniband           up         32000  unknown   ibp0
net19             Infiniband           up         32000  unknown   ibp8
net9              Infiniband           up         32000  unknown   ibp9
net18             Infiniband           up         32000  unknown   ibp4
net11             Infiniband           up         32000  unknown   ibp5
```

Exécutez la commande `ls -l` pour afficher les chemins d'accès de périphérique des ports InfiniBand (IB port). Un périphérique IB port est subordonné à un chemin d'accès de périphérique visible dans la sortie de la commande `ldm list-io -l`. Une fonction physique possède une adresse d'unité en une partie, par exemple `pciex15b3,673c@0`, tandis que les fonctions virtuelles ont des adresses d'unité en deux parties, par exemple `pciex15b3,1002@0,2`. La valeur de la seconde partie de l'adresse d'unité est égale au numéro de la fonction virtuelle additionné de un. (Dans ce cas, le deuxième composant est `2`, ce qui signifie que ce périphérique correspond à la fonction virtuelle `1`.) La sortie suivante indique que `/dev/ibp0` est une fonction physique et `/dev/ibp5` une fonction virtuelle.

```
primary# ls -l /dev/ibp0
lrwxrwxrwx   1 root     root          83 Apr 18 12:02 /dev/ibp0 ->
```

```
../devices/pci@400/pci@1/pci@0/pci@0/pciex15b3,673c@0/hermon@0/ibport@1,0,ipib:ibp0
primary# ls -l /dev/ibp5
lrwxrwxrwx   1 root     root          85 Apr 22 23:29 /dev/ibp5 ->
../devices/pci@400/pci@1/pci@0/pci@0/pciex15b3,1002@0,2/hermon@3/ibport@2,0,ipib:ibp5
```

Vous pouvez utiliser la commande `ibv_devices` pour afficher le nom du périphérique OpenFabrics ainsi que le GUID du noeud (HCA). Lorsque des fonctions virtuelles sont présentes, la colonne Type indique si la fonction est physique ou virtuelle.

```
primary# ibv_devices
device                 node GUID            type
------              ----------------        ----
mlx4_4              0002c90300a38910         PF
mlx4_5              0021280001a17f56         PF
mlx4_0              0002cb0300a38910         VF
mlx4_1              0002ca0300a38910         VF
mlx4_2              00212a0001a17f56         VF
mlx4_3              0021290001a17f56         VF
```

Utilisation des fonctions virtuelles SR-IOV Fibre Channel

Un adaptateur de bus hôte (HBA) SR-IOV Fibre Channel peut posséder un ou plusieurs ports qui apparaissent chacun comme fonction SR-IOV physique. Vous pouvez identifier les fonctions Fibre Channel physiques grâce à la chaîne `IOVFC` dans son nom de périphérique.

Chaque fonction Fibre Channel physique possède des valeurs uniques de nom universel (WWN) de port et de noeud fournies par le fabricant de cartes. Lorsque vous créez des fonctions virtuelles à partir d'une fonction physique Fibre Channel, les fonctions virtuelles se comportent comme un périphérique HBA Fibre Channel. Chaque fonction virtuelle doit avoir une identité unique spécifiée par le nom universel de port et de noeud de la structure SAN. Vous pouvez utiliser Logical Domains Manager pour attribuer automatiquement ou manuellement les noms universels de port et de noeud. En attribuant vos propres valeurs, vous pouvez entièrement contrôler l'identité de toute fonction virtuelle.

Les fonctions virtuelles HBA de Fibre Channel utilisent la méthode de Virtualisation de l'ID N_Port (NPIV) pour se connecter à la structure SAN. La configuration requise NPIV vous oblige à connecter le port HBA Fibre Channel à un commutateur Fibre Channel compatible avec NPIV. Les fonctions virtuelles sont entièrement gérées par le matériel ou le microprogramme de la carte SR-IOV. A part ces exceptions, les fonctions virtuelles Fibre Channel fonctionnent et se comportent de la même manière qu'un périphérique HBA Fibre Channel non compatible avec SR-IOV. Les fonctions virtuelles SR-IOV possèdent les mêmes capacités que les périphériques non compatibles avec SR-IOV. Tous les types de périphériques de stockage SAN sont donc pris en charge par toutes les configurations.

Les valeurs uniques de nom universel de port et de noeud des fonctions virtuelles permettent à un administrateur SAN d'attribuer du stockage aux fonctions virtuelles de la même manière que

pour un port HBA Fibre Channel non compatible avec SR-IOV. Cette gestion inclut le zonage, le masquage LUN et la qualité de service (QoS). Vous pouvez configurer le stockage afin qu'il soit exclusivement accessible à un domaine logique spécifique sans être visible par la fonction physique dans le domaine root.

Vous pouvez utiliser les méthodes SR-IOV statiques et dynamiques pour gérer les périphériques SR-IOV Fibre Channel.

Configuration matérielle requise pour SR-IOV Fibre Channel

Pour plus d'informations sur le matériel PCIe Fibre Channel SR-IOV requis, reportez-vous à la section "Configuration matérielle et logicielle requise pour SR-IOV" à la page 84.

- **Domaine de contrôle.**
  - **Cartes QLogic.** Au moins le SE Oracle Solaris 11.2.
  - **Cartes Emulex.** Au moins le Oracle Solaris 11.1 SRU 17.
- **Domaine d'E/S.**
  - **Cartes QLogic.** Au moins le SE Oracle Solaris 11.2.
  - **Cartes Emulex.** Au moins le Oracle Solaris 11.1 SRU 17.

Configuration matérielle requise et restrictions pour SR-IOV Fibre Channel

La fonction SR-IOV Fibre Channel dispose des recommandations et restrictions suivantes :

- La carte SR-IOV doit exécuter la dernière version de microprogramme qui prend en charge la fonction SR-IOV.
- La carte Fibre Channel PCIe doit être connectée à un commutateur Fibre Channel qui prend en charge NPIV et qui est compatible avec la carte PCIe.
- Le Logical Domains Manager génère automatiquement les valeurs de propriété `port-wwn` et `node-wwn` unique en connectant les domaines de contrôle de tous les systèmes à la même structure SAN et en faisant partie du même domaine multidiffusion.

  Si vous ne pouvez pas configurer cet environnement, vous devez fournir les valeurs `node-wwn` et `port-wwn` manuellement lorsque vous créez la fonction virtuelle. Ce comportement permet de s'assurer qu'il n'y a aucun conflit de nom. Reportez-vous à la section "Allocation d'un nom universel pour les fonctions virtuelles Fibre Channel" à la page 132.

Propriétés du périphérique Fibre Channel en fonction de la classe

Vous pouvez utiliser les commandes `ldm create-vf` ou `ldm set-io` pour définir les propriétés suivantes de la fonction Fibre Channel virtuelle :

`bw-percent` Indiquez le pourcentage de bande passante à allouer à la fonction virtuelle Fibre Channel. Les valeurs valides sont comprises entre 0 et 100. La valeur totale de bande passante attribuée aux fonctions virtuelles d'une fonction Fibre Channel physique ne peut pas dépasser 100. La valeur par défaut est de 0 pour que la fonction virtuelle obtienne une part équitable de la bande passante qui n'est pas encore réservée par d'autres fonctions virtuelles qui partagent la même fonction virtuelle.

`node-wwn` Indiquez le nom universel de noeud (WWN) pour la fonction virtuelle Fibre Channel. Les valeurs valides ne sont pas égales à zéro. Par défaut, cette valeur est automatiquement allouée. Si vous indiquez manuellement la valeur, vous devez également indiquer une valeur pour la propriété `port-wwn`. Pour plus d'informations, reportez-vous à la section "Allocation d'un nom universel pour les fonctions virtuelles Fibre Channel" à la page 132.

`port-wwn` Indiquez le nom universel de port (WWN) pour la fonction virtuelle Fibre Channel. Les valeurs valides ne sont pas égales à zéro. Par défaut, cette valeur est automatiquement allouée. Si vous indiquez manuellement la valeur, vous devez également indiquer une valeur pour la propriété `node-wwn`. Pour plus d'informations, reportez-vous à la section "Allocation d'un nom universel pour les fonctions virtuelles Fibre Channel" à la page 132.

Vous ne pouvez pas modifier les valeurs de propriété `node-wwn` ou `port-wwn` si vous utilisez la fonction Fibre Channel virtuelle. Vous pouvez cependant modifier la valeur de propriété `bw-percent` de manière dynamique même lorsque vous utilisez la fonction virtuelle Fibre Channel.

Allocation d'un nom universel pour les fonctions virtuelles Fibre Channel

Le Logical Domains Manager prend en charge l'allocation automatique et manuelle de noms universels pour les fonctions virtuelles Fibre Channel.

Allocation automatique d'un nom universel

Logical Domains Manager alloue une adresse MAC unique depuis son pool automatique d'allocation d'adresses MAC et crée des valeurs de propriété `node-wwn` et `port-wwn` au format IEEE.

```
port-wwn = 10:00:XX:XX:XX:XX:XX:XX
node-wwn = 20:00:XX:XX:XX:XX:XX:XX
```

XX:*XX*:*XX*:*XX*:*XX*:*XX* est l'adresse MAC automatiquement allouée.

Cette méthode d'allocation automatique produit des noms universels uniques lorsque les domaines de contrôle de tous les systèmes connectés à la même structure Fibre Channel sont également connectés par un réseau Ethernet et font partie du même domaine de multidiffusion. Si vous ne pouvez pas satisfaire ces exigences, vous devez attribuer des noms universels uniques manuellement, ce qui est requis sur le réseau SAN.

Allocation manuel d'un nom universel

Vous pouvez construire des noms universels uniques en utilisant n'importe quelle méthode. Cette section décrit comment créer des noms universels à partir du pool d'allocation manuelle d'adresses MAC Logical Domains Manager. Vous devez garantir l'unicité des noms universels que vous allouez.

Logical Domains Manager possède un pool de 256 000 adresses MAC disponibles à l'allocation manuelle sur la plage `00:14:4F:FC:00:00` - `00:14:4F:FF:FF:FF`.

L'exemple suivant affiche les valeurs de propriété `port-wwn` et `node-wwn` en fonction de l'adresse MAC `00:14:4F:FC:00:01` :

```
port-wwn = 10:00:00:14:4F:FC:00:01
node-wwn = 20:00:00:14:4F:FC:00:01
```

`00:14:4F:FC:00:01` est l'adresse MAC allouée manuellement. Pour plus d'informations à propos de l'allocation d'adresses MAC automatiques, reportez-vous au manuel "Assignation automatique et manuelle des adresses MAC" à la page 270.

Remarque - Il est recommandé d'attribuer manuellement les noms universels pour assurer une configuration prévisible du stockage SAN.

Vous devez utiliser la méthode d'allocation manuelle de noms universels lorsque tous les systèmes ne sont pas connectés au même domaine de multidiffusion par réseau Ethernet. Vous pouvez également utiliser cette méthode pour vous assurer que les mêmes noms universels sont utilisés lorsque les fonctions virtuelles Fibre Channel sont détruites et recréées.

Création de fonctions virtuelles SR-IOV Fibre Channel

Cette section décrit la création et la destruction dynamiques de fonctions virtuelles. Si vous ne pouvez pas utiliser les méthodes dynamiques pour exécuter ces actions, lancez une reconfiguration retardée sur le domaine root avant de créer ou de détruire des fonctions virtuelles.

▼ Procédure de création d'une fonction virtuelle SR-IOV Fibre Channel

Si vous ne pouvez pas utiliser cette méthode dynamique, utilisez plutôt la méthode statique. Voir la section "SR-IOV statique" à la page 89.

1. **Identifiez le périphérique de fonction physique.**

   ```
   primary# ldm list-io
   ```

   Notez que le nom de la fonction physique inclut les informations d'emplacement de la carte SR-IOV PCIe ou du périphérique intégré.

2. **Effectuez uniquement cette étape si la virtualisation d'E/S n'est pas déjà activée pour le bus qui a la fonction physique.**

   Effectuez uniquement cette étape si la virtualisation d'E/S n'est pas déjà activée pour le bus qui a la fonction physique.

   Voir la section "Procédure d'activation de la virtualisation d'E/S pour un bus PCIe" à la page 93.

3. **Vous pouvez créer une fonction virtuelle unique ou plusieurs fonctions virtuelles à partir d'une fonction physique, de manière dynamique ou statique.**

   Après avoir créé une ou plusieurs fonctions virtuelles, vous pouvez les attribuer à un domaine invité.

   - **Méthode dynamique :**

     - **Pour créer plusieurs fonctions virtuelles à partir d'une fonction physique en une seule fois, utilisez la commande suivante :**

       ```
       primary# ldm create-vf -n number | max pf-name
       ```

       Utilisez la commande `ldm create-vf -n max` pour créer en une fois toutes les fonctions virtuelles de cette fonction physique. Cette commande alloue automatiquement

les noms universels de port et de noeud pour chaque fonction virtuelle et définit la propriété `bw-percent` sur `0`, la valeur par défaut. Cette valeur indique qu'une part équitable de la bande passante est allouée à toutes les fonctions virtuelles.

Astuce - Créez des fonctions virtuelles pour la fonction physique en une seule fois. Si vous souhaitez attribuer manuellement les noms universels, créez tout d'abord toutes les fonctions virtuelles puis utilisez la commande `ldm set-io` pour attribuer manuellement vos valeurs de nom universel à chaque fonction virtuelle. Cette technique permet de minimiser le nombre de transitions d'état lors de la création de fonctions virtuelles à partir d'une fonction physique.

Vous pouvez spécifier des fonctions virtuelles soit par le biais de leur nom de chemin d'accès, soit par leur pseudonyme. Toutefois, la pratique recommandée est d'utiliser le pseudonyme.

- **Pour créer une fonction virtuelle à partir d'une fonction physique, utilisez la commande suivante :**

  `ldm create-vf [bw-percent=`*value*`] [port-wwn=`*value* `node-wwn=`*value*`]` *pf-name*

  Vous pouvez également indiquer manuellement les valeurs de propriété Fibre Channel en fonction de la classe.

Remarque - Parfois, une fonction virtuelle nouvellement créée n'est pas disponible pour une utilisation immédiate, car le SE recherche toujours les périphériques IOV. Utilisez la commande `ldm list-io` pour déterminer si la fonction physique parent et ses fonctions virtuelles enfant ont la valeur `INV` dans la colonne Statut. Si cette valeur s'affiche, attendez que la sortie`ldm list-io` n'affiche plus la valeur `INV` dans la colonne Statut (environ 45 secondes) avant d'utiliser cette fonction physique ou l'une de ses fonctions enfant. Si ce statut persiste, cela signifie qu'il y a un problème avec le périphérique.

Le statut de l'appareil pourra être `INV` immédiatement après une réinitialisation du domaine root (y compris celle du domaine `primary`) ou immédiatement après avoir utilisé la commande `ldm create-vf` ou `ldm destroy-vf`.

- **Méthode statique :**

  a. **Lancez une reconfiguration retardée.**

  ```
  primary# ldm start-reconf root-domain-name
  ```

  b. **Vous pouvez créer une fonction virtuelle unique ou plusieurs fonctions virtuelles à partir d'une fonction physique.**

Utilisez les mêmes commandes affichées précédemment pour créer les fonctions virtuelles de manière dynamique.

c. Réinitialisez le domaine root.

- **Procédure de réinitialisation du domaine root non-`primary` :**

  ```
  primary# ldm stop-domain -r root-domain
  ```

- **Procédure de réinitialisation du domaine root `primary` :**

  ```
  primary# shutdown -i6 -g0 -y
  ```

Exemple 22 Affichage des informations relatives à la fonction Fibre Channel physique

Cet exemple affiche des informations sur la fonction physique `/SYS/MB/PCIE7/IOVFC.PF0` :

- Cette fonction physique vient d'une carte dans l'emplacement PCIe `PCIE7`.
- La chaîne `IOVFC` indique que la fonction physique est un périphérique SR-IOV Fibre Channel.

```
primary# ldm list-io
NAME                                    TYPE   BUS      DOMAIN   STATUS
----                                    ----   ---      ------   ------
pci_0                                   BUS    pci_0    primary  IOV
pci_1                                   BUS    pci_1    rootdom1 IOV
niu_0                                   NIU    niu_0    primary
niu_1                                   NIU    niu_1    primary
/SYS/MB/PCIE0                           PCIE   pci_0    primary  OCC
/SYS/MB/PCIE2                           PCIE   pci_0    primary  OCC
/SYS/MB/PCIE4                           PCIE   pci_0    primary  OCC
/SYS/MB/PCIE6                           PCIE   pci_0    primary  EMP
/SYS/MB/PCIE8                           PCIE   pci_0    primary  EMP
/SYS/MB/SASHBA                          PCIE   pci_0    primary  OCC
/SYS/MB/NET0                            PCIE   pci_0    primary  OCC
/SYS/MB/PCIE1                           PCIE   pci_1    rootdom1 OCC
/SYS/MB/PCIE3                           PCIE   pci_1    rootdom1 OCC
/SYS/MB/PCIE5                           PCIE   pci_1    rootdom1 OCC
/SYS/MB/PCIE7                           PCIE   pci_1    rootdom1 OCC
/SYS/MB/PCIE9                           PCIE   pci_1    rootdom1 OCC
/SYS/MB/NET2                            PCIE   pci_1    rootdom1 OCC
/SYS/MB/NET0/IOVNET.PF0                 PF     pci_0    primary
/SYS/MB/NET0/IOVNET.PF1                 PF     pci_0    primary
/SYS/MB/PCIE5/IOVNET.PF0                PF     pci_1    rootdom1
/SYS/MB/PCIE5/IOVNET.PF1                PF     pci_1    rootdom1
/SYS/MB/PCIE7/IOVFC.PF0                 PF     pci_1    rootdom1
/SYS/MB/PCIE7/IOVFC.PF1                 PF     pci_1    rootdom1
/SYS/MB/NET2/IOVNET.PF0                 PF     pci_1    rootdom1
/SYS/MB/NET2/IOVNET.PF1                 PF     pci_1    rootdom1
```

La commande suivante fournit plus d'informations sur la fonction physique spécifiée. La valeur `maxvfs` indique le nombre maximal de fonctions virtuelles pris en charge par le périphérique.

```
primary# ldm list-io -l /SYS/MB/PCIE7/IOVFC.PF0
```

```
NAME                                      TYPE   BUS      DOMAIN   STATUS
----                                      ----   ---      ------   ------
/SYS/MB/PCIE7/IOVFC.PF0                   PF     pci_0    rootdom1
[pci@400/pci@1/pci@0/pci@6/SUNW,emlxs@0]
    maxvfs = 8
```

Exemple 23 Création dynamique d'une fonction Fibre Channel virtuelle sans paramétrage des propriétés facultatives

Cet exemple crée une fonction virtuelle de façon dynamique sans définir de propriété facultative. Dans ce cas, la commande `ldm create-vf` alloue automatiquement le pourcentage de bande passante par défaut, ainsi que les valeurs de nom universel de port et de noeud.

Assurez-vous que la virtualisation d'E/S est activée sur le bus PCIe `pci_1`. Voir la section "Procédure d'activation de la virtualisation d'E/S pour un bus PCIe" à la page 93.

Vous pouvez utiliser la commande `ldm create-vf` pour créer toutes les fonctions virtuelles à partir de la fonction physique `/SYS/MB/PCIE7/IOVFC.PF0`.

```
primary# ldm create-vf -n max /SYS/MB/PCIE7/IOVFC.PF0
Created new vf: /SYS/MB/PCIE7/IOVFC.PF0.VF0
Created new vf: /SYS/MB/PCIE7/IOVFC.PF0.VF1
Created new vf: /SYS/MB/PCIE7/IOVFC.PF0.VF2
Created new vf: /SYS/MB/PCIE7/IOVFC.PF0.VF3
Created new vf: /SYS/MB/PCIE7/IOVFC.PF0.VF4
Created new vf: /SYS/MB/PCIE7/IOVFC.PF0.VF5
Created new vf: /SYS/MB/PCIE7/IOVFC.PF0.VF6
Created new vf: /SYS/MB/PCIE7/IOVFC.PF0.VF7
```

Exemple 24 Création dynamique d'une fonction Fibre Channel virtuelle avec paramétrage des propriétés

Cet exemple crée une fonction virtuelle de manière dynamique tout en définissant la valeur de propriété `bw-percent` sur `25` et indique les noms universels de port et de noeud.

```
primary# ldm create-vf port-wwn=10:00:00:14:4F:FC:00:01 \
node-wwn=20:00:00:14:4F:FC:00:01 bw-percent=25 /SYS/MB/PCIE7/IOVFC.PF0
```

Exemple 25 Création statique d'une fonction Fibre Channel virtuelle sans paramétrage des propriétés facultatives

Cet exemple crée une fonction virtuelle de façon statique sans définir de propriété facultative. Dans ce cas, la commande `ldm create-vf` alloue automatiquement le pourcentage de bande passante par défaut, ainsi que les valeurs de nom universel de port et de noeud.

Lancez tout d'abord une reconfiguration retardée sur le domaine `rootdom1`. Activez ensuite la virtualisation d'E/S sur le bus PCIe `pci_1`. Le bus `pci_1` ayant déjà été assigné au domaine root `rootdom1`, vous devez utiliser la commande `ldm set-io` pour activer la virtualisation d'E/S.

```
primary# ldm start-reconf rootdom1
Initiating a delayed reconfiguration operation on the rootdom1 domain.
```

```
All configuration changes for other domains are disabled until the rootdom1
domain reboots, at which time the new configuration for the rootdom1 domain
will also take effect.

primary# ldm set-io iov=on pci_1
```

Vous pouvez ensuite utiliser la commande `ldm create-vf` pour créer toutes les fonctions virtuelles à partir de la fonction physique `/SYS/MB/PCIE7/IOVFC.PF0`.

```
primary# ldm create-vf -n max /SYS/MB/PCIE7/IOVFC.PF0

------------------------------------------------------------------------------
Notice: The rootdom1 domain is in the process of a delayed reconfiguration.
Any changes made to the rootdom1 domain will only take effect after it reboots.
------------------------------------------------------------------------------

Created new vf: /SYS/MB/PCIE7/IOVFC.PF0.VF0
Created new vf: /SYS/MB/PCIE7/IOVFC.PF0.VF1
Created new vf: /SYS/MB/PCIE7/IOVFC.PF0.VF2
Created new vf: /SYS/MB/PCIE7/IOVFC.PF0.VF3
Created new vf: /SYS/MB/PCIE7/IOVFC.PF0.VF4
Created new vf: /SYS/MB/PCIE7/IOVFC.PF0.VF5
Created new vf: /SYS/MB/PCIE7/IOVFC.PF0.VF6
Created new vf: /SYS/MB/PCIE7/IOVFC.PF0.VF7
```

Enfin, réinitialisez le domaine root `rootdom1` pour appliquer les modifications d'une des manières suivantes :

- `rootdom1` est un domaine root non-`primary`

  ```
  primary# ldm stop-domain -r rootdom1
  ```

- `rootdom1` est le domaine `primary`

  ```
  primary# shutdown -i6 -g0 -y
  ```

Destruction de fonctions virtuelles SR-IOV Fibre Channel

Vous pouvez détruire une fonction virtuelle si elle n'est pas affectée à un domaine. Vous pouvez uniquement détruire une fonction virtuelle dans l'ordre inverse de la création en d'autres termes, seule la dernière fonction virtuelle créée peut être détruite. La configuration résultante est validée par le pilote de la fonction physique.

▼ Procédure de destruction d'une fonction virtuelle SR-IOV Fibre Channel

Si vous ne pouvez pas utiliser cette méthode dynamique, utilisez plutôt la méthode statique. Voir la section "SR-IOV statique" à la page 89.

1. **Identifiez le périphérique de fonction physique.**

   ```
   primary# ldm list-io
   ```

2. **Vous pouvez détruire une fonction virtuelle unique ou plusieurs fonctions virtuelles de manière dynamique ou statique.**

   - **Méthode dynamique :**

     - **Pour détruire toutes les fonctions virtuelles à partir d'une fonction physique en une seule fois, utilisez la commande suivante :**

       ```
       primary# ldm destroy-vf -n number | max pf-name
       ```

       Vous pouvez spécifier des fonctions virtuelles soit par le biais de leur nom de chemin d'accès, soit par leur pseudonyme. Toutefois, la pratique recommandée est d'utiliser le pseudonyme.

       Utilisez la commande `ldm destroy-vf -n max` pour détruire en une fois toutes les fonctions virtuelles de cette fonction physique.

       Si vous indiquez un *nombre* comme argument pour l'option `-n`, le dernier *nombre* des fonctions virtuelles est détruit. Utilisez cette méthode car elle exécute l'opération en une seule transition d'état de pilote de périphérique de fonction physique.

     - **Procédure de destruction d'une fonction virtuelle spécifiée :**

       ```
       primary# ldm destroy-vf vf-name
       ```

     En raison de retards du périphérique matériel affecté au SE, la fonction physique affectée et toutes les autres fonctions virtuelles enfant risquent de ne plus être disponibles pour une utilisation immédiate. Utilisez la commande `ldm list-io` pour déterminer si la fonction physique parent et ses fonctions virtuelles enfant ont la valeur `INV` dans la colonne Statut. Si cette valeur s'affiche, attendez que la sortie`ldm list-io` n'affiche plus `INV` dans la colonne Statut (environ 45 secondes). A ce moment, vous pouvez utiliser cette fonction physique en toute sécurité, ou n'importe quelle fonction virtuelle enfant. Si ce statut persiste, cela signifie qu'il y a un problème avec le périphérique.

     Le statut de l'appareil pourra être `INV` immédiatement après une réinitialisation du domaine root (y compris celle du domaine `primary`) ou immédiatement après avoir utilisé la commande `ldm create-vf` ou `ldm destroy-vf`.

   - **Méthode statique :**

     a. **Lancez une reconfiguration retardée.**

        ```
        primary# ldm start-reconf root-domain-name
        ```

b. Vous pouvez détruire une fonction virtuelle unique ou plusieurs fonctions virtuelles.

- **Pour détruire toutes les fonctions virtuelles à partir d'une fonction physique spécifique en une seule fois, utilisez la commande suivante :**

  ```
  primary# ldm destroy-vf -n number | max pf-name
  ```

  Vous pouvez spécifier des fonctions virtuelles soit par le biais de leur nom de chemin d'accès, soit par leur pseudonyme. Toutefois, la pratique recommandée est d'utiliser le pseudonyme.

- **Procédure de destruction d'une fonction virtuelle spécifiée :**

  ```
  primary# ldm destroy-vf vf-name
  ```

c. Réinitialisez le domaine root.

- **Procédure de réinitialisation du domaine root non-`primary` :**

  ```
  primary# ldm stop-domain -r root-domain
  ```

- **Procédure de réinitialisation du domaine root `primary` :**

  ```
  primary# shutdown -i6 -g0 -y
  ```

Exemple 26 Destruction dynamique de plusieurs fonctions virtuelles SR-IOV Fibre Channel

Cet exemple illustre les résultats de la destruction de toutes les fonctions virtuelles `/SYS/MB/PCIE5/IOVFC.PF1` de la fonction physique. La sortie `ldm list-io` indique que la fonction physique possède huit fonctions virtuelles. La commande `ldm destroy-vf -n max` détruit toutes les fonctions virtuelles et la dernière sortie `ldm list-io` indique qu'il ne reste aucune fonction virtuelle.

```
primary# ldm list-io
...
/SYS/MB/PCIE5/IOVFC.PF1                   PF     pci_1
/SYS/MB/PCIE5/IOVFC.PF1.VF0               VF     pci_1
/SYS/MB/PCIE5/IOVFC.PF1.VF1               VF     pci_1
/SYS/MB/PCIE5/IOVFC.PF1.VF2               VF     pci_1
/SYS/MB/PCIE5/IOVFC.PF1.VF3               VF     pci_1
/SYS/MB/PCIE5/IOVFC.PF1.VF4               VF     pci_1
/SYS/MB/PCIE5/IOVFC.PF1.VF5               VF     pci_1
/SYS/MB/PCIE5/IOVFC.PF1.VF6               VF     pci_1
/SYS/MB/PCIE5/IOVFC.PF1.VF7               VF     pci_1
primary# ldm destroy-vf -n max /SYS/MB/PCIE5/IOVFC.PF1
primary# ldm list-io
...
/SYS/MB/PCIE5/IOVFC.PF1                   PF     pci_1
```

Exemple 27 Destruction d'une fonction virtuelle Fibre Channel

Cet exemple illustre la destruction statique des fonctions virtuelles `/SYS/MB/PCIE7/IOVFC.PF0` de la fonction physique.

```
primary# ldm start-reconf rootdom1
Initiating a delayed reconfiguration operation on the rootdom1 domain.
All configuration changes for other domains are disabled until the rootdom1
domain reboots, at which time the new configuration for the rootdom1 domain
will also take effect.

primary# ldm destroy-vf -n max /SYS/MB/PCIE7/IOVFC.PF0
primary# ldm stop-domain -r rootdom1
```

Modification de fonctions virtuelles SR-IOV Fibre Channel

La commande `ldm set-io` modifie la configuration en cours d'une fonction virtuelle en modifiant les valeurs des propriétés ou en définissant de nouvelles propriétés.

Si vous ne pouvez pas utiliser cette méthode dynamique, utilisez à sa place la méthode statique. Voir la section "SR-IOV statique" à la page 89.

Vous pouvez utiliser la commande `ldm set-io` pour modifier les propriétés `bw-percent`, `port-wwn` et `node-wwn`.

Vous pouvez modifier les propriétés `bw-percent` de façon dynamique alors que les fonctions virtuelles sont assignées à un domaine.

▼ Procédures de modification des propriétés des fonctions virtuelles SR-IOV Fibre Channel

1. Identifiez le périphérique de fonction physique.

```
primary# ldm list-io
```

Notez que le nom de la fonction physique inclut les informations d'emplacement de la carte SR-IOV PCIe ou du périphérique intégré.

2. Modifiez une propriété de fonction virtuelle.

`ldm set-io [bw-percent=`*value*`] [port-wwn=`*value* `node-wwn=`*value*`]` *pf-name*

Contrairement à la valeur de propriété `bw-percent`, que vous pouvez modifier de manière dynamique à n'importe quel moment, vous pouvez uniquement modifier les valeurs de propriété

`port-wwn` et `node-wwn` de manière dynamique lorsque la fonction virtuelle n'est pas attribuée à un domaine.

Exemple 28 Modification des propriétés des fonctions virtuelles SR-IOV Fibre Channel

Cet exemple modifie les propriétés de la fonction virtuelle spécifiée, `/SYS/MB/PCIE7/IOVFC.PF0.VF0`, pour indiquer le pourcentage de bande passante et les valeurs de nom universel de port et de noeud.

```
primary# ldm set-io port-wwn=10:00:00:14:4f:fc:f4:7c \
node-wwn=20:00:00:14:4f:fc:f4:7c bw-percent=25 /SYS/MB/PCIE7/IOVFC.PF0.VF0
```

Ajout et suppression de fonctions virtuelles SR-IOV Fibre Channel sur des domaines d'E/S

▼ Procédure d'ajout d'une fonction virtuelle SR-IOV Fibre Channel à un domaine d'E/S

Si vous ne pouvez pas supprimer la fonction virtuelle de façon dynamique, utilisez la méthode statique. Voir la section "SR-IOV statique" à la page 89.

1. **Identifiez la fonction virtuelle que vous voulez ajouter à un domaine d'E/S.**

   ```
   primary# ldm list-io
   ```

2. **Ajoutez une fonction virtuelle de manière dynamique ou statique.**

   - **Procédure d'ajout dynamique d'une fonction virtuelle :**

     ```
     primary# ldm add-io vf-name domain-name
     ```

     vf-name est le pseudonyme ou le nom du chemin d'accès de la fonction virtuelle. La pratique recommandée est d'utiliser le pseudonyme. *domain-name* représente le nom du domaine auquel vous ajoutez la fonction virtuelle.

     Le nom du chemin d'accès du périphérique de la fonction virtuelle dans le domaine est le chemin indiqué dans la sortie `list-io -l`.

   - **Procédure d'ajout statique d'une fonction virtuelle :**

     a. **Arrêtez le domaine, puis ajoutez la fonction virtuelle.**

        ```
        primary# ldm stop-domain domain-name
        ```

```
primary# ldm add-io vf-name domain-name
```

vf-name est le pseudonyme ou le nom du chemin d'accès de la fonction virtuelle. La pratique recommandée est d'utiliser le pseudonyme. *domain-name* représente le nom du domaine auquel vous ajoutez la fonction virtuelle. L'invité spécifié doit avoir l'état inactif ou lié.

Le nom du chemin d'accès du périphérique de la fonction virtuelle dans le domaine est le chemin indiqué dans la sortie `list-io -l`.

b. Redémarrez le domaine.

```
primary# ldm start-domain domain-name
```

Exemple 29 Ajout d'une fonction virtuelle Fibre Channel

Cet exemple illustre l'ajout dynamique de la fonction virtuelle `/SYS/MB/PCIE7/IOVFC.PF0.VF0` au domaine `ldg2`.

```
primary# ldm add-io /SYS/MB/PCIE7/IOVFC.PF0.VF0 ldg2
```

Si vous ne pouvez pas ajouter la fonction virtuelle de manière dynamique, utilisez la méthode statique :

```
primary# ldm stop-domain ldg2
primary# ldm add-io /SYS/MB/PCIE7/IOVFC.PF0.VF0 ldg2
primary# ldm start-domain ldg2
```

▼ Procédure de suppression d'une fonction virtuelle SR-IOV Fibre Channel d'un domaine d'E/S

Si vous ne pouvez pas utiliser cette méthode dynamique, utilisez plutôt la méthode statique. Voir la section "SR-IOV statique" à la page 89.

Attention - Avant de supprimer la fonction virtuelle du domaine, assurez-vous qu'elle n'est pas indispensable à l'initialisation du domaine.

1. Identifiez la fonction virtuelle que vous souhaitez supprimer d'un domaine d'E/S.

```
primary# ldm list-io
```

2. Supprimez une fonction virtuelle de manière dynamique ou statique.

- **Procédure de suppression dynamique d'une fonction virtuelle :**

```
primary# ldm remove-io vf-name domain-name
```

vf-name est le pseudonyme ou le nom du chemin d'accès de la fonction virtuelle. La pratique recommandée est d'utiliser le pseudonyme du périphérique. *domain-name* représente le nom du domaine dans lequel vous souhaitez supprimer la fonction virtuelle.

- **Procédure de suppression statique d'une fonction virtuelle :**

  a. **Arrêtez le domaine d'E/S.**

  ```
  primary# ldm stop-domain domain-name
  ```

  b. **Supprimez la fonction virtuelle.**

  ```
  primary# ldm remove-io vf-name domain-name
  ```

  vf-name est le pseudonyme ou le nom du chemin d'accès de la fonction virtuelle. La pratique recommandée est d'utiliser le pseudonyme du périphérique. *domain-name* représente le nom du domaine dans lequel vous souhaitez supprimer la fonction virtuelle. L'invité spécifié doit avoir l'état inactif ou lié.

  c. **Démarrez le domaine d'E/S.**

  ```
  primary# ldm start-domain domain-name
  ```

Exemple 30 Suppression dynamique d'une fonction virtuelle Fibre Channel

Cet exemple illustre la suppression dynamique de la fonction virtuelle `/SYS/MB/PCIE7/IOVFC.PF0.VF0` du domaine `ldg2`.

```
primary# ldm remove-io /SYS/MB/PCIE7/IOVFC.PF0.VF0 ldg2
```

Si la commande s'exécute correctement, la fonction virtuelle est supprimée du domaine `ldg2`. Lorsque `ldg2` est redémarré, la fonction virtuelle spécifiée n'apparaît plus dans le domaine.

Si vous ne pouvez pas retirer la fonction virtuelle de manière dynamique, utilisez la méthode statique :

```
primary# ldm stop-domain ldg2
primary# ldm remove-io /SYS/MB/PCIE7/IOVFC.PF0.VF0 ldg2
primary# ldm start-domain ldg2
```

Rubriques SR-IOV avancées : SR-IOV Fibre Channel

Cette section décrit quelques-unes des rubriques avancées relatives à l'utilisation des fonctions virtuelles SR-IOV Fibre Channel.

Accès à une fonction virtuelle Fibre Channel dans un domaine invité

Le journal de console `ldg2` affiche les opérations du périphérique de fonction virtuelle Fibre Channel attribué. Utilisez la commande `fcadm` pour afficher et accéder au périphérique de fonction virtuelle Fibre Channel.

```
ldg2# fcadm hba-port
HBA Port WWN: 100000144ffb8a99
       Port Mode: Initiator
       Port ID: 13d02
       OS Device Name: /dev/cfg/c3
       Manufacturer: Emulex
       Model: 7101684
       Firmware Version: 7101684 1.1.60.1
       FCode/BIOS Version: Boot:1.1.60.1 Fcode:4.03a4
       Serial Number: 4925382+133400002R
       Driver Name: emlxs
       Driver Version: 2.90.15.0 (2014.01.22.14.50)
       Type: N-port
       State: online
       Supported Speeds: 4Gb 8Gb 16Gb
       Current Speed: 16Gb
       Node WWN: 200000144ffb8a99
       NPIV Not Supported
```

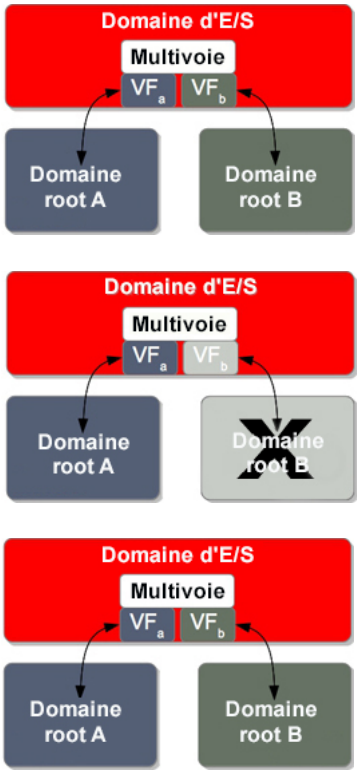
Utilisez la commande `format` pour afficher les LUN visibles.

```
ldg2# format
Searching for disks...done
AVAILABLE DISK SELECTIONS:
   0. c2d0 <Unknown-Unknown-0001-25.00GB>
      /virtual-devices@100/channel-devices@200/disk@0
   1. c3t21000024FF4C4BF8d0 <SUN-COMSTAR-1.0-10.00GB>
      /pci@340/pci@1/pci@0/pci@6/SUNW,emlxs@0,2/fp@0,0/ssd@w21000024ff4c4bf8,0
Specify disk (enter its number): ^D
ldg2#
```

Résilience de domaine d'E/S

La résilience de domaine d'E/S en cas de panne améliore la disponibilité et les performances d'un domaine d'E/S en lui permettant de continuer de fonctionner même lorsque l'un de ses domaines root associés est interrompu. Quand un domaine root est interrompu, les domaines d'E/S qui utilisent ses services continuent de fonctionner en permettant aux périphériques affectés de basculer vers l'autre chemin d'E/S. Quand le domaine root est de nouveau en service, les périphériques affectés du domaine d'E/S résilient reprennent également le service et les capacités de basculement sont restaurées.

Les diagrammes suivants décrivent ce qui se passe lorsque l'un des domaines root configuré échoue et lorsque le domaine root reprend le service.



Chaque domaine root fournit une fonction virtuelle au domaine d'E/S. Le domaine d'E/S utilise la fonctionnalité de chemins d'accès multiples IPMP pour les périphériques virtuels et la fonctionnalité multipathing d'E/S d'Oracle Solaris pour les périphériques Fibre Channel.

Lorsque le domaine root B est interrompu par une panique ou une réinitialisation, la fonction virtuelle B est suspendue dans le domaine d'E/S, puis la fonctionnalité de chemins d'accès multiples achemine toutes les E/S via le domaine root A.

Quand le domaine root B est de nouveau en service, la fonction B virtuelle reprend dans le domaine d'E/S. Le groupe de chemins d'accès multiple est restauré en redondance complète.

Dans cette configuration, la fonction virtuelle peut être un périphérique réseau virtuel ou un périphérique de stockage virtuel, ce qui signifie que le domaine d'E/S peut être configuré selon une combinaison quelconque de fonctions virtuelles ou de périphériques virtuels.

Vous pouvez créer une configuration permettant de disposer à la fois de domaines d'E/S résilients et non résilients. Voir "Example – Utilisation des configurations résilientes et non résilientes" à la page 151.

Configuration requise des domaines d'E/S résilients

Remarque - Le SE Oracle Solaris 10 n'assure pas la résilience du domaine d'E/S.

Un domaine d'E/S résilient doit répondre aux exigences suivantes :

- Exécute au moins le SE Oracle Solaris 11.2 SRU 8 et son domaine `primary` exécute au moins le logiciel Oracle VM Server for SPARC 3.2.
- Utilise la fonctionnalité de chemins d'accès multiples pour créer des configurations de basculement pour les fonctions et les périphériques virtuels. Cette configuration requiert que les fonctions virtuelles et les périphériques virtuels soient de la même catégorie : réseau ou stockage.
- A la valeur de la propriété `master` définie sur le nom d'un domaine root dont la propriété `failure-policy` est définie sur `ignore`. Toutes les autres stratégies de panne, comme `stop`, `reset` ou `panic`, remplacent la résilience d'E/S et le domaine d'E/S est interrompu.
- Utilise uniquement des fonctions virtuelles SR-IOV, des périphériques réseau virtuels et des périphériques de stockage virtuels qui prennent en charge la résilience du domaine d'E/S. Voir `https://support.oracle.com/CSP/main/article?cmd=show&type=NOT&doctype=REFERENCE&id=1325454.1`.

Limitations de la résilience de domaine d'E/S

- Lorsque vous enfichez à chaud une carte SR-IOV dans le domaine root et que vous lui affectez des fonctions virtuelles dans un domaine d'E/S, le domaine d'E/S peut ne pas fournir de résilience quand le domaine root échoue. Par conséquent, vous ne devez ajouter la carte SR-IOV que lorsque le domaine root est arrêté. Affectez ensuite les fonctions virtuelles une fois le domaine root réinitialisé.
- Si vous disposez d'un domaine d'E/S résilient et que vous affectez un périphérique de l'une des manières suivantes, alors le domaine d'E/S n'est plus résilient :
  - Ajoutez une fonction virtuelle à partir d'une carte qui ne prend pas en charge la résilience d'E/S
  - Affectez directement un périphérique à l'aide de la fonctionnalité d'E/S directe

  Dans ce cas, définissez la stratégie de panne de `ignore` à `reset` ou `stop`.

Configuration des domaines d'E/S résilients

▼ Procédure de configuration d'un domaine d'E/S résilient

Avant de commencer

N'utilisez que des cartes PCIe prenant en charge la fonctionnalité de résilience des domaines d'E/S. Voir `https://support.oracle.com/CSP/main/article?cmd=show&type=NOT&doctype=REFERENCE&id=1325454.1`.

Assurez-vous que le domaine d'E/S, le domaine root et le domaine `primary` exécutent au moins le SE Oracle Solaris 11.2 SRU 8 et le logiciel Logical Domains Manager 3.2.

1. **Sur le domaine root, définissez la propriété `failure-policy` sur `ignore`.**

   ```
   primary# ldm set-domain failure-policy=ignore root-domain-name
   ```

   Remarque - Si vous ajoutez n'importe quel dispositif au domaine d'E/S qui n'est pas pris en charge par la résilience, ce domaine n'est plus résilient. Réinitialisez alors la valeur de la propriété `failure-policy` sur `stop`, `reset` ou `panic`.

   Pour plus d'informations sur les dépendances de domaine, reportez-vous à la section "Configuration des dépendances de domaine" à la page 413.

2. **Sur le domaine d'E/S, définissez la propriété `master` sur le nom du domaine root.**

   ```
   primary# ldm set-domain master=root-domain-name I/O-domain-name
   ```

3. **Configurez la fonctionnalité de chemins d'accès multiples sur les chemins.**

   - **Ethernet. Utilisez la fonction IPMP pour configurer la fonctionnalité de chemins d'accès multiples sur les chemins.**

     Pour plus d'informations sur la fonction IPMP permettant de configurer la fonctionnalité de chemins d'accès multiples, reportez-vous à *Administering TCP/IP Networks, IPMP, and IP Tunnels in Oracle Solaris 11.3*.

   - **Fibre Channel. Utilisez la fonctionnalité multipathing d'E/S d'Oracle Solaris pour configurer la fonctionnalité de chemins d'accès multiples sur les chemins.**

     Pour plus d'informations sur la fonctionnalité multipathing d'E/S d'Oracle Solaris permettant de configurer la fonctionnalité de chemins d'accès multiples, reportez-vous à la section *Managing SAN Devices and Multipathing in Oracle Solaris 11.3*.

Exemple 31 Utiliser la fonctionnalité IPMP pour configurer la fonctionnalité de chemins d'accès multiples à l'aide des fonctions Ethernet SR-IOV

Cet exemple illustre comment utiliser la fonction IPMP pour configurer des périphériques réseau virtuels pour un domaine d'E/S résilient. Pour plus d'informations, reportez-vous à la section *Administering TCP/IP Networks, IPMP, and IP Tunnels in Oracle Solaris 11.3*.

1. Identifiez deux fonctions physiques Ethernet SR-IOV affectées à différents domaines root.

   Dans cet exemple, les domaines root `root-1` et `root-2` ont des fonctions physiques Ethernet SR-IOV.

```
primary# ldm list-io | grep root-1 | grep PF
/SYS/PCI-EM8/IOVNET.PF0                    PF     pci_1    root-1
primary# ldm list-io | grep root-2 | grep PF
/SYS/RIO/NET2/IOVNET.PF0                   PF     pci_2    root-2
```

2. Créez deux fonctions virtuelles Ethernet sur chacune des fonctions physiques indiquées.

```
primary# ldm create-vf /SYS/MB/NET0/IOVNET.PF0
Created new vf: /SYS/PCI-EM8/IOVNET.PF0.VF0
primary# ldm create-vf /SYS/RIO/NET2/IOVNET.PF0
Created new vf: /SYS/RIO/NET2/IOVNET.PF0.VF0
```

3. Affectez les fonctions virtuelles Ethernet au domaine d'E/S `io-1`.

```
primary# ldm add-io /SYS/PCI-EM8/IOVNET.PF0.VF0 io-1
primary# ldm add-io /SYS/RIO/NET2/IOVNET.PF0.VF0 io-1
```

4. Configurez les fonctions virtuelles Ethernet dans un groupe IPMP sur le domaine d'E/S.

   a. Identifiez les périphériques réseau que vous venez d'ajouter, `net1` et `net2`, sur le domaine d'E/S.

```
i0-1# dladm show-phys
LINK         MEDIA        STATE    SPPED    DUPLEX    DEVICE
net0         Ethernet     up       0        unknown   vnet0
net1         Ethernet     up       1000     full      igbvf0
net2         Ethernet     up       1000     full      igbvf1
```

   b. Créez des interfaces IP pour les nouveaux périphériques réseau.

```
io-1# ipadm create-ip net1
io-1# ipadm create-ip net2
```

   c. Créez le groupe IPMP `ipmp0` pour les deux interfaces réseau.

```
io-1# ipadm create-ipmp -i net1 -i net2 ipmp0
```

   d. Attribuez une adresse IP au groupe IPMP.

      Cet exemple illustre comment configurer l'option DHCP.

```
io-1# ipadm create-addr -T dhcp ipmp0/v4
```

   e. Vérifiez le statut de l'interface du groupe IPMP.

```
io-1# ipmpstat -g
```

Exemple 32 Utilisation de la fonctionnalité multipathing d'E/S d'Oracle Solaris pour configurer la fonctionnalité de chemins d'accès multiples avec fonctions SR-IOV Fibre Channel

Cet exemple illustre l'utilisation de la fonctionnalité multipathing d'E/S d'Oracle Solaris pour configurer des périphériques de fonction virtuelle Fibre Channel pour un domaine d'E/S

résilient. Pour plus d'informations, reportez-vous à *Managing SAN Devices and Multipathing in Oracle Solaris 11.3*.

1. Identifiez deux fonctions physiques Ethernet SR-IOV Fibre Channel affectées à différents domaines root.

   Dans cet exemple, les domaines root `root-1` et `root-2` ont des fonctions physiques SR-IOV Fibre Channel.

   ```
   primary# ldm list-io | grep root-1 | grep PF
   /SYS/PCI-EM4/IOVFC.PF0                    PF     pci_1    root-1
   primary# ldm list-io | grep root-2 | grep PF
   /SYS/PCI-EM15/IOVFC.PF0                   PF     pci_2    root-2
   ```

2. Créez deux fonctions virtuelles sur chacune des fonctions physiques indiquées.

   Pour plus d'informations, reportez-vous à la section "Procédure de création d'une fonction virtuelle SR-IOV Fibre Channel" à la page 134.

   ```
   primary# ldm create-vf port-wwn=10:00:00:14:4f:fc:60:00 \
   node-wwn=20:00:00:14:4f:fc:60:00 /SYS/PCI-EM4/IOVFC.PF0
   Created new vf: /SYS/PCI-EM4/IOVFC.PF0.VF0
   primary# ldm create-vf port-wwn=10:00:00:14:4f:fc:70:00 \
   node-wwn=20:00:00:14:4f:fc:70:00 /SYS/PCI-EM15/IOVFC.PF0
   Created new vf: /SYS/PCI-EM15/IOVFC.PF0.VF0
   ```

3. Affectez les fonctions virtuelles que vous venez de créer au domaine d'E/S `io-1`.

   ```
   primary# ldm add-io /SYS/PCI-EM4/IOVFC.PF0.VF0 io-1
   primary# ldm add-io /SYS/PCI-EM15/IOVFC.PF0.VF0 io-1
   ```

4. Déterminez si la fonctionnalité multipathing d'E/S d'Oracle Solaris est activée sur le domaine d'E/S en exécutant la commande `prtconf -v`.

   Si la sortie pour le périphérique `fp` inclut le paramètre de propriété de périphérique suivant, la fonctionnalité multipathing d'E/S d'Oracle Solaris est activée :

   ```
   mpxio-disable="no"
   ```

   Si la propriété `mpxio-disable` est définie sur `yes`, modifiez la valeur de propriété sur `no` dans le fichier `/etc/driver/drv/fp.conf`, puis réinitialisez le domaine d'E/S.

   Si la propriété `mpxio-disable` du périphérique n'apparaît pas dans la sortie `prtconf -v`, ajoutez l'entrée `mpxio-disable="no"` dans le fichier `/etc/driver/drv/fp.conf`, puis réinitialisez le domaine d'E/S.

5. Vérifiez l'état du groupe de la fonctionnalité multipathing d'E/S d'Oracle Solaris.

   ```
   io-1# mpathadm show LU

   Logical Unit:  /dev/rdsk/c0t600A0B80002A384600003D6B544EECD0d0s2
        mpath-support:  libmpscsi_vhci.so
   ```

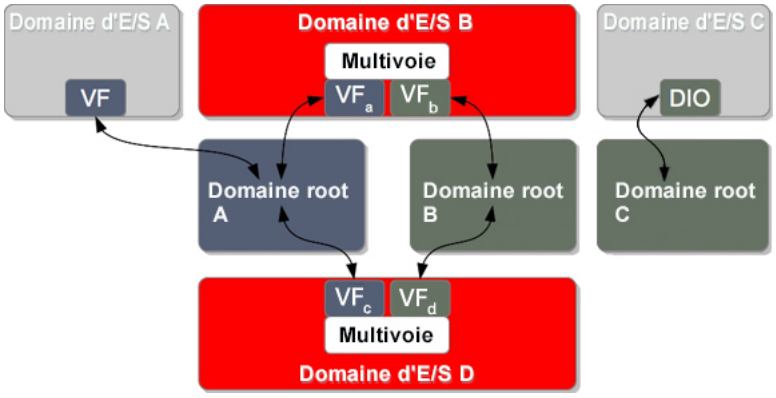
```
Vendor:  SUN
Product:  CSM200_R
Revision:  0660
Name Type:  unknown type
Name:  600a0b80002a384600003d6b544eecd0
Asymmetric:  yes
Current Load Balance:  round-robin
Logical Unit Group ID:  NA
Auto Failback:  on
Auto Probing:  NA

Paths:
        Initiator Port Name:  100000144ffc6000
        Target Port Name:  201700a0b82a3846
        Override Path:  NA
        Path State:  OK
        Disabled:  no

        Initiator Port Name:  100000144ffc7000
        Target Port Name:  201700a0b82a3846
        Override Path:  NA
        Path State:  OK
        Disabled:  no

Target Port Groups:
        ID:  1
        Explicit Failover:  yes
        Access State:  active
        Target Ports:
                Name:  201700a0b82a3846
                Relative ID:  0
```

Example – Utilisation des configurations résilientes et non résilientes

Vous pouvez utiliser des configurations avec les domaines résilients et non résilients.

Le schéma suivant montre les domaines d'E/S A et C, non résilients, du fait qu'ils n'utilisent pas la fonctionnalité de chemins d'accès multiples. Le domaine d'E/S A a une fonction virtuelle et le domaine d'E/S C est associé à un périphérique d'E/S direct.

FIGURE 4 Configuration avec des domaines d'E/S résilients et non résilients



Les domaines d'E/S B et D sont résilients. Les domaines d'E/S A, B et D dépendent du domaine root A. Les domaines d'E/S B et D dépendent du domaine root B. Le domaine d'E/S C dépend du domaine root C.

Si le domaine root A est interrompu, le domaine d'E/S l'est également. Les domaines d'E/S B et D, cependant, basculent vers d'autres et continuent à exécuter des applications. Si le domaine root C est interrompu, le domaine d'E/S C échoue, comme indiqué par la valeur de propriété `failure-policy` du domaine root C.

Réinitialisation du domaine root avec des domaines d'E/S non résilients configurés

Remarque - Si votre domaine d'E/S est résilient, il peut continuer à fonctionner, même lorsque le domaine root qui le sert est interrompu. Pour plus d'informations sur la configuration des domaines d'E/S, reportez-vous à la section "Résilience de domaine d'E/S" à la page 145.

Au sujet des emplacements PCIe dans le domaine d'E/S, les précautions évoquées dans la section "Réinitialisation du domaine root avec des extrémités PCIe configurées" à la page 162 s'appliquent également aux fonctions virtuelles assignées à un domaine d'E/S.

Remarque - Un domaine d'E/S ne peut pas démarrer si le domaine root qui lui est associé n'est pas en cours d'exécution.

♦ ♦ ♦ CHAPITRE 9

Création d'un domaine d'E/S en utilisant les E/S directes

Ce chapitre aborde les sujets suivants, relatifs aux E/S directes :

Création d'un domaine d'E/S en assignant les périphériques d'extrémité PCIe

Vous pouvez assigner un périphérique d'extrémité PCIe (ou un périphérique pouvant être associé à des E/S directes) individuel à un domaine. Cette utilisation des périphériques d'extrémité PCIe augmente la granularité de l'assignation de périphériques aux domaines d'E/S. Cette fonctionnalité est fournie au moyen de la fonction d'E/S directes (DIO).

La fonction DIO vous permet de créer un nombre de domaines d'E/S supérieur au nombre de bus PCIe dans un système. Le nombre de domaines d'E/S possibles n'est alors plus limité que par le nombre de périphériques d'extrémité PCIe.

Un périphérique d'extrémité PCIe peut être l'un des éléments suivants :

- Une carte PCIe dans un emplacement

- Un périphérique PCIe intégré identifié par la plate-forme

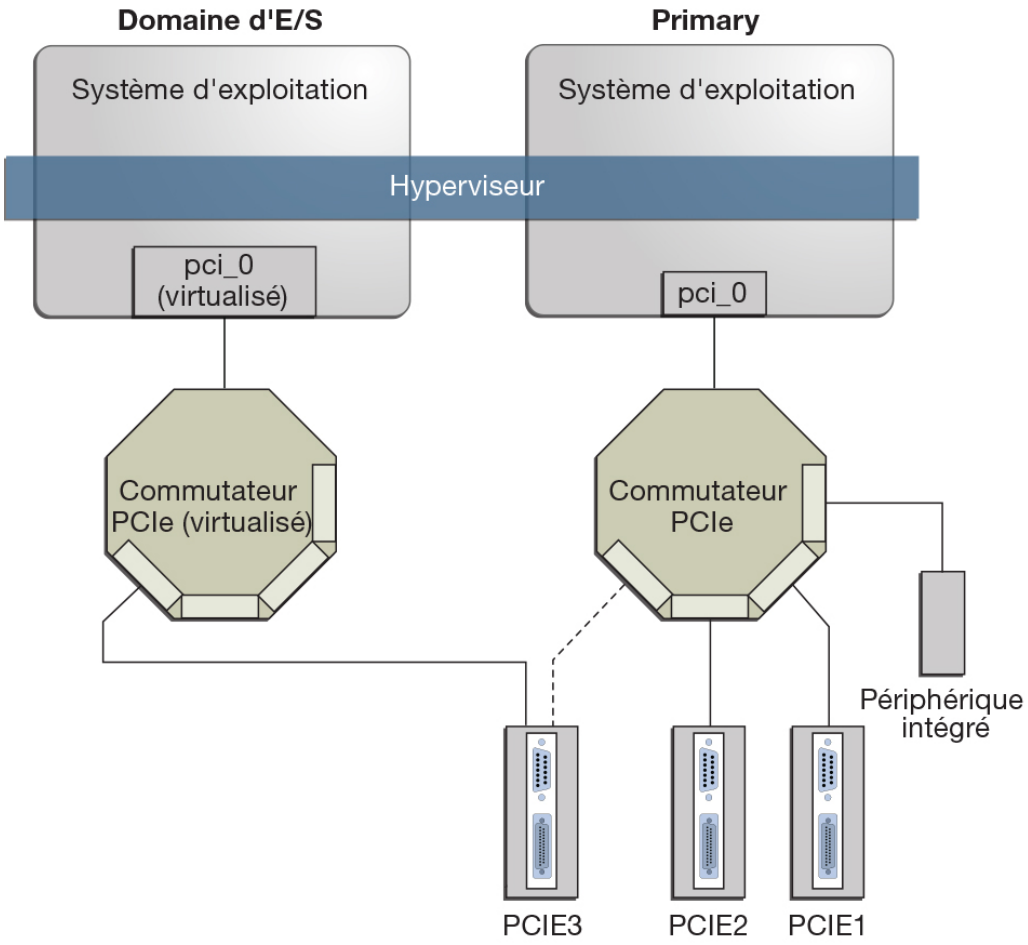
Remarque - Etant donné que les domaines root ne peuvent pas présenter de dépendances à d'autres domaines root, les périphériques d'extrémité PCIe et les fonctions virtuelles SR-IOV présents sur un bus PCIe détenu par un domaine root ne peuvent pas être assignées à un autre domaine root. En revanche, un périphérique d'extrémité PCIe ou une fonction virtuelle de bus PCIe *peuvent* être affectés au domaine root propriétaire du bus.

Le schéma suivant montre que le périphérique d'extrémité PCIe `PCIE3` est assigné à un domaine d'E/S. Le bus `pci_0` et le commutateur du domaine d'E/S sont virtuels. Le périphérique d'extrémité `PCIE3` n'est plus accessible dans le domaine `primary`.

Dans le domaine d'E/S, le bloc `pci_0` et le commutateur constituent respectivement un complexe root virtuel et un commutateur PCIe virtuel. Ce bloc et ce commutateur sont semblables au bloc `pci_0` et au commutateur du domaine `primary`. Dans le domaine `primary`, les périphériques situés dans l'emplacement `PCIE3` sont en quelque sorte des “doubles” des périphériques d'origine et sont identifiés par `SUNW,assigned`.

Attention - Vous ne pouvez pas recourir aux opérations d'enfichage à chaud d'Oracle Solaris pour retirer à chaud un périphérique d'extrémité PCIe lorsque celui-ci a été supprimé du domaine `primary` à l'aide de la commande `ldm remove-io`. Pour plus d'informations sur le remplacement ou la suppression d'un périphérique d'extrémité PCIe, reportez-vous à la section "Procédure de modification matérielle PCIe" à la page 164.

FIGURE 5 Assignation d'un périphérique d'extrémité PCIe à un domaine d'E/S



Utilisez la commande `ldm list-io` pour répertorier les périphériques d'extrémité PCIe.

Bien que la fonction DIO permette à toute carte PCIe d'un emplacement d'être assignée à un domaine d'E/S, seules certaines cartes PCIe sont prises en charge. Voir la section "Configuration matérielle et logicielle requise pour les E/S directes" à la page 158.

Attention - Les cartes PCIe comportant un pont ne sont pas prises en charge. L'assignation au niveau de la fonction PCIe n'est pas prise en charge. L'assignation d'une carte PCIe non prise en charge à un domaine d'E/S peut provoquer un comportement imprévisible.

Les points suivants livrent des informations importantes à propos de la fonction DIO :

- Cette fonction n'est activée que lorsque la configuration logicielle requise est respectée. Voir la section "Configuration matérielle et logicielle requise pour les E/S directes" à la page 158.
- Seules les extrémités PCIe connectées à un bus PCIe assigné à un domaine root peuvent être assignées à un autre domaine à l'aide de la fonction DIO.
- Les domaines d'E/S utilisant la fonction DIO ont uniquement accès aux périphériques d'extrémité PCIe lorsque le domaine root est en cours d'exécution.
- La réinitialisation du domaine root a une incidence sur les domaines d'E/S qui possèdent des périphériques d'extrémité PCIe. Voir la section "Réinitialisation du domaine root avec des extrémités PCIe configurées" à la page 162. Le domaine root effectue également les tâches suivantes :
  - Initialisation et gestion du bus PCIe.
  - Traitement de toutes les erreurs de bus déclenchées par les périphériques d'extrémité PCIe assignés à des domaines d'E/S. Notez que seul le domaine `primary` reçoit toutes les erreurs relatives au bus PCIe.

Configuration matérielle et logicielle requise pour les E/S directes

Pour utiliser la fonctionnalité d'E/S directes (DIO) de manière optimale afin d'assigner des périphériques d'E/S directes à des domaines, vous devez exécuter les logiciels appropriés et utiliser des cartes PCIe prises en charge.

- **Configuration matérielle.** Seules certaines cartes PCIe peuvent être utilisées comme périphériques d'extrémité d'E/S directes sur un domaine d'E/S. Vous pouvez toutefois utiliser d'autres cartes dans votre environnement Oracle VM Server for SPARC, mais sachez qu'elles ne fonctionneront pas avec les E/S directes. Par contre, vous pouvez les utiliser avec des domaines de service et des domaines d'E/S auxquels ont été assignés des complexes root entiers.

  Reportez-vous à la documentation matérielle de votre plate-forme pour vérifier les cartes que vous pouvez utiliser. Pour consulter une liste plus récente des cartes PCIe prises en charge, consultez la page `https://support.oracle.com/CSP/main/article?cmd=show&type=NOT&doctype=REFERENCE&id=1325454.1`.

Remarque - Les serveurs SPARC des séries T7, M7 et S7 disposent d'un contrôleur d'E/S qui fournit plusieurs bus PCIe. En outre, vous pouvez affecter des cartes PCIe à différents domaines. Pour plus d'informations, reportez-vous au Chapitre 7, Création d'un domaine root par assignation de bus PCIe.

- **Configuration logicielle requise.** Pour utiliser la fonctionnalité d'E/S directes, les domaines suivants doivent s'exécuter sur le SE pris en charge
  - **Domaine root.** Au moins le SE Oracle Solaris 11.3.

    Pour les domaines, il est recommandé d'exécuter le SE Oracle Solaris 10 1/13 avec les patchs requis dans la section "Versions du Système d'exploitation Oracle Solaris complètes" du manuel *Guide d'installation d'Oracle VM Server for SPARC 3.4* ou le SE Oracle Solaris 11.3.
  - **Domaine d'E/S.** Au moins le SE Oracle Solaris 11. Notez que la prise en charge de fonctions supplémentaires est incluse dans les versions Oracle Solaris 11 les plus récentes.

Remarque - Toutes les cartes PCIe prises en charge sur la plate-forme sont prises en charge dans les domaines root. Reportez-vous à la documentation de votre plate-forme pour obtenir la liste des cartes PCIe prises en charge. Cependant, seules les cartes PCIe prises en charge par la fonction DIO peuvent être assignées à des domaines d'E/S.

Pour ajouter ou supprimer des périphériques d'extrémité PCIe à l'aide de la fonctionnalité d'E/S directes, vous devez commencer par activer la virtualisation d'E/S sur le bus PCIe lui-même.

Vous pouvez utiliser la commande `ldm set-io` ou `ldm add-io` pour paramétrer la propriété `iov` sur `on`. Vous pouvez également utiliser la commande `ldm add-domain` ou `ldm set-domain` pour paramétrer la propriété `rc-add-policy` sur `iov`. Reportez-vous à la page de manuel `ldm`(1M).

La réinitialisation du domaine root affecte les E/S directes. Planifiez donc soigneusement les modifications que vous apportez à la configuration des E/S directes pour regrouper autant que possible les modifications touchant le domaine root et réduire ainsi le nombre de réinitialisations nécessaires.

Restrictions actuelles de la fonctionnalité d'E/S directes

Pour plus d'informations sur la manière de contourner les restrictions, reportez-vous à la section "Planification de la configuration des périphériques d'extrémité PCIe" à la page 160.

L'assignation d'un périphérique d'extrémité PCIe à un autre domaine non root ou sa suppression est uniquement autorisée lorsque ce domaine est arrêté ou inactif.

Remarque - Le Serveur Fujitsu M10 charge la reconfiguration dynamique de périphériques d'extrémité PCIe. Vous pouvez assigner ou supprimer des périphériques d'extrémité PCIe sans réinitialiser le domaine root ou arrêter le domaine d'E/S.

Pour des informations mises à jour sur cette fonction, reportez-vous au *Guide d'administration et de fonctionnement système des systèmes Fujitsu M10/SPARC M10* de votre modèle à l'adresse suivante : `http://www.fujitsu.com/global/services/computing/server/sparc/downloads/manual/`.

Remarque - La fonctionnalité d'E/S directes n'est pas prise en charge sur les serveurs SPARC des séries M7, T7 et S7. Utilisez la fonction d'affectation de bus PCIe à la place. Reportez-vous au Chapitre 7, Création d'un domaine root par assignation de bus PCIe.

Les systèmes SPARC, jusqu'aux plates-formes SPARC T5 et SPARC M6 incluses, indiquent un nombre limité d'interruptions afin qu'Oracle Solaris limite le nombre d'interruptions utilisables par chaque périphérique. La limite par défaut doit correspondre aux besoins d'une configuration système type, sachant que la valeur devra peut-être être ajustée pour certaines configurations système. Pour plus d'informations, reportez-vous à la section "Ajustement de la limite d'interruption" à la page 427.

Planification de la configuration des périphériques d'extrémité PCIe

Planifiez soigneusement à l'avance l'assignation ou la suppression des périphériques d'extrémité PCIe pour éviter les indisponibilités du domaine root. La réinitialisation du domaine root a non seulement une incidence sur les services disponibles sur le domaine root lui-même, mais affecte également les domaines d'E/S auxquels des périphériques d'extrémité PCIe sont assignés. Bien que les modifications apportées à chaque domaine d'E/S n'affectent pas les autres domaines, la planification anticipée permet de minimiser les conséquences sur les services fournis par ce domaine.

Dans le cadre d'une reconfiguration retardée, vous pouvez ajouter ou supprimer successivement des périphériques, puis réinitialiser une seule fois le domaine root pour appliquer toutes les modifications.

Pour consulter un exemple, reportez-vous à la section "Procédure de création d'un domaine d'E/S par assignation d'un périphérique d'extrémité PCIe" à la page 166.

Vous devez effectuer les étapes suivantes pour planifier et effectuer une configuration de périphérique DIO :

1. Prenez connaissance de la configuration matérielle de votre système et enregistrez-la.

   Plus précisément, enregistrez les informations sur les numéros de référence et d'autres détails des cartes PCIe dans le système.

   Utilisez les commandes `ldm list-io -l` et `prtdiag -v` pour obtenir les informations et enregistrez-les pour pouvoir vous y référer ultérieurement.
2. Déterminez quels périphériques d'extrémité PCIe doivent se trouver dans le domaine `primary`.

   Par exemple, déterminez les périphériques d'extrémité PCIe fournissant l'accès aux éléments suivants :

   - Périphérique du disque d'initialisation
   - Périphérique réseau
   - Autres périphériques que le domaine `primary` offre comme services
3. Supprimez tous les périphériques d'extrémité PCIe que vous êtes susceptible d'utiliser dans des domaines d'E/S.

   Cette étape permet de réduire le nombre de réinitialisations ultérieures du domaine root, car ces réinitialisations affectent les domaines d'E/S.

   Utilisez la commande `ldm remove-io` pour supprimer les périphériques d'extrémité PCIe. Utilisez des pseudonymes plutôt que les chemins d'accès au périphérique pour indiquer les périphériques aux sous-commandes `remove-io` et `add-io`.

   Remarque - Une fois que vous avez supprimé tous les périphériques souhaités dans le cadre d'une reconfiguration retardée, il vous suffit de réinitialiser une fois le domaine root pour appliquer toutes les modifications.

   ---
4. Enregistrez cette configuration sur le processeur de service (SP).

   Utilisez la commande `ldm add-config`.
5. Réinitialisez le domaine root pour libérer les périphériques d'extrémité PCIe que vous avez supprimés à l'étape 3.
6. Confirmez que les périphériques d'extrémité supprimés ne sont plus assignés au domaine root.

   Utilisez la commande `ldm list-io -l` pour vérifier que les périphériques que vous avez supprimés apparaissent comme `SUNW,assigned-device` dans la sortie.
7. Assignez un périphérique d'extrémité PCIe disponible à un domaine invité pour fournir un accès direct au périphérique physique.

   Après avoir effectué cette assignation, vous ne pouvez plus migrer le domaine invité vers un autre système physique au moyen de la fonction de migration.

8. Ajoutez un périphérique d'extrémité PCIe à un domaine invité ou supprimez-en un.

   Utilisez la commande `ldm add-io`.

   Minimisez les modifications aux domaines d'E/S en réduisant les opérations de réinitialisation et en évitant les indisponibilités des services offerts par ce domaine.
9. (Facultatif) Apportez des modifications au matériel PCIe.

   Voir la section "Procédure de modification matérielle PCIe" à la page 164.

Réinitialisation du domaine root avec des extrémités PCIe configurées

Le domaine root est le propriétaire du bus PCIe et est responsable de l'initialisation et de la gestion de ce bus. Le domaine root doit être actif et exécuter une version du SE Oracle Solaris prenant en charge la fonction DIO ou SR-IOV. L'arrêt, l'interruption et la réinitialisation du domaine root interrompent l'accès au bus PCIe. Lorsque le bus PCIe est indisponible, les périphériques PCIe sur ce bus sont affectés et risquent de devenir indisponibles.

Le comportement de domaines d'E/S comportant des périphériques d'extrémité PCIe est imprévisible si le domaine root est réinitialisé alors que ces domaines d'E/S sont en cours d'exécution. Par exemple, les domaines d'E/S avec des périphériques d'extrémité PCIe peuvent paniquer au cours ou après une réinitialisation. Dans cas, il faut arrêter et démarrer manuellement chaque domaine après la réinitialisation du domaine root.

Notez que si le domaine d'E/S est résilient, il peut continuer à fonctionner même si le domaine root qui est le propriétaire du bus PCIe devient indisponible. Voir "Résilience de domaine d'E/S" à la page 145.

Remarque - Un domaine d'E/S ne peut pas démarrer si le domaine root qui lui est associé n'est pas en cours d'exécution.

Pour contourner ces problèmes, procédez comme suit :

- Arrêtez manuellement tous les domaines du système auxquels des périphériques d'extrémité PCIe sont assignés *avant* d'arrêter le domaine root.

  Cette étape garantit que ces domaines sont arrêtés correctement avant que vous n'arrêtiez, ne suspendiez ou ne réinitialisiez le domaine root.

  Pour identifier tous les domaines auxquels des périphériques d'extrémité PCIe sont assignés, exécutez la commande `ldm list-io`. Cette commande vous permet de répertorier les

périphériques d'extrémité PCIe ayant été assignés aux domaines sur le système. Pour obtenir une description détaillée de la sortie de cette commande, reportez-vous à la page de manuel `ldm`(1M).

Pour chaque domaine trouvé, arrêtez le domaine en exécutant la commande `ldm stop`.

- Configurez une relation de dépendance de domaine entre le domaine root et les domaines auxquels des périphériques d'extrémité PCIe sont assignés.

  Cette relation de dépendance garantit que les domaines auxquels des périphériques d'extrémité PCIe sont assignés sont automatiquement redémarrés lorsque le domaine root est réinitialisé pour quelque raison que ce soit.

  Notez que cette relation de dépendance réinitialise ces domaines et qu'ils ne peuvent pas s'arrêter correctement. Cependant, la relation de dépendance n'a aucune incidence sur les domaines arrêtés manuellement.

  ```
  primary# ldm set-domain failure-policy=reset primary
  primary# ldm set-domain master=primary domain-name
  ```

EXEMPLE 33 Configuration des dépendances de la stratégie des pannes pour une configuration avec un domaine root différent de `primary` et des domaines d'E/S.

L'exemple suivant décrit la configuration des dépendances de la stratégie des pannes dans une configuration contenant un domaine root différent de `primary` et des domaines d'E/S.

Dans cet exemple, `ldg1` est un domaine root différent de `primary`. `ldg2` est un domaine d'E/S contenant des fonctions virtuelles SR-IOV PCIe ou des périphériques d'extrémité PCIe assignés à partir d'un complexe root appartenant au domaine `ldg1`.

```
primary# ldm set-domain failure-policy=stop ldg1
primary# ldm set-domain master=ldg1 ldg2
```

Cette relation de dépendance assure que le domaine d'E/S est arrêté lorsque le domaine root `ldg1` se réinitialise.

- En cas de réinitialisation du domaine root différent de `primary`, cette relation de dépendance assure que le domaine d'E/S est arrêté. Démarrez le domaine d'E/S après l'initialisation du domaine root différent de `primary`.

  ```
  primary# ldm start ldg2
  ```

- En cas de réinitialisation du domaine `primary`, la configuration de cette stratégie arrête à la fois le domaine root différent de `primary` et les domaines d'E/S qui en dépendent. Après l'initialisation du domaine `primary`, vous devez tout d'abord démarrer le domaine root différent de `primary`. Lorsque le domaine est initialisé, démarrez le domaine d'E/S.

  ```
  primary# ldm start ldg1
  ```

  Attendez que le domaine `ldg1` soit actif, puis démarrez le domaine d'E/S.

```
primary# ldm start ldg2
```

Procédure de modification matérielle PCIe

La procédure suivante vous aide à éviter la mauvaise configuration des assignations d'extrémité PCIe. Pour obtenir des informations spécifiques à la plate-forme sur l'installation et la suppression de matériel donné, reportez-vous à la documentation de votre plate-forme.

- Aucune action n'est nécessaire si vous installez une carte PCIe dans un emplacement vide. La carte PCIe est automatiquement détenue par le domaine propriétaire du bus PCIe.

  Pour assigner la nouvelle carte PCIe à un domaine d'E/S, utilisez la commande `ldm remove-io` pour, dans un premier temps, supprimer la carte du domaine root. Utilisez ensuite la commande `ldm add-io` pour assigner la carte à un domaine d'E/S.
- Aucune action n'est nécessaire si la carte PCIe est supprimée du système et assignée au domaine root.
- Pour supprimer une carte PCIe assignée à un domaine d'E/S, supprimez d'abord le périphérique du domaine d'E/S. Ajoutez ensuite le périphérique au domaine root avant de retirer physiquement le périphérique du système.
- Pour remplacer une carte PCIe assignée à un domaine d'E/S, vérifiez que la nouvelle carte est prise en charge par la fonction DIO.

  Si tel est le cas, aucune action n'est nécessaire pour assigner automatiquement la nouvelle carte au domaine d'E/S actuel.

  Sinon, supprimez d'abord la carte PCIe du domaine d'E/S à l'aide de la commande `ldm remove-io`. Utilisez ensuite la commande `ldm add-io` pour réassigner la carte PCIe au domaine root. Puis remplacez physiquement la carte PCIe que vous avez assignée au domaine root par une carte PCIe différente. Ces étapes vous permettent d'éviter une configuration non prise en charge par la fonction DIO.

Réduction des interruptions des domaines invités lors du retrait d'une carte PCIe

Lorsque vous retirez ou remplacez une carte PCIe dans un système qui exécute le logiciel Oracle VM Server for SPARC, les domaines qui dépendent de ce matériel ne sont pas disponibles. Pour réduire de telles interruptions de domaines invités, vous devez préparer votre système à utiliser les capacités d'enfichage à chaud afin de retirer la carte physiquement.

▼ Procédure de réduction des interruptions des domaines invités lors du retrait d'une carte PCIe

Cette procédure vous permet d'éviter l'interruption d'un domaine invité qui ne dispose pas d'un périphérique d'E/S directes ou SR-IOV et dont plusieurs chemins sont configurés. Remarquez que cette procédure nécessite deux réinitialisations du domaine `primary`.

Remarque - Cette procédure ne s'applique pas lorsque la carte PCIe se trouve sur un complexe root appartenant à un domaine root qui n'est pas `primary`. Reportez-vous plutôt au document How to Replace PCIe Direct I/O Cards Assigned to an Oracle VM Server for SPARC Guest Domain (Doc ID 1684273.1) (`https://support.oracle.com/epmos/faces/DocumentDisplay?_afrLoop=226878266536565&id=1684273.1&_adf.ctrl-state=bo9fbmr1n_49`).

---

1. **Arrêtez le domaine invité auquel l'emplacement PCIe est assigné.**

   ```
   primary# ldm stop domain-name
   ```

2. **Retirez l'emplacement PCIe du domaine invité.**

   ```
   primary# ldm remove-io PCIe-slot domain-name
   ```

3. **Arrêtez les domaines invités auxquels les emplacements PCIe et les fonctions virtuelles SR-IOV sont assignés.**

   ```
   primary# ldm stop domain-name
   ```

   Remarque - Vous n'avez pas besoin d'arrêter les domaines invités auxquels les bus PCIe sont assignés car ils peuvent fournir d'autres chemins vers les périphériques réseau et de disque aux domaines invités.

   ---

4. **Démarrez une reconfiguration retardée sur le domaine `primary` de manière à pouvoir lui assigner cet emplacement.**

   ```
   primary# ldm start-reconf primary
   ```

5. **Ajoutez l'emplacement PCIe au domaine `primary`.**

   ```
   primary# ldm add-io PCIe-slot domain-name
   ```

6. **Réinitialisez le domaine `primary`.**

   ```
   primary# shutdown -i6 -g0 -y
   ```

7. **Utilisez les commandes d'enfichage à chaud pour remplacer la carte PCIe.**
   Pour plus d'informations sur les capacités d'enfichage à chaud du SE Oracle Solaris, reportez-vous au Chapitre 2, "Dynamically Configuring Devices" du manuel *Managing Devices in Oracle Solaris 11.3*.

8. **Après le remplacement de la carte, procédez aux étapes suivantes si vous devez réassigner ce même emplacement PCIe au domaine invité :**

   a. **Déclenchez une reconfiguration retardée sur le domaine `primary`.**

      ```
      primary# ldm start-reconf primary
      ```

   b. **Retirez l'emplacement PCIe du domaine `primary`.**

      ```
      primary# ldm remove-io PCIe-slot domain-name
      ```

   c. **Réinitialisez le domaine `primary` pour que le retrait de l'emplacement PCIe prenne effet.**

      ```
      primary# shutdown -i6 -g0 -y
      ```

   d. **Réassignez l'emplacement PCIe au domaine invité.**

      ```
      primary# ldm add-io PCIe-slot domain-name
      ```

   e. **Démarrez les domaines invités auxquels vous souhaitez assigner les emplacements PCIe et les fonctions virtuelles SR-IOV.**

      ```
      primary# ldm start domain-name
      ```

Procédure de création d'un domaine d'E/S par assignation d'un périphérique d'extrémité PCIe

▼ Procédure de création d'un domaine d'E/S par assignation d'un périphérique d'extrémité PCIe

Planifiez tous les déploiements DIO à l'avance pour réduire les indisponibilités.

Attention - Le domaine `primary` n'a plus accès au périphérique DVD intégré si vous affectez l'emplacement `/SYS/MB/SASHBA1` sur un système SPARC T3-1 ou SPARC T4-1 à un domaine DIO.

Les systèmes SPARC T3-1 et SPARC T4-1 comportent deux emplacements d'E/S directes pour un stockage intégré ; ces emplacements sont représentés par les chemins `/SYS/MB/SASHBA0` et `/SYS/MB/SASHBA1`. Outre des disques intégrés à têtes multiples, l'emplacement `/SYS/MB/SASHBA1` héberge le périphérique DVD intégré. Par conséquent, si vous affectez `/SYS/MB/SASHBA1` à un domaine DIO, le domaine `primary` n'a plus accès au périphérique DVD intégré.

Les systèmes SPARC T3-2 et SPARC T4-2 ont un emplacement `SASHBA` unique hébergeant tous les disques intégrés ainsi que le périphérique DVD intégré. Par conséquent, si vous affectez `SASHBA` à un domaine DIO, les disques intégrés *et* le périphérique DVD intégré sont prêtés au domaine DIO et ne sont pas disponibles pour le domaine `primary`.

Pour obtenir un exemple d'ajout de périphérique d'extrémité PCIe pour créer un domaine d'E/S, reportez-vous à la section "Planification de la configuration des périphériques d'extrémité PCIe" à la page 160.

Remarque - Dans cette version, il est préférable d'utiliser le NCP `DefaultFixed` pour configurer les liaisons de données et les interfaces réseau sur les systèmes Oracle Solaris 11.

Le SE Oracle Solaris 11 inclut les NCP suivants :

- `DefaultFixed` – Vous permet d'utiliser la commande `dladm` ou `ipadm` pour gérer la mise en réseau
- `Automatic` – Vous permet d'utiliser la commande `netcfg` ou `netadm` pour gérer la mise en réseau

Assurez-vous à l'aide de la commande `netadm list` que le NCP `DefaultFixed` est activé. Reportez-vous au Chapitre 7, "Using Datalink and Interface Configuration Commands on Profiles" du manuel *Oracle Solaris Administration: Network Interfaces and Network Virtualization*.

1. **Identifiez et archivez les périphériques qui sont actuellement installés sur le système.**

   La sortie de la commande `ldm list-io -l` montre comment les périphériques d'E/S sont actuellement configurés. Vous pouvez obtenir des informations plus détaillées à l'aide de la commande `prtdiag -v`.

   Remarque - Après l'assignation des périphériques à des domaines d'E/S, l'identité des périphériques peut uniquement être déterminée dans les domaines d'E/S.

```
primary# ldm list-io -l
NAME                                     TYPE   BUS      DOMAIN   STATUS
----                                     ----   ---      ------   ------
niu_0                                    NIU    niu_0    primary
[niu@480]
niu_1                                    NIU    niu_1    primary
[niu@580]
pci_0                                    BUS    pci_0    primary
[pci@400]
pci_1                                    BUS    pci_1    primary
[pci@500]
/SYS/MB/PCIE0                            PCIE   pci_0    primary  OCC
[pci@400/pci@2/pci@0/pci@8]
    SUNW,emlxs@0/fp/disk
    SUNW,emlxs@0/fp/tape
    SUNW,emlxs@0/fp@0,0
    SUNW,emlxs@0,1/fp/disk
    SUNW,emlxs@0,1/fp/tape
    SUNW,emlxs@0,1/fp@0,0
/SYS/MB/PCIE2                            PCIE   pci_0    primary  OCC
[pci@400/pci@2/pci@0/pci@4]
    pci/scsi/disk
    pci/scsi/tape
    pci/scsi/disk
    pci/scsi/tape
/SYS/MB/PCIE4                            PCIE   pci_0    primary  OCC
[pci@400/pci@2/pci@0/pci@0]
    ethernet@0
    ethernet@0,1
    SUNW,qlc@0,2/fp/disk
    SUNW,qlc@0,2/fp@0,0
    SUNW,qlc@0,3/fp/disk
    SUNW,qlc@0,3/fp@0,0
/SYS/MB/PCIE6                            PCIE   pci_0    primary  EMP
[pci@400/pci@1/pci@0/pci@8]
/SYS/MB/PCIE8                            PCIE   pci_0    primary  EMP
[pci@400/pci@1/pci@0/pci@c]
/SYS/MB/SASHBA                           PCIE   pci_0    primary  OCC
[pci@400/pci@2/pci@0/pci@e]
    scsi@0/iport@1
    scsi@0/iport@2
    scsi@0/iport@4
    scsi@0/iport@8
    scsi@0/iport@80/cdrom@p7,0
    scsi@0/iport@v0
/SYS/MB/NET0                             PCIE   pci_0    primary  OCC
[pci@400/pci@1/pci@0/pci@4]
    network@0
    network@0,1
/SYS/MB/PCIE1                            PCIE   pci_1    primary  OCC
[pci@500/pci@2/pci@0/pci@a]
    SUNW,qlc@0/fp/disk
    SUNW,qlc@0/fp@0,0
    SUNW,qlc@0,1/fp/disk
    SUNW,qlc@0,1/fp@0,0
/SYS/MB/PCIE3                            PCIE   pci_1    primary  OCC
[pci@500/pci@2/pci@0/pci@6]
    network@0
    network@0,1
    network@0,2
    network@0,3
```

```
/SYS/MB/PCIE5                              PCIE   pci_1    primary  OCC
[pci@500/pci@2/pci@0/pci@0]
    network@0
    network@0,1
/SYS/MB/PCIE7                              PCIE   pci_1    primary  EMP
[pci@500/pci@1/pci@0/pci@6]
/SYS/MB/PCIE9                              PCIE   pci_1    primary  EMP
[pci@500/pci@1/pci@0/pci@0]
/SYS/MB/NET2                               PCIE   pci_1    primary  OCC
[pci@500/pci@1/pci@0/pci@5]
    network@0
    network@0,1
    ethernet@0,80
/SYS/MB/NET0/IOVNET.PF0                    PF     pci_0    primary
[pci@400/pci@1/pci@0/pci@4/network@0]
    maxvfs = 7
/SYS/MB/NET0/IOVNET.PF1                    PF     pci_0    primary
[pci@400/pci@1/pci@0/pci@4/network@0,1]
    maxvfs = 7
/SYS/MB/PCIE5/IOVNET.PF0                   PF     pci_1    primary
[pci@500/pci@2/pci@0/pci@0/network@0]
    maxvfs = 63
/SYS/MB/PCIE5/IOVNET.PF1                   PF     pci_1    primary
[pci@500/pci@2/pci@0/pci@0/network@0,1]
    maxvfs = 63
/SYS/MB/NET2/IOVNET.PF0                    PF     pci_1    primary
[pci@500/pci@1/pci@0/pci@5/network@0]
    maxvfs = 7
/SYS/MB/NET2/IOVNET.PF1                    PF     pci_1    primary
[pci@500/pci@1/pci@0/pci@5/network@0,1]
    maxvfs = 7
```

2. **Déterminez le chemin du périphérique du disque d'initialisation qui doit être conservé.**

   Reportez-vous à l'étape 2 de la section "Procédure de création d'un domaine root par affectation d'un bus PCIe" à la page 74.

3. **Déterminez le périphérique physique auquel le périphérique en mode bloc est connecté.**

   Reportez-vous à l'étape 3 de la section "Procédure de création d'un domaine root par affectation d'un bus PCIe" à la page 74.

4. **Identifiez l'interface réseau utilisée par le système.**

   Reportez-vous à l'étape 4 de la section "Procédure de création d'un domaine root par affectation d'un bus PCIe" à la page 74.

5. **Identifiez le périphérique physique auquel l'interface réseau est connectée.**

   La commande suivante utilise l'interface réseau `igb0` :

```
primary# ls -l /dev/igb0
lrwxrwxrwx   1 root     root          46 Jul 30 17:29 /dev/igb0 ->
../devices/pci@500/pci@0/pci@8/network@0:igb0
```

Dans cet exemple, le périphérique physique de l'interface réseau utilisée par le domaine `primary` est connecté au périphérique d'extrémité PCIe (`pci@500/pci@0/pci@8`), qui correspond à la liste de `MB/NET0` à l'étape 1. Par conséquent, vous ne voulez pas supprimer ce périphérique du domaine `primary`. Vous pouvez assigner en toute sécurité tous les autres périphériques PCIe aux autres domaines, car ils ne sont pas utilisés par le domaine `primary`.

Si l'interface réseau utilisée par le domaine `primary` se trouve sur un bus que vous voulez assigner à un autre domaine, le domaine `primary` doit être reconfiguré pour utiliser une interface réseau différente.

6. Supprimez les périphériques d'extrémité PCIe que vous risquez d'utiliser dans les domaines d'E/S.

Dans cet exemple, vous pouvez supprimer les périphériques d'extrémité `PCIE2`, `PCIE3`, `PCIE4` et `PCIE5`, car ils ne sont pas utilisés par le domaine `primary`.

a. Supprimez les périphériques d'extrémité PCIe.

Attention - Ne supprimez pas les périphériques utilisés ou requis par le domaine `primary`. Ne supprimez pas un bus dont les périphériques, tels que les ports réseau ou périphériques `usbecm` sont utilisés par un domaine.

Si vous supprimez par erreur les mauvais périphériques, utilisez la commande `ldm cancel-reconf primary` pour annuler la reconfiguration retardée sur le domaine `primary`.

Vous pouvez supprimer plusieurs périphériques à la fois pour éviter plusieurs réinitialisations.

```
primary# ldm start-reconf primary
primary# ldm set-io iov=on pci_1
All configuration changes for other domains are disabled until the primary
domain reboots, at which time the new configuration for the primary domain
will also take effect.
primary# ldm remove-io /SYS/MB/PCIE1 primary
------------------------------------------------------------------------------
Notice: The primary domain is in the process of a delayed reconfiguration.
Any changes made to the primary domain will only take effect after it reboots.
------------------------------------------------------------------------------
primary# ldm remove-io /SYS/MB/PCIE3 primary
------------------------------------------------------------------------------
Notice: The primary domain is in the process of a delayed reconfiguration.
Any changes made to the primary domain will only take effect after it reboots.
------------------------------------------------------------------------------
primary# ldm remove-io /SYS/MB/PCIE5 primary
------------------------------------------------------------------------------
Notice: The primary domain is in the process of a delayed reconfiguration.
Any changes made to the primary domain will only take effect after it reboots.
------------------------------------------------------------------------------
```

b. Enregistrez la nouvelle configuration sur le processeur de service (SP).

La commande suivante enregistre la configuration dans un fichier nommé `dio` :

```
primary# ldm add-config dio
```

c. Réinitialisez le système pour prendre en compte la suppression des périphériques d'extrémité PCIe.

```
primary# shutdown -i6 -g0 -y
```

7. Connectez-vous au domaine `primary` et vérifiez que les périphériques d'extrémité PCIe ne sont plus assignés au domaine.

```
primary# ldm list-io
NAME                                    TYPE   BUS      DOMAIN   STATUS
----                                    ----   ---      ------   ------
niu_0                                   NIU    niu_0    primary
niu_1                                   NIU    niu_1    primary
pci_0                                   BUS    pci_0    primary
pci_1                                   BUS    pci_1    primary  IOV
/SYS/MB/PCIE0                           PCIE   pci_0    primary  OCC
/SYS/MB/PCIE2                           PCIE   pci_0    primary  OCC
/SYS/MB/PCIE4                           PCIE   pci_0    primary  OCC
/SYS/MB/PCIE6                           PCIE   pci_0    primary  EMP
/SYS/MB/PCIE8                           PCIE   pci_0    primary  EMP
/SYS/MB/SASHBA                          PCIE   pci_0    primary  OCC
/SYS/MB/NET0                            PCIE   pci_0    primary  OCC
/SYS/MB/PCIE1                           PCIE   pci_1             OCC
/SYS/MB/PCIE3                           PCIE   pci_1             OCC
/SYS/MB/PCIE5                           PCIE   pci_1             OCC
/SYS/MB/PCIE7                           PCIE   pci_1    primary  EMP
/SYS/MB/PCIE9                           PCIE   pci_1    primary  EMP
/SYS/MB/NET2                            PCIE   pci_1    primary  OCC
/SYS/MB/NET0/IOVNET.PF0                 PF     pci_0    primary
/SYS/MB/NET0/IOVNET.PF1                 PF     pci_0    primary
/SYS/MB/NET2/IOVNET.PF0                 PF     pci_1    primary
/SYS/MB/NET2/IOVNET.PF1                 PF     pci_1    primary
```

Remarque - La sortie `ldm list-io -l` peut indiquer `SUNW,assigned-device` pour les périphériques d'extrémité PCIe supprimés. Les informations actuelles ne sont plus disponibles à partir du domaine `primary`, mais le domaine auquel le périphérique est assigné a ces informations.

8. Assignez un périphérique d'extrémité PCIe à un domaine.

a. Ajoutez le périphérique `PCI3` au domaine `ldg1`.

```
primary# ldm add-io /SYS/MB/PCIE3 ldg1
```

b. Associez et démarrez le domaine `ldg1`.

```
primary# ldm bind ldg1
primary# ldm start ldg1
LDom ldg1 started
```

9. **Connectez-vous au domaine `ldg1` et vérifiez que le périphérique est disponible à l'utilisation.**

   Assurez-vous que le périphérique réseau est disponible, puis configurez le périphérique réseau à utiliser dans le domaine.

```
primary# dladm show-phys
LINK              MEDIA               STATE      SPEED  DUPLEX    DEVICE
net0              Ethernet            unknown    0      unknown   nxge0
net1              Ethernet            unknown    0      unknown   nxge1
net2              Ethernet            unknown    0      unknown   nxge2
net3              Ethernet            unknown    0      unknown   nxge3
```

Problèmes propres aux E/S directes

La carte fibre 10 Gigabit Ethernet double, PCI Express Atlas affiche quatre sous-périphériques dans la sortie `ldm list-io -l`

Lorsque vous exécutez la commande `ldm ls-io -l` sur un système équipé d'une carte fibre 10 Gigabit Ethernet double, PCI Express (X1027A-Z), la sortie peut afficher les informations suivantes :

```
primary# ldm list-io -l
...
pci@500/pci@0/pci@c PCIE5 OCC primary
network@0
network@0,1
ethernet
ethernet
```

La sortie affiche quatre sous-périphériques même si la carte Ethernet ne possède que deux ports. Cette anomalie se présente si la carte comporte quatre fonctions PCI. Deux de ces fonctions sont désactivées en interne et s'affichent comme étant des ports `ethernet` dans la sortie `ldm ls-io -l`.

Vous pouvez ignorer les entrées `ethernet` dans la sortie `ldm ls-io -l`.

♦ ♦ ♦ CHAPITRE 10

Utilisation de domaines root différents du domaine `primary`

Ce chapitre aborde les sujets relatifs aux domaines root non `primary` suivants :

Présentation des domaines root non `primary`

Chaque *domaine root* a un complexe root PCIe qui lui est affecté. Ce domaine possède la topologie Fabric PCIe et fournit tous les services associés à la Fabric, notamment le traitement des erreurs Fabric. Un domaine root est également un domaine d'E/S, car il possède et a un accès direct aux périphériques d'E/S. Le domaine `primary` est le domaine root par défaut.

Vous pouvez effectuer des E/S directes et des opérations SR-IOV sur les bus PCIe qui sont affectés à l'un des domaines root. Vous pouvez alors effectuer les opérations suivantes pour tous les domaines root, y compris les domaines root non `primary` :

- Afficher le statut des emplacements PCIe
- Afficher les fonctions physiques SR-IOV présentes
- Assigner un emplacement PCIe à un domaine d'E/S ou un domaine root
- Supprimer un emplacement PCIe d'un domaine d'E/S ou d'un domaine root
- Créer une fonction virtuelle à partir de sa fonction physique
- Détruire une fonction virtuelle
- Assigner une fonction virtuelle à un autre domaine
- Supprimer une fonction virtuelle d'un autre domaine

Logical Domains Manager récupère les périphériques d'extrémité PCIe et les périphériques de fonctions virtuelles SR-IOV à partir des agents Logical Domains qui s'exécutent dans les domaines root non `primary`. Après leur détection initiale, ces informations sont placées dans la mémoire cache pendant l'arrêt du domaine root, mais uniquement pendant la durée de la réinitialisation du domaine root.

Vous ne pouvez effectuer des opérations d'E/S directes et SR-IOV que lorsque le domaine root est actif. Logical Domains Manager s'exécute sur les périphériques qui sont présents à ce moment-là. Les données mises en cache sont susceptibles d'être actualisées lorsque les opérations suivantes sont exécutées :

- Redémarrage de l'agent Logical Domains dans le domaine root spécifié
- Un remplacement de matériel, par exemple une opération d'enfichage à chaud, est effectué dans le domaine root spécifié

Utilisez la commande `ldm list-io` pour afficher le statut du périphérique d'extrémité PCIe. La sortie indique également les sous-périphériques et les périphériques de fonction physique associés aux complexes racine détenus par chaque domaine root non `primary`.

Vous pouvez appliquer les commandes suivantes sur n'importe quel domaine root :

- `ldm add-io`
- `ldm remove-io`
- `ldm set-io`
- `ldm create-vf`
- `ldm destroy-vf`
- `ldm start-reconf`
- `ldm cancel-reconf`

La prise en charge de la reconfiguration retardée a été étendue aux domaines root non `primary`. Toutefois, elle peut *uniquement* être utilisée pour exécuter les commandes `ldm add-io`, `ldm remove-io`, `ldm set-io`, `ldm create-vf` et `ldm destroy-vf`. La reconfiguration retardée peut être utilisée pour n'importe quelle opération qui ne peut pas être réalisée à l'aide d'opérations dynamiques telles que :

- Réalisation d'opérations d'E/S directes
- Création et destruction de fonctions virtuelles à partir d'une fonction physique qui ne répond pas aux exigences de configuration de la fonction SR-IOV dynamique.

Attention - Planifiez ces opérations à l'avance afin de limiter le nombre de réinitialisations du domaine root et réduire les périodes d'indisponibilité.

Configuration requise pour les domaines root non primary

Il est possible d'utiliser des domaines root non-primary en plus du domaine de contrôle pour fournir des fonctions d'E/S directes et SR-IOV à d'autres domaines. Cette fonctionnalité est prise en charge à partir de la série SPARC T4 et sur les Serveurs Fujitsu M10.

- **Configuration matérielle.**

  En plus des cartes PCIe pour l'E/S directe et des fonctions SR-IOV décrites dans https://support.oracle.com/CSP/main/article?cmd=show&type=NOT&doctype=REFERENCE&id=1325454.1, d'autres cartes PCIe peuvent être utilisées, mais pas pour les fonctions DIO et SR-IOV. Pour déterminer les cartes que vous pouvez utiliser sur votre plate-forme, reportez-vous à la documentation matérielle de votre plate-forme.
- **Configuration de microprogramme requise.**

  Les plates-formes SPARC T4 doivent au moins exécuter la version 8.4.0.a du microprogramme du système.

  Les serveurs SPARC T5, SPARC M5 et SPARC M6 doivent au moins exécuter la version 9.1.0.*x* du microprogramme du système.

  Les serveurs de série SPARC T7 et SPARC M7 doivent au moins exécuter la version 9.4.3 du microprogramme du système.

  Les Serveurs Fujitsu M10 doivent exécuter au moins la version XCP2210 du microprogramme système.
- **Configuration logicielle requise.**

  Les domaines non-primary doivent au moins exécuter le SE Oracle Solaris 11.2.

Restrictions du domaine root non primary

L'utilisation du domaine root non primary est soumise aux restrictions suivantes :

- Un domaine d'E/S ne peut pas démarrer si le domaine root qui lui est associé n'est pas en cours d'exécution.
- La prise en charge de la reconfiguration retardée a été étendue aux domaines root non primary. Jusqu'à la réinitialisation du domaine root ou l'annulation de la reconfiguration retardée, vous pouvez uniquement exécuter les commandes suivantes :
  - `ldm add-io`
  - `ldm remove-io`
  - `ldm set-io`

  - `ldm create-vf`
  - `ldm destroy-vf`
- Le domaine root doit être actif et initialisé pour exécuter les opérations suivantes :
  - Création et destruction de fonctions virtuelles SR-IOV
  - Ajout et suppression d'emplacements PCIe
  - Ajout et suppression de fonctions virtuelles SR-IOV
- Vous devez démarrer une reconfiguration retardée sur le domaine root lorsque vous effectuez les opérations d'E/S directes `ldm add-io` et `ldm remove-io` pour les emplacements PCIe.
- Si votre configuration ne satisfait pas les conditions requises pour la virtualisation d'E/S dynamique, vous devez utiliser la reconfiguration retardée pour les opérations de fonction virtuelle SR-IOV suivantes :
  - `ldm create-vf`
  - `ldm destroy-vf`
  - `ldm add-io`
  - `ldm remove-io`
  - `ldm set-io`
- La réinitialisation d'un domaine root a une incidence sur tous les domaines d'E/S comportant un périphérique d'un bus PCIe appartenant au domaine root. Voir la section "Réinitialisation du domaine root avec des extrémités PCIe configurées" à la page 162.
- Vous ne pouvez pas assigner une fonction virtuelle SR-IOV ou un emplacement PCIe d'un domaine root à un autre domaine root. Cette restriction empêche les dépendances circulaires.

Exemples de domaines root non `primary`

Les exemples suivants expliquent comment activer la virtualisation d'E/S pour un bus PCIe, gérer les périphériques d'E/S directes sur les domaines root non `primary` et comment gérer les fonctions virtuelles SR-IOV sur les domaines root `primary`.

Activation de la virtualisation d'E/S pour un bus PCIe

L'exemple suivant illustre l'activation de la virtualisation d'E/S à l'aide des commandes `ldm add-io` et `ldm set-io`

La configuration d'E/S SPARC T4-2 suivante montre que le bus pci_1 a déjà été supprimé du domaine primary.

```
primary# ldm list-io
NAME                                      TYPE   BUS      DOMAIN   STATUS
----                                      ----   ---      ------   ------
pci_0                                     BUS    pci_0    primary  IOV
pci_1                                     BUS    pci_1
niu_0                                     NIU    niu_0    primary
niu_1                                     NIU    niu_1    primary
/SYS/MB/PCIE0                             PCIE   pci_0    primary  OCC
/SYS/MB/PCIE2                             PCIE   pci_0    primary  OCC
/SYS/MB/PCIE4                             PCIE   pci_0    primary  OCC
/SYS/MB/PCIE6                             PCIE   pci_0    primary  EMP
/SYS/MB/PCIE8                             PCIE   pci_0    primary  EMP
/SYS/MB/SASHBA                            PCIE   pci_0    primary  OCC
/SYS/MB/NET0                              PCIE   pci_0    primary  OCC
/SYS/MB/PCIE1                             PCIE   pci_1             UNK
/SYS/MB/PCIE3                             PCIE   pci_1             UNK
/SYS/MB/PCIE5                             PCIE   pci_1             UNK
/SYS/MB/PCIE7                             PCIE   pci_1             UNK
/SYS/MB/PCIE9                             PCIE   pci_1             UNK
/SYS/MB/NET2                              PCIE   pci_1             UNK
/SYS/MB/NET0/IOVNET.PF0                   PF     pci_0    primary
/SYS/MB/NET0/IOVNET.PF1                   PF     pci_0    primary
```

La liste suivante indique que les domaines invités sont dans un état lié :

```
primary# ldm list
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  NORM  UPTIME
primary          active     -n-cv-  UART    8     8G       0.6%  0.6%  8m
rootdom1         bound      ------  5000    8     4G
ldg2             bound      ------  5001    8     4G
ldg3             bound      ------  5002    8     4G
```

La commande ldm add-io suivante ajoute le bus pci_1 au domaine rootdom1 en activant la virtualisation d'E/S pour ce bus. La commande ldm start démarre le domaine rootdom1.

```
primary# ldm add-io iov=on pci_1 rootdom1
primary# ldm start rootdom1
LDom rootdom1 started
```

Si un bus PCIe spécifié est déjà assigné à un domaine root, servez-vous de la commande ldm set-io pour activer la virtualisation d'E/S.

```
primary# ldm start-reconf rootdom1
primary# ldm set-io iov=on pci_1
primary# ldm stop-domain -r rootdom1
```

Vous ne pouvez configurer les périphériques d'E/S qu'une fois le système d'exploitation du domaine root démarré. Connectez-vous à la console du domaine invité rootdom1 puis initialisez le SE du domaine root rootdom1 si l'initialisation automatique n'est pas activée sur les domaines invités.

```
primary# telnet localhost 5000
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
```

```
Connecting to console "rootdom1" in group "rootdom1" ....
Press ~? for control options ..
ok> boot
...
primary#
```

La commande suivante indique que le bus PCIe pci_1 et ses enfants appartiennent désormais au domaine root rootdom1.

```
primary# ldm list-io
NAME                                    TYPE   BUS      DOMAIN   STATUS
----                                    ----   ---      ------   ------
pci_0                                   BUS    pci_0    primary  IOV
pci_1                                   BUS    pci_1    rootdom1     IOV
niu_0                                   NIU    niu_0    primary
niu_1                                   NIU    niu_1    primary
/SYS/MB/PCIE0                           PCIE   pci_0    primary  OCC
/SYS/MB/PCIE2                           PCIE   pci_0    primary  OCC
/SYS/MB/PCIE4                           PCIE   pci_0    primary  OCC
/SYS/MB/PCIE6                           PCIE   pci_0    primary  EMP
/SYS/MB/PCIE8                           PCIE   pci_0    primary  EMP
/SYS/MB/SASHBA                          PCIE   pci_0    primary  OCC
/SYS/MB/NET0                            PCIE   pci_0    primary  OCC
/SYS/MB/PCIE1                           PCIE   pci_1    rootdom1     OCC
/SYS/MB/PCIE3                           PCIE   pci_1    rootdom1     OCC
/SYS/MB/PCIE5                           PCIE   pci_1    rootdom1     OCC
/SYS/MB/PCIE7                           PCIE   pci_1    rootdom1     OCC
/SYS/MB/PCIE9                           PCIE   pci_1    rootdom1     EMP
/SYS/MB/NET2                            PCIE   pci_1    rootdom1     OCC
/SYS/MB/NET0/IOVNET.PF0                 PF     pci_0    primary
/SYS/MB/NET0/IOVNET.PF1                 PF     pci_0    primary
/SYS/MB/PCIE5/IOVNET.PF0                PF     pci_1    rootdom1
/SYS/MB/PCIE5/IOVNET.PF1                PF     pci_1    rootdom1
/SYS/MB/NET2/IOVNET.PF0                 PF     pci_1    rootdom1
/SYS/MB/NET2/IOVNET.PF1                 PF     pci_1    rootdom1
```

Gestion des périphériques d'E/S directes sur des domaines root non primary

L'exemple suivant indique comment gérer des périphériques d'E/S directes sur des domaines root non primary.

La commande suivante génère une erreur car elle tente de supprimer un emplacement du domaine root alors que celui-ci est encore actif :

```
primary# ldm remove-io /SYS/MB/PCIE7 ldg1
Dynamic I/O operations on PCIe slots are not supported.
Use start-reconf command to trigger delayed reconfiguration and make I/O
changes statically.
```

La commande suivante illustre la méthode adaptée de suppression d'un emplacement, à savoir en lançant tout d'abord une reconfiguration retardée sur le domaine root.

```
primary# ldm start-reconf ldg1
Initiating a delayed reconfiguration operation on the ldg1 domain.
All configuration changes for other domains are disabled until the ldg1
domain reboots, at which time the new configuration for the ldg1 domain
will also take effect.
primary# ldm remove-io /SYS/MB/PCIE7 ldg1
--------------------------------------------------------------------------------
Notice: The ldg1 domain is in the process of a delayed reconfiguration.
Any changes made to the ldg1 domain will only take effect after it reboots.
--------------------------------------------------------------------------------
primary# ldm stop-domain -r ldg1
```

La commande `ldm list-io` suivante vérifie que l'emplacement `/SYS/MB/PCIE7` ne se trouve plus sur le domaine root.

```
primary# ldm list-io
NAME                                      TYPE   BUS      DOMAIN   STATUS
----                                      ----   ---      ------   ------
pci_0                                     BUS    pci_0    primary  IOV
pci_1                                     BUS    pci_1    ldg1     IOV
niu_0                                     NIU    niu_0    primary
niu_1                                     NIU    niu_1    primary
/SYS/MB/PCIE0                             PCIE   pci_0    primary  OCC
/SYS/MB/PCIE2                             PCIE   pci_0    primary  OCC
/SYS/MB/PCIE4                             PCIE   pci_0    primary  OCC
/SYS/MB/PCIE6                             PCIE   pci_0    primary  EMP
/SYS/MB/PCIE8                             PCIE   pci_0    primary  EMP
/SYS/MB/SASHBA                            PCIE   pci_0    primary  OCC
/SYS/MB/NET0                              PCIE   pci_0    primary  OCC
/SYS/MB/PCIE1                             PCIE   pci_1    ldg1     OCC
/SYS/MB/PCIE3                             PCIE   pci_1    ldg1     OCC
/SYS/MB/PCIE5                             PCIE   pci_1    ldg1     OCC
/SYS/MB/PCIE7                             PCIE   pci_1             OCC
/SYS/MB/PCIE9                             PCIE   pci_1    ldg1     EMP
/SYS/MB/NET2                              PCIE   pci_1    ldg1     OCC
/SYS/MB/NET0/IOVNET.PF0                   PF     pci_0    primary
/SYS/MB/NET0/IOVNET.PF1                   PF     pci_0    primary
/SYS/MB/PCIE5/IOVNET.PF0                  PF     pci_1    ldg1
/SYS/MB/PCIE5/IOVNET.PF1                  PF     pci_1    ldg1
/SYS/MB/NET2/IOVNET.PF0                   PF     pci_1    ldg1
/SYS/MB/NET2/IOVNET.PF1                   PF     pci_1    ldg1
```

Les commandes suivantes assignent l'emplacement `/SYS/MB/PCIE7` au domaine `ldg2`. La commande `ldm start` démarre le domaine `ldg2`.

```
primary# ldm add-io /SYS/MB/PCIE7 ldg2
primary# ldm start ldg2
LDom ldg2 started
```

Gestion de fonctions virtuelles SR-IOV sur des domaines root non `primary`

Ces commandes créent deux fonctions virtuelles à partir de chacune des deux fonctions physiques appartenant au domaine root non `primary`.

```
primary# ldm create-vf /SYS/MB/PCIE5/IOVNET.PF0
Created new vf: /SYS/MB/PCIE5/IOVNET.PF0.VF0
primary# ldm create-vf /SYS/MB/PCIE5/IOVNET.PF0
Created new vf: /SYS/MB/PCIE5/IOVNET.PF0.VF1
primary# ldm create-vf /SYS/MB/NET2/IOVNET.PF1
Created new vf: /SYS/MB/NET2/IOVNET.PF1.VF0
primary# ldm create-vf /SYS/MB/NET2/IOVNET.PF1
Created new vf: /SYS/MB/NET2/IOVNET.PF1.VF1
```

Vous pouvez également utiliser l'option -n pour créer deux fonctions virtuelles à l'aide des deux commandes suivantes :

```
primary# ldm create-vf -n 2 /SYS/MB/PCIE5/IOVNET.PF0
Created new vf: /SYS/MB/PCIE5/IOVNET.PF0.VF0
Created new vf: /SYS/MB/PCIE5/IOVNET.PF0.VF1
primary# ldm create-vf -n 2 /SYS/MB/NET2/IOVNET.PF1
Created new vf: /SYS/MB/NET2/IOVNET.PF1.VF0
Created new vf: /SYS/MB/NET2/IOVNET.PF1.VF1
```

Si vous n'êtes pas parvenu à créer les fonctions virtuelles sur une fonction physique donnée de façon dynamique, démarrez une reconfiguration retardée afin de les créer de façon statique.

```
primary# ldm start-reconf ldg1
primary# ldm create-vf /SYS/MB/PCIE5/IOVNET.PF0
Created new vf: /SYS/MB/PCIE5/IOVNET.PF0.VF0
primary# ldm create-vf /SYS/MB/PCIE5/IOVNET.PF0
Created new vf: /SYS/MB/PCIE5/IOVNET.PF0.VF1
primary# ldm create-vf /SYS/MB/NET2/IOVNET.PF1
Created new vf: /SYS/MB/NET2/IOVNET.PF1.VF0
primary# ldm create-vf /SYS/MB/NET2/IOVNET.PF1
Created new vf: /SYS/MB/NET2/IOVNET.PF1.VF1
```

Ensuite, réinitialisez le domaine root, ldg1, afin d'appliquer les modifications.

```
primary# ldm stop-domain -r ldg1
```

La commande suivante affiche les nouvelles fonctions virtuelles :

```
primary# ldm list-io
NAME                                      TYPE   BUS      DOMAIN   STATUS
----                                      ----   ---      ------   ------
pci_0                                     BUS    pci_0    primary  IOV
pci_1                                     BUS    pci_1    ldg1     IOV
niu_0                                     NIU    niu_0    primary
niu_1                                     NIU    niu_1    primary
/SYS/MB/PCIE0                             PCIE   pci_0    primary  OCC
/SYS/MB/PCIE2                             PCIE   pci_0    primary  OCC
/SYS/MB/PCIE4                             PCIE   pci_0    primary  OCC
/SYS/MB/PCIE6                             PCIE   pci_0    primary  EMP
/SYS/MB/PCIE8                             PCIE   pci_0    primary  EMP
/SYS/MB/SASHBA                            PCIE   pci_0    primary  OCC
/SYS/MB/NET0                              PCIE   pci_0    primary  OCC
/SYS/MB/PCIE1                             PCIE   pci_1    ldg1     OCC
/SYS/MB/PCIE3                             PCIE   pci_1    ldg1     OCC
/SYS/MB/PCIE5                             PCIE   pci_1    ldg1     OCC
/SYS/MB/PCIE7                             PCIE   pci_1    ldg2     OCC
/SYS/MB/PCIE9                             PCIE   pci_1    ldg1     EMP
/SYS/MB/NET2                              PCIE   pci_1    ldg1     OCC
/SYS/MB/NET0/IOVNET.PF0                   PF     pci_0    primary
/SYS/MB/NET0/IOVNET.PF1                   PF     pci_0    primary
```

```
/SYS/MB/PCIE5/IOVNET.PF0                PF     pci_1    ldg1
/SYS/MB/PCIE5/IOVNET.PF1                PF     pci_1    ldg1
/SYS/MB/NET2/IOVNET.PF0                 PF     pci_1    ldg1
/SYS/MB/NET2/IOVNET.PF1                 PF     pci_1    ldg1
/SYS/MB/PCIE5/IOVNET.PF0.VF0            VF     pci_1
/SYS/MB/PCIE5/IOVNET.PF0.VF1            VF     pci_1
/SYS/MB/NET2/IOVNET.PF1.VF0             VF     pci_1
/SYS/MB/NET2/IOVNET.PF1.VF1             VF     pci_1
```

La commande suivante ajoute de façon dynamique la fonction virtuelle `/SYS/MB/PCIE5/IOVNET.PF0.VF1` au domaine root `ldg1` non `primary` :

```
primary# ldm add-io /SYS/MB/PCIE5/IOVNET.PF0.VF1 ldg1
```

La commande suivante ajoute de façon dynamique la fonction virtuelle `/SYS/MB/NET2/IOVNET.PF1.VF0` au domaine `ldg2` :

```
primary# ldm add-io /SYS/MB/NET2/IOVNET.PF1.VF0 ldg2
```

La commande suivante ajoute la fonction virtuelle `/SYS/MB/NET2/IOVNET.PF1.VF1` au domaine `ldg3` lié :

```
primary# ldm add-io /SYS/MB/NET2/IOVNET.PF1.VF1 ldg3
primary# ldm start ldg3
LDom ldg3 started
```

Connectez-vous à la console du domaine `ldg3` puis initialisez son SE.

La sortie suivante indique que toutes les assignations s'affichent comme prévu. Une fonction virtuelle n'est pas attribuée et peut donc être attribuée de façon dynamique au domaine root `ldg1`, `ldg2` ou `ldg3`.

```
# ldm list-io
NAME                                    TYPE   BUS      DOMAIN   STATUS
----                                    ----   ---      ------   ------
pci_0                                   BUS    pci_0    primary  IOV
pci_1                                   BUS    pci_1    ldg1     IOV
niu_0                                   NIU    niu_0    primary
niu_1                                   NIU    niu_1    primary
/SYS/MB/PCIE0                           PCIE   pci_0    primary  OCC
/SYS/MB/PCIE2                           PCIE   pci_0    primary  OCC
/SYS/MB/PCIE4                           PCIE   pci_0    primary  OCC
/SYS/MB/PCIE6                           PCIE   pci_0    primary  EMP
/SYS/MB/PCIE8                           PCIE   pci_0    primary  EMP
/SYS/MB/SASHBA                          PCIE   pci_0    primary  OCC
/SYS/MB/NET0                            PCIE   pci_0    primary  OCC
/SYS/MB/PCIE1                           PCIE   pci_1    ldg1     OCC
/SYS/MB/PCIE3                           PCIE   pci_1    ldg1     OCC
/SYS/MB/PCIE5                           PCIE   pci_1    ldg1     OCC
/SYS/MB/PCIE7                           PCIE   pci_1    ldg2     OCC
/SYS/MB/PCIE9                           PCIE   pci_1    ldg1     EMP
/SYS/MB/NET2                            PCIE   pci_1    ldg1     OCC
/SYS/MB/NET0/IOVNET.PF0                 PF     pci_0    primary
/SYS/MB/NET0/IOVNET.PF1                 PF     pci_0    primary
/SYS/MB/PCIE5/IOVNET.PF0                PF     pci_1    ldg1
/SYS/MB/PCIE5/IOVNET.PF1                PF     pci_1    ldg1
/SYS/MB/NET2/IOVNET.PF0                 PF     pci_1    ldg1
/SYS/MB/NET2/IOVNET.PF1                 PF     pci_1    ldg1
```

```
/SYS/MB/PCIE5/IOVNET.PF0.VF0             VF     pci_1
/SYS/MB/PCIE5/IOVNET.PF0.VF1             VF     pci_1    ldg1
/SYS/MB/NET2/IOVNET.PF1.VF0              VF     pci_1    ldg2
/SYS/MB/NET2/IOVNET.PF1.VF1              VF     pci_1    ldg3
```

♦ ♦ ♦ CHAPITRE 11

Utilisation des disques virtuels

Ce chapitre décrit comment utiliser les disques virtuels avec le logiciel Oracle VM Server for SPARC.

Ce chapitre aborde les sujets suivants :

Présentation des disques virtuels

Un disque virtuel contient deux composants : le disque virtuel lui-même tel qu'il apparaît dans un domaine invité et le backend du disque virtuel, qui est l'endroit où les données sont stockées et où l'E/S virtuelle est envoyée. Le backend du disque virtuel est exporté à partir d'un domaine de service vers le pilote du serveur de disque virtuel (`vds`). Le pilote `vds` communique avec le

pilote du client de disque virtuel (`vdc`) dans le domaine invité via l'hyperviseur à l'aide d'un canal de domaine logique (LDC). Enfin, un disque virtuel apparaît en tant que périphériques `/dev/[r]dsk/cXdYsZ` dans le domaine invité.

Remarque - Vous pouvez vous référer à un disque, à l'aide soit de `/dev/dsk` soit de `/dev/rdsk`, comme partie du nom du chemin d'accès au disque. Les deux références donnent le même résultat.

Attention - N'utilisez pas le périphérique `d0` pour représenter l'intégralité du disque. Ce périphérique représente uniquement le disque entier quand ce dernier dispose d'une étiquette EFI et non VTOC. L'utilisation du périphérique `d0` crée un disque virtuel à tranche unique, ce qui peut provoquer l'endommagement de l'étiquette si vous écrivez sur le début du disque.

Utilisez plutôt la tranche `s2` pour virtualiser l'ensemble du disque. La tranche `s2` est indépendante de l'étiquette.

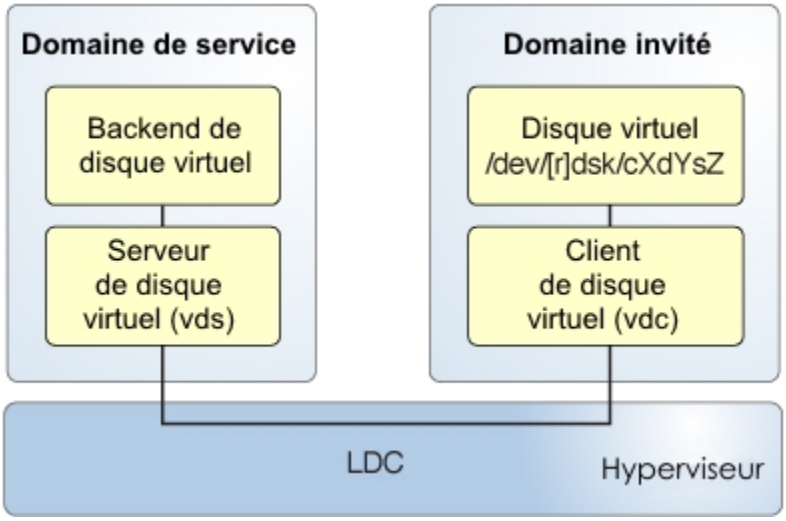
Le backend du disque virtuel peut être physique ou logique. Les périphériques physiques peuvent comprendre :

- Un disque physique ou un numéro d'unité logique (LUN)
- Une tranche de disque physique

Les périphériques logiques peuvent être les suivants :

- Un fichier sur un système de fichiers local, tel que ZFS ou UFS par exemple, ou un système de fichiers distant rendu disponible via NFS
- Un volume logique à partir d'un gestionnaire de volumes, notamment ZFS, VxVM ou Solaris Volume Manager
- Tout pseudopériphérique de disque accessible à partir du domaine de service

FIGURE 6 Disques virtuels avec Oracle VM Server for SPARC



Pour utiliser le nombre maximal de disques virtuels sur le serveur, assurez-vous que le noyau `segkpsize` réglable ait une valeur d'au moins 524288. Notez qu'une valeur insuffisante pour `segkpsize` peut provoquer le blocage d'un domaine invité lors de l'initialisation ou de l'ajout dynamique d'un disque virtuel. Pour plus d'informations sur `segkpsize`, voir "segkpsize" du manuel *Oracle Solaris 11.3 Tunable Parameters Reference Manual*.

Identificateur de disque virtuel et nom de périphérique

Lorsque vous utilisez la commande `ldm add-vdisk` pour ajouter un disque virtuel à un domaine, vous pouvez définir son numéro de périphérique en définissant la propriété `id`.

`ldm add-vdisk [id=`*disk-id*`]` *disk-name volume-name*@*service-name domain-name*

Chaque disque virtuel d'un domaine a un numéro de périphérique unique qui est assigné lorsque le domaine est lié. Si un disque virtuel a été ajouté avec un numéro de périphérique explicite (en définissant la propriété `id`), le numéro de périphérique défini est utilisé. Sinon, le système assigne automatiquement le numéro de périphérique le plus petit disponible. Dans ce cas, le numéro de périphérique assigné dépend de la manière dont les disques virtuels ont été ajoutés au domaine. Le numéro de périphérique qui fini par être assigné à un disque virtuel est visible dans la sortie de la commande `ldm list-bindings` lorsqu'un domaine est lié.

Lorsqu'un domaine avec des disques virtuels exécute le SE Oracle Solaris, chaque disque virtuel apparaît dans le domaine comme un périphérique de disque `c0d`*n*, où *n* est le numéro de périphérique du disque virtuel.

Dans l'exemple suivant, le domaine `ldg1` a deux disques virtuels : `rootdisk` et `pdisk`. `rootdisk` a le numéro de périphérique `0` (`disk@0`) et apparaît dans le domaine en tant que périphérique de disque `c0d0`. `pdisk` a le numéro de périphérique `1` (`disk@1`) et apparaît dans le domaine en tant que périphérique de disque `c0d1`.

```
primary# ldm list-bindings ldg1
...
DISK
    NAME             VOLUME                          TOUT DEVICE  SERVER         MPGROUP
    rootdisk         dsk_nevada@primary-vds0              disk@0  primary
    pdisk            c3t40d1@primary-vds0                 disk@1  primary
...
```

Attention - Si un numéro de périphérique n'est pas assigné explicitement à un disque virtuel, son numéro de périphérique peut changer lorsque le domaine est dissocié, puis est réassocié ultérieurement. Dans ce cas, le nom de périphérique assigné par le SE s'exécutant dans le domaine peut également changer et rompre la configuration existante du système. Cela peut se produite, par exemple, lorsqu'un disque virtuel est supprimé de la configuration du domaine.

Gestion des disques virtuels

Cette section décrit l'ajout d'un disque virtuel à un domaine invité, la modification des options d'un disque virtuel et de délai d'attente et la suppression d'un disque virtuel d'un domaine invité. Reportez-vous à la section "Options de moteur de traitement du disque virtuel" à la page 190 pour obtenir une description des options du disque virtuel. Reportez-vous à la section "Délai d'attente du disque virtuel" à la page 209 pour obtenir une description du délai d'attente du disque virtuel.

Le backend d'un disque virtuel peut être exporté plusieurs fois par le même serveur de disque virtuel ou par des serveurs différents. Chaque instance exportée du backend du disque virtuel peut ensuite être assignée au même domaine invité ou à des domaines invités différents.

Lorsque le backend d'un disque virtuel est exporté plusieurs fois, il ne doit pas être exporté avec l'option exclusive (`excl`). La définition de l'option `excl` ne permettra l'exportation du backend qu'une seule fois. Le backend peut être exporté en toute sécurité plusieurs fois en tant que périphérique en lecture seule avec l'option `ro`.

L'assignation d'un périphérique de disque virtuel à un domaine crée une dépendance implicite sur le domaine fournissant le service de disque virtuel. Vous pouvez afficher ces dépendances

ou domaines dépendant du service de disque virtuel à l'aide de la commande `ldm list-dependencies`. Voir "Liste des dépendances d'E/S de domaine" à la page 430.

▼ Procédure d'ajout d'un disque virtuel

1. **Exportez le backend du disque virtuel à partir d'un domaine de service.**

   ```
   ldm add-vsdev [-fq] [options={ro,slice,excl}] [mpgroup=mpgroup] \
   backend  volume-name@service-name
   ```

2. **Assignez le backend à un domaine invité.**

   ```
   ldm add-vdisk [timeout=seconds] [id=disk-id] disk-name  volume-name@service-name  domain-name
   ```

   Vous pouvez spécifier un ID personnalisé pour le nouveau périphérique de disque virtuel en définissant la propriété `id`. Par défaut, les valeurs d'ID sont générées automatiquement. Par conséquent, définissez cette propriété si vous devez faire correspondre un nom de périphérique existant dans le SE. Reportez-vous à la section "Identificateur de disque virtuel et nom de périphérique" à la page 185.

Remarque - Un backend est généralement exporté à partir du domaine de service et assigné au domaine invité lorsque le domaine invité (*domain-name*) est lié.

▼ Procédure d'exportation multiple du backend d'un disque virtuel

Attention - Lorsque le backend d'un disque virtuel est exporté plusieurs fois, les applications s'exécutant sur les domaines invités et utilisant ce disque virtuel sont responsables de la coordination et de la synchronisation de l'accès simultané en écriture afin de garantir la cohérence des données.

L'exemple suivant décrit comment ajouter le même disque virtuel à deux domaines invités différents via le même serveur de disque virtuel.

1. **Exportez le backend du disque virtuel deux fois à partir d'un domaine de service.**

   ```
   ldm add-vsdev [options={ro,slice}] backend volume1@service-name
   # ldm add-vsdev -f [options={ro,slice}] backend volume2@service-name
   ```

   Notez que la seconde commande `ldm add-vsdev` utilise l'option `-f` pour forcer la seconde exportation du backend. Utilisez cette option lorsque vous utilisez le même chemin d'accès

backend pour les deux commandes et lorsque les serveurs de disque virtuel sont situés sur le même domaine de service.

2. **Assignez le backend exporté à chaque domaine invité.**
disk-name peut être différent pour `ldom1` et `ldom2`.

```
ldm add-vdisk [timeout=seconds] disk-name volume1@service-name ldom1
# ldm add-vdisk [timeout=seconds] disk-name volume2@service-name ldom2
```

Procédure de modification des options du disque virtuel

Pour plus d'informations sur les options du disque virtuel, reportez-vous à la section "Options de moteur de traitement du disque virtuel" à la page 190.

- **Après l'exportation d'un backend à partir du domaine de service, vous pouvez modifier les options du disque virtuel.**

```
primary# ldm set-vdsdev options=[{ro,slice,excl}] volume-name@service-name
```

Procédure de modification de l'option de délai d'attente

Pour plus d'informations sur les options du disque virtuel, reportez-vous à la section "Options de moteur de traitement du disque virtuel" à la page 190.

- **Après l'assignation d'un disque virtuel à un domaine invité, vous pouvez modifier le délai d'attente du disque virtuel.**

```
primary# ldm set-vdisk timeout=seconds disk-name domain-name
```

Procédure de suppression d'un disque virtuel

1. **Supprimez un disque virtuel d'un domaine invité.**

```
primary# ldm rm-vdisk disk-name domain-name
```

2. **Arrêtez l'exportation du backend correspondant à partir du domaine de service.**

```
primary# ldm rm-vdsdev volume-name@service-name
```

Apparence d'un disque virtuel

Lorsqu'un backend est exporté en tant que disque virtuel, il peut apparaître dans le domaine invité en tant que disque complet ou sous forme de disque à tranche unique. La manière dont il apparaît dépend du type de moteur de traitement et des options utilisées pour l'exportation.

Remarque - Le stockage NVMe (Non-Volatile Memory Express) est disponible à partir des séries SPARC T7 et SPARC M7. Il peut s'agir d'une unité de disque ou d'une carte PCIe Flash Accelerator F160. Ce type de disque peut être utilisé pour la création d'un backend de disque virtuel.

A compter du SE Oracle Solaris 11.3 SRU 2.4, vous pouvez utiliser le type de disque de stockage NVMe comme un disque complet ou un disque à tranche unique.

Avant le SE Oracle Solaris 11.3 SRU 2.4, vous ne pouviez utiliser le type de disque de stockage NVMe que comme un disque à tranche unique.

Attention - Les disques à tranche unique n'ont pas d'ID de périphérique. Si l'ID est obligatoire utilisez un backend de disque physique complet.

Disque complet

Lorsqu'un backend est exporté sur un domaine en tant que disque complet, il apparaît dans ce domaine comme un disque normal avec 8 tranches (`s0` à `s7`). Ce type de disque est visible avec la commande `format`(1M). La table de partition du disque peut être modifiée à l'aide de la commande `fmthard` ou de la commande `format`.

Un disque complet est également visible au logiciel d'installation du SE et peut être sélectionné en tant que disque sur lequel le SE peut être installé.

Tout backend peut être exporté en tant que disque complet sauf pour les tranches de disque physique qui peuvent uniquement être exportées comme des disques à tranche unique.

Disque à tranche unique

Lorsqu'un backend est exporté sur un domaine en tant que disque à tranche unique, il apparaît dans ce domaine comme un disque normal avec 8 tranches (`s0` à `s7`). Cependant, seule la

première tranche (`s0`) est utilisable. Ce type de disque est visible avec la commande `format`(1M), mais la table de partition du disque ne peut pas être modifiée.

Un disque à tranche unique est également visible à partir du logiciel d'installation du SE et peut être sélectionné comme disque sur lequel vous pouvez installer le SE. Dans ce cas, si vous installez le SE à l'aide du système de fichiers UNIX (UFS), seule la partition root (`/`) doit être définie, et cette partition doit utiliser tout l'espace disque.

Tout backend peut être exporté en tant que disque à tranche unique sauf pour les disques physiques qui peuvent uniquement être exportés comme des disques complets.

Remarque - Avant la version du SE Oracle Solaris 10 10/08, un disque à tranche unique apparaissait comme un disque avec une seule partition (`s0`). Ce type de disque n'était pas visible à l'aide de la commande `format`. Le disque n'était également pas visible à partir du logiciel d'installation du SE et ne pouvait pas être sélectionné comme périphérique de disque sur lequel installer le SE.

Options de moteur de traitement du disque virtuel

Différentes options peuvent être définies lors de l'exportation du backend d'un disque virtuel. Ces options sont indiquées dans l'argument `options=` de ma commande `ldm add-vdsdev` sous forme de liste délimitée par des virgules. Les options valides sont : `ro`, `slice` et `excl`.

Option de lecture seule (`ro`)

L'option de lecture seule (`ro`) indique que le backend doit être exporté comme un périphérique en lecture seule. Dans ce cas, le disque virtuel assigné au domaine invité est uniquement accessible pour les opérations de lecture et toute opération d'écriture sur le disque virtuel est vouée à l'échec.

Option exclusive (`excl`)

L'option exclusive (`excl`) indique que le backend du domaine de service doit être ouvert exclusivement par le serveur de disque virtuel lorsqu'il est exporté en tant que disque virtuel vers un autre domaine. Lorsqu'un backend est ouvert en exclusivité, il n'est pas accessible par les autres applications du domaine de service. Cette restriction empêche les applications

s'exécutant sur le domaine de service d'utiliser par inadvertance un backend également utilisé par un domaine invité.

Remarque - Certains pilotes ne remplissent pas l'option `excl` et n'autoriseront pas certains moteurs de traitement de disque virtuel à s'ouvrir en exclusivité. L'option `excl` est connue pour fonctionner avec les disques physiques et les tranches, mais l'option ne fonctionne pas sur les fichiers. Elle est susceptible de fonctionner avec les pseudopériphériques, tels que les volumes de disque. Si le pilote du backend n'effectue pas l'ouverture exclusive, l'option `excl` du backend est ignorée et celui-ci n'est pas ouvert en exclusivité.

Comme l'option `excl` empêche les applications s'exécutant dans le domaine de service d'accéder à un backend exporté sur un domaine invité, ne définissez pas l'option `excl` dans les cas suivants :

- Lorsque les domaines invités sont en cours d'exécution, si vous souhaitez utiliser des commandes telles que `format` pour gérer les disques physiques, n'exportez pas ces disques avec l'option `excl`.
- Lorsque vous exportez un volume Solaris Volume Manager, tel qu'un volume RAID ou en miroir, ne définissez pas l'option `excl`. Sinon, cela peut empêcher Solaris Volume Manager de démarrer certaines opérations de récupération si un composant du volume RAID ou en miroir échoue. Pour plus d'informations, reportez-vous à la section "Utilisation des disques virtuels avec Solaris Volume Manager" à la page 217
- Si Veritas Volume Manager (VxVM) est installé dans le domaine de service et que Veritas Dynamic Multipathing (VxDMP) est activé pour les disques physiques, ceux-ci doivent être exportés sans l'option `excl` (n'est pas la valeur par défaut). Sinon, l'exportation échoue, car le serveur de disque virtuel (`vds`) est inutilisable pour ouvrir le périphérique de disque virtuel. Reportez-vous à la section "Utilisation des disques virtuels lorsque VxVM est installé" à la page 218 pour plus d'informations.
- Si vous exportez le même backend de disque virtuel plusieurs fois à partir du même service de disque virtuel, reportez-vous à la section "Procédure d'exportation multiple du backend d'un disque virtuel" à la page 187 pour plus d'informations.

Par défaut, le backend n'est pas ouvert en exclusivité. De cette manière, le backend peut toujours être utilisé par les applications s'exécutant dans le domaine de service lorsqu'il est exporté dans un autre domaine.

Option de tranche (`slice`)

Un backend est généralement exporté comme un disque complet ou un disque à tranche unique en fonction de son type. Si l'option `slice` est indiquée, le backend est exporté de force comme un disque à tranche unique.

Cette option est utile lorsque vous voulez exporter le contenu brut d'un backend. Par exemple, si vous avez un volume ZFS ou Solaris Volume Manager sur lequel vous avez déjà stocké des données et si vous voulez que votre domaine invité accède à ces données, vous devez exporter le volume ZFS ou Solaris Volume Manager à l'aide de l'option `slice`.

Pour plus d'informations sur cette option, reportez-vous à la section "Backend du disque virtuel" à la page 192.

Backend du disque virtuel

Le backend du disque virtuel est l'emplacement où les données d'un disque virtuel sont stockées. Le backend peut être un disque, une tranche de disque, un fichier ou un volume tel que ZFS, Solaris Volume Manager ou VxVM. Un backend apparaît dans un domaine invité comme un disque complet ou un disque à tranche unique, selon que l'option `slice` a été définie ou non lorsque le backend a été exporté à partir du domaine de service. Par défaut, un backend de disque virtuel est exporté de manière non exclusive en tant que disque complet accessible en lecture et en écriture.

Disque physique ou LUN de disque

Un disque physique ou un LUN de disque est toujours exporté comme un disque complet. Dans ce cas, les pilotes de disque virtuel (`vds` et `vdc`) transmettent des E/S du disque virtuel et agissent comme une intercommunication vers le disque virtuel et le LUN de disque.

Un disque physique ou un LUN de disque est exporté à partir d'un domaine de service en exportant le périphérique qui correspond à la tranche 2 (`s2`) de ce disque sans définir l'option `slice`. Si vous exportez la tranche 2 d'un disque avec l'option `slice`, seule cette tranche est exportée et non pas le disque entier.

▼ Procédure d'exportation d'un disque physique en tant que disque virtuel

Attention - Lorsque vous configurez des disques virtuels, assurez-vous que chaque disque virtuel se rapporte à des ressources physiques distinctes (backend), tels que des disques physiques, des tranches de disque, des fichiers ou des volumes.

Certains disques, tels que FibreChannel et SAS, ont une nature "double-port", ce qui signifie qu'un disque peut être référencé par deux chemins différents. Assurez-vous que les chemins que vous affectez à différents domaines ne se rapportent pas au même disque physique.

1. **Exportez un disque physique en tant que disque virtuel.**

   Par exemple, pour exporter le disque physique `c1t48d0` en tant que disque virtuel, vous devez exporter la tranche 2 de ce disque (`c1t48d0s2`).

   ```
   primary# ldm add-vdsdev /dev/dsk/c1t48d0s2 c1t48d0@primary-vds0
   ```

2. **Assignez le disque à un domaine invité.**

   Par exemple, assignez le disque (`pdisk`) au domaine invité `ldg1`.

   ```
   primary# ldm add-vdisk pdisk c1t48d0@primary-vds0 ldg1
   ```

3. **Après que le domaine invité est démarré et exécute le SE Oracle Solaris, vérifiez que le disque est accessible et est un disque complet.**

   Un disque complet est un disque normal qui a huit (8) tranches.

   Par exemple, le disque en cours de vérification est `c0d1`.

   ```
   ldg1# ls -1 /dev/dsk/c0d1s*
   /dev/dsk/c0d1s0
   /dev/dsk/c0d1s1
   /dev/dsk/c0d1s2
   /dev/dsk/c0d1s3
   /dev/dsk/c0d1s4
   /dev/dsk/c0d1s5
   /dev/dsk/c0d1s6
   /dev/dsk/c0d1s7
   ```

Tranche de disque physique

Une tranche de disque physique est toujours exportée comme un disque à tranche unique. Dans ce cas, les pilotes de disque virtuel (`vds` et `vdc`) transmettent des E/S du disque virtuel et agissent comme une intercommunication vers la tranche de disque virtuel.

Une tranche de disque physique est exportée d'un domaine de service en exportant le périphérique de tranche correspondant. Si le périphérique est différent de la tranche 2, il est automatiquement exporté comme un disque à tranche unique, que vous indiquiez ou non l'option `slice`. Si le périphérique est la tranche 2 du disque, vous devez définir l'option `slice` pour exporter uniquement la tranche 2 en tant que disque à tranche unique. Sinon, le disque entier est exporté en tant que disque complet.

Remarque - Le stockage NVMe (Non-Volatile Memory Express) est disponible à partir des séries SPARC T7 et SPARC M7. Il peut s'agir d'une unité de disque ou d'une carte PCIe Flash Accelerator F160.

A compter du SE Oracle Solaris 11.3 SRU 2.4, vous pouvez utiliser le type de disque de stockage NVMe comme un disque complet ou un disque à tranche unique.

Avant le SE Oracle Solaris 11.3 SRU 2.4, vous ne pouviez utiliser le type de disque de stockage NVMe que comme un disque à tranche unique.

▼ Procédure d'exportation d'une tranche de disque physique en tant que disque virtuel

1. **Exportez une tranche de disque physique en tant que disque virtuel.**

   Par exemple, pour exporter la tranche 0 d'un disque physique `c1t57d0` en tant que disque virtuel, vous devez exporter le périphérique qui correspond à cette tranche (`c1t57d0s0`) comme suit.

   ```
   primary# ldm add-vsdev /dev/dsk/c1t57d0s0 c1t57d0s0@primary-vds0
   ```

   Il est inutile de spécifier l'option `slice` car une tranche est toujours exportée en tant que disque à tranche unique.

2. **Assignez le disque à un domaine invité.**

   Par exemple, assignez le disque (`pslice`) au domaine invité `ldg1`.

   ```
   primary# ldm add-vdisk pslice c1t57d0s0@primary-vds0 ldg1
   ```

3. **Après que le domaine invité est démarré et exécute le SE Oracle Solaris, vous pouvez répertorier le disque (`c0d13`, par exemple) et voir que le disque est accessible.**

   ```
   ldg1# ls -1 /dev/dsk/c0d13s*
   /dev/dsk/c0d13s0
   /dev/dsk/c0d13s1
   /dev/dsk/c0d13s2
   /dev/dsk/c0d13s3
   ```

```
/dev/dsk/c0d13s4
/dev/dsk/c0d13s5
/dev/dsk/c0d13s6
/dev/dsk/c0d13s7
```

Bien qu'il y ait 8 périphériques, car le disque est un disque à tranche unique, seule la première tranche (`s0`) est utilisable.

▼ Procédure d'exportation de la tranche 2

- **Pour exporter la tranche 2 (le disque `c1t57d0s2` par exemple), vous devez spécifier l'option `slice`. Dans le cas contraire, le disque entier est exporté.**

```
primary# ldm add-vdsdev options=slice /dev/dsk/c1t57d0s2 c1t57d0s2@primary-vds0
```

Exportation de fichier et de volume

Un fichier ou un volume (provenant de ZFS ou Solaris Volume Manager par exemple) est exporté soit comme un disque complet, soit comme un disque à tranche unique, selon que vous définissiez ou non l'option `slice`.

Fichier ou volume exporté en tant que disque complet

Si vous ne définissez pas l'option `slice`, un fichier ou un volume est exporté en tant que disque complet. Dans ce cas, les pilotes de disque virtuel (`vds` et `vdc`) transmettent les E/S à partir du disque virtuel et gèrent le partitionnement du disque virtuel. Le fichier ou le volume devient une image de disque contenant les données de toutes les tranches du disque virtuel et les métadonnées utilisées pour gérer le partitionnement et la structure du disque.

Si un fichier ou un volume vide est exporté en tant que disque complet, il apparaît dans le domaine invité comme un disque non formaté, c'est-à-dire un disque sans partition. Vous devez ensuite exécuter la commande `format` dans le domaine invité pour définir les partitions utilisables et pour écrire une étiquette de disque valide. Toutes les E/S vers le disque virtuel échouent lorsque le disque n'est pas formaté.

Remarque - Vous devez exécuter la commande `format` dans le domaine invité pour créer des partitions.

▼ Procédure d'exportation d'un fichier en tant que disque complet

1. **A partir du domaine de service, créez un fichier (`fdisk0` par exemple) à utiliser comme disque virtuel.**

   ```
   service# mkfile 100m /ldoms/domain/test/fdisk0
   ```

   La taille du fichier définit la taille du disque virtuel. Cet exemple crée un fichier vide de 100 Mo pour obtenir un disque virtuel de 100 Mo.

2. **A partir du domaine de contrôle, exportez le fichier en tant que disque virtuel.**

   ```
   primary# ldm add-vdsdev /ldoms/domain/test/fdisk0 fdisk0@primary-vds0
   ```

   Dans cet exemple, l'option `slice` n'est pas définie. Par conséquent, le fichier est exporté comme un disque complet.

3. **A partir du domaine de contrôle, assignez le disque à un domaine invité.**

   Par exemple, assignez le disque (`fdisk`) au domaine invité `ldg1`.

   ```
   primary# ldm add-vdisk fdisk fdisk0@primary-vds0 ldg1
   ```

4. **Après que le domaine invité est démarré et exécute le SE Oracle Solaris, vérifiez que le disque est accessible et est un disque complet.**

   Un disque complet est un disque normal avec huit tranches.

   L'exemple suivant montre comment répertorier le disque, `c0d5` et vérifier qu'il est accessible et est un disque complet.

   ```
   ldg1# ls -1 /dev/dsk/c0d5s*
   /dev/dsk/c0d5s0
   /dev/dsk/c0d5s1
   /dev/dsk/c0d5s2
   /dev/dsk/c0d5s3
   /dev/dsk/c0d5s4
   /dev/dsk/c0d5s5
   /dev/dsk/c0d5s6
   /dev/dsk/c0d5s7
   ```

▼ Procédure d'exportation d'un volume ZFS en tant que disque complet

1. **Créez un volume ZFS à utiliser en tant que disque complet.**

   L'exemple suivant montre comment créer un volume ZFS, `zdisk0`, à utiliser comme disque complet.

```
service# zfs create -V 100m ldoms/domain/test/zdisk0
```

La taille du volume définit la taille du disque virtuel. Cet exemple crée un volume de 100 Mo pour obtenir un disque virtuel de 100 Mo.

2. **A partir du domaine de contrôle, exportez le périphérique correspondant vers le volume ZFS.**

```
primary# ldm add-vdsdev /dev/zvol/dsk/ldoms/domain/test/zdisk0 \
zdisk0@primary-vds0
```

Dans cet exemple, l'option `slice` n'est pas définie. Par conséquent, le fichier est exporté comme un disque complet.

3. **A partir du domaine de contrôle, assignez le volume à un domaine invité.**

L'exemple suivant indique comment assigner le volume `zdisk0` au domaine invité `ldg1` :

```
primary# ldm add-vdisk zdisk0 zdisk0@primary-vds0 ldg1
```

4. **Après que le domaine invité est démarré et exécute le SE Oracle Solaris, vérifiez que le disque est accessible et est un disque complet.**

Un disque complet est un disque normal avec huit tranches.

L'exemple suivant montre comment répertorier le disque `c0d9` et vérifier qu'il est accessible et est un disque complet :

```
ldg1# ls -1 /dev/dsk/c0d9s*
/dev/dsk/c0d9s0
/dev/dsk/c0d9s1
/dev/dsk/c0d9s2
/dev/dsk/c0d9s3
/dev/dsk/c0d9s4
/dev/dsk/c0d9s5
/dev/dsk/c0d9s6
/dev/dsk/c0d9s7
```

Fichier ou volume exporté en tant que disque à tranche unique

Si l'option `slice` est définie, le fichier ou le volume est exporté en tant que disque à tranche unique. Dans ce cas, le disque virtuel n'a qu'une seule partition (`s0`), qui est directement mappée sur le backend du fichier ou du volume. Le fichier ou le volume ne contient que des données écrites sur le disque virtuel avec données supplémentaires telles que des informations de partitionnement ou la structure du disque.

Si un fichier ou un volume est exporté en tant que disque à tranche unique, le système simule un faux partitionnement de disque qui fait apparaître ce fichier ou ce volume en tant que tranche de

disque. Comme le partitionnement de disque est simulé, vous ne créez pas de partitionnement pour ce disque.

▼ Procédure d'exportation d'un volume ZFS en tant que disque à tranche unique

1. **Créez un volume ZFS à utiliser en tant que disque à tranche unique.**

   L'exemple suivant montre comment créer un volume ZFS, `zdisk0`, à utiliser comme disque à tranche unique.

   ```
   service# zfs create -V 100m ldoms/domain/test/zdisk0
   ```

   La taille du volume définit la taille du disque virtuel. Cet exemple crée un volume de 100 Mo pour obtenir un disque virtuel de 100 Mo.

2. **A partir du domaine de contrôle, exportez le périphérique correspondant sur ce volume ZFS et définissez l'option `slice` afin que le volume soit exporté comme un disque à tranche unique.**

   ```
   primary# ldm add-vdsdev options=slice /dev/zvol/dsk/ldoms/domain/test/zdisk0 \
   zdisk0@primary-vds0
   ```

3. **A partir du domaine de contrôle, assignez le volume à un domaine invité.**

   Voici comment assigner le volume, `zdisk0`, au domaine invité `ldg1`.

   ```
   primary# ldm add-vdisk zdisk0 zdisk0@primary-vds0 ldg1
   ```

4. **Une fois que le domaine invité a démarré et exécute le SE Oracle Solaris, vous pouvez répertorier le disque (`c0d9`, par exemple) et voir que le disque est accessible et est un disque à tranche unique (`s0`).**

   ```
   ldg1# ls -1 /dev/dsk/c0d9s*
   /dev/dsk/c0d9s0
   /dev/dsk/c0d9s1
   /dev/dsk/c0d9s2
   /dev/dsk/c0d9s3
   /dev/dsk/c0d9s4
   /dev/dsk/c0d9s5
   /dev/dsk/c0d9s6
   /dev/dsk/c0d9s7
   ```

Exportation de volumes et rétrocompatibilité

Si vous avez une configuration exportant les volumes en tant que disques virtuels, les volumes sont maintenant exportés comme des disques complets plutôt que comme des disques à tranche

unique. Pour préserver l'ancien comportement et que vos volumes soient exportés comme des disques à tranche unique, vous devez effectuer l'une des opérations suivantes :

- Utilisez la commande `ldm set-vdsdev` dans le logiciel Oracle VM Server for SPARC 3.4 et définissez l'option `slice` pour tous les volumes que vous voulez exporter en tant que disques à tranche unique. Reportez-vous à la page de manuel `ldm`(1M).
- Ajoutez la ligne suivante au fichier `/etc/system` sur le domaine de service.

  ```
  set vds:vd_volume_force_slice = 1
  ```

  Pour plus d'informations sur la création ou la mise à jour des valeurs de propriété `/etc/system`, reportez-vous à la section "Mise à jour des valeurs de propriété dans le fichier `/etc/system`" à la page 409.

Remarque - Ce paramétrage force l'exportation de tous les volumes en tant que disques à tranche unique et vous ne pouvez pas exporter de volume en tant que disque complet.

Récapitulatif des méthodes d'exportation des différents types de moteurs de traitement

Backend	Sans option de segmentation	Option de segmentation définie
Disque (tranche de disque 2)	Disque complet[†]	Disque à tranche unique[‡]
Tranche de disque (pas la tranche 2)	Disque à tranche unique[*]	Disque à tranche unique
Fichier	Disque complet	Disque à tranche unique
Volume, y compris ZFS, Solaris Volume Manager ou VxVM	Disque complet	Disque à tranche unique

[†]Exportez tout le disque.

[‡]Exportez uniquement la tranche 2.

[*]Une tranche est toujours exportée en tant que disque à tranche unique.

Consignes d'exportation des fichiers et des tranches de fichiers en tant que disques virtuels

Cette section comprend des consignes pour l'exportation d'un fichier et d'une tranche de disque en tant que disque virtuel.

Utilisation du pilote de fichier loopback (`lofi`)

L'utilisation du pilote de fichier loopback (`lofi`) pour exporter un fichier en tant que disque virtuel ajoute une couche de pilote supplémentaire et a un impact sur les performances du disque virtuel. Au lieu de cela, vous pouvez exporter directement un fichier en tant que disque complet ou en tant que disque à fragment unique. Reportez-vous à la section "Exportation de fichier et de volume" à la page 195.

Exportation directe ou indirecte d'une tranche de disque

Pour exporter une tranche en tant que disque virtuel directement ou indirectement (par exemple via un volume Solaris Volume Manager), vérifiez que la tranche ne commence pas au premier bloc (bloc 0) du disque physique à l'aide de la commande `prtvtoc`.

Si vous exportez directement ou indirectement une tranche de disque qui commence sur le premier bloc d'un disque physique, vous risquez d'écraser la table des partitions du disque physique et de rendre toutes les partitions de ce disque inaccessibles.

Configuration de la fonctionnalité de chemins d'accès multiples d'un disque virtuel

Le *multipathing de disque virtuel* vous permet de configurer un disque virtuel sur un domaine invité pour accéder à son stockage backend par plusieurs chemins. Les chemins traversent différents domaines de service qui fournissent l'accès au même stockage backend, notamment un LUN de disque. Cette fonction permet à un disque virtuel dans un domaine invité de rester accessible même si l'un des domaines de service est en panne. Par exemple, vous pouvez définir une configuration de fonctionnalité multipathing de disque virtuel pour accéder à un fichier ou à un serveur de système de fichier réseau (NFS). Vous pouvez également utiliser cette configuration pour accéder à un LUN à partir d'un stockage partagé qui est connecté à plusieurs domaines de service. Par conséquent, lorsque le domaine invité accède au disque virtuel, le pilote du disque virtuel traverse l'un des domaines de service pour accéder au stockage backend. Si le pilote du disque virtuel peut se connecter au domaine de service, le disque virtuel tente d'atteindre le stockage backend via un domaine de service différent.

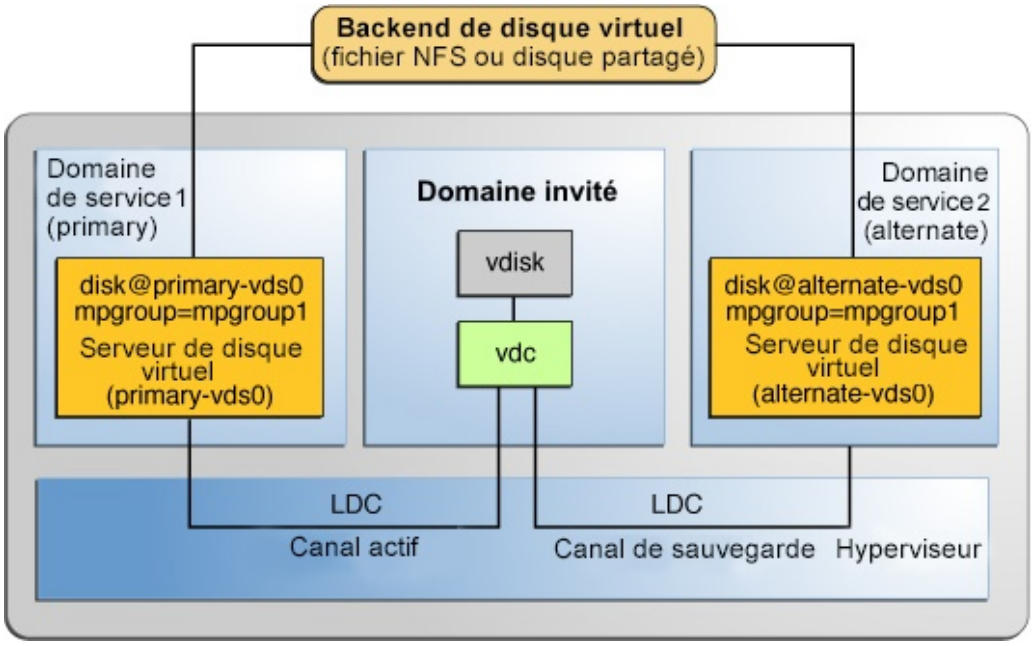
Remarque - Vous ne pouvez pas utiliser des mpgroups et des réservations SCSI ensemble.

Notez que la fonctionnalité de chemins d'accès multiples de disque virtuel peut détecter si un domaine de service ne peut pas accéder au stockage backend. Dans ce cas, le pilote du disque virtuel tente d'accéder au stockage backend par un autre chemin.

Pour activer la fonctionnalité multipathing de disque virtuel, vous devez exporter un backend de disque virtuel à partir de chaque domaine de service et ajouter le disque virtuel au même groupe multipathing (`mpgroup`). `mpgroup` est identifié par un nom et configuré lorsque vous exportez le backend du disque virtuel.

La figure suivante représente une configuration de fonctionnalité multipathing de disque virtuel qui est utilisée comme exemple dans la procédure "Procédure de configuration de la fonctionnalité multipathing de disque virtuel" à la page 202. Dans cet exemple, un groupe de chemins d'accès multiples nommé `mpgroup1` est utilisé pour créer un disque virtuel, dont le backend est accessible à partir de deux domaines de service : `primary` et `alternate`.

FIGURE 7 Configuration de la fonctionnalité de chemins d'accès multiples d'un disque virtuel



Fonctionnalité de chemins d'accès multiples de disque virtuel et NFS

Quand vous configurez le basculement mpgroup avec des systèmes de fichiers NFS, veillez à monter le système de fichiers NFS avec l'option de montage NFS `soft`.

Grâce à l'option `soft`, les erreurs d'E/S sont signalées au VDS ou VDC, et le VDC envoie des messages supplémentaires pour déterminer s'il s'agit d'une erreur d'E/S ou si le backend complet est indisponible. Le délai de basculement est proportionnel à la temporisation NFS et aux retransmissions éventuelles.

Ne modifiez pas la valeur par défaut de l'option de montage NFS `timeo` qui est de 60 secondes (`timeo=600`). Avec une valeur de temporisation inférieure, telle que `timeo=40`, vous pouvez rencontrer des erreurs d'E/S non justifiées. Par exemple, quand un serveur ou un réseau NFS est indisponible pendant cinq secondes puis revient à la normale, des erreurs d'E/S peuvent être signalées car l'interruption n'était pas assez longue pour provoquer un basculement. Une temporisation plus longue, par exemple de 60 secondes, masque les interruptions qui ne durent que quelques secondes.

Fonctionnalité multipathing de disque virtuel et délai d'attente de disque virtuel

Avec la fonctionnalité de chemins d'accès multiples (fonctionnalité multipathing) de disque virtuel, le chemin d'accès utilisé pour accéder au backend est automatiquement modifié si le backend devient inaccessible par le chemin d'accès actif. Cette modification du chemin se produit indépendamment de la valeur de la propriété `timeout` du disque virtuel.

La propriété `timeout` du disque virtuel définit la durée après laquelle une opération d'E/S échoue si aucun domaine d'E/S n'est disponible pour traiter l'E/S. Ce délai s'applique à tous les disques virtuels, même à ceux qui utilisent la fonctionnalité multipathing de disque virtuel.

Par conséquent, la définition d'un délai d'attente de disque virtuel lorsque la fonctionnalité multipathing de disque virtuel est configurée peut empêcher le fonctionnement correct de cette fonctionnalité, en particulier si la valeur de délai d'attente définie est faible. Il est donc préférable de ne pas définir de délai d'attente de disque virtuel pour les disques virtuels qui font partie d'un groupe multipathing.

Pour plus d'informations, reportez-vous à la section "Délai d'attente du disque virtuel" à la page 209.

▼ Procédure de configuration de la fonctionnalité multipathing de disque virtuel

Voir Figure 7, "Configuration de la fonctionnalité de chemins d'accès multiples d'un disque virtuel".

1. **Exportez le backend du disque virtuel à partir du domaine de service `primary`.**

   ```
   primary# ldm add-vdsdev mpgroup=mpgroup1 backend-path1 volume@primary-vds0
   ```

   où *backend-path1* est le chemin d'accès au backend du disque virtuel à partir du domaine `primary`.

2. **Exportez le même backend du disque virtuel à partir du domaine de service `alternate`.**

   ```
   primary# ldm add-vdsdev mpgroup=mpgroup1 backend-path2 volume@alternate-vds0
   ```

   où *backend-path2* est le chemin d'accès au backend du disque virtuel à partir du domaine `alternate`.

Remarque - *backend-path1* et *backend-path2* sont les chemins vers le même backend de disque virtuel, mais à partir de deux domaines différents (`primary` et `alternate`). Ces chemins peuvent être identiques ou différents en fonction de la configuration des domaines `primary` et `alternate`. Le nom *volume* est un choix de l'utilisateur. Il peut être identique ou différent pour les deux commandes.

---

3. **Exportez le disque virtuel sur le domaine invité.**

   ```
   primary# ldm add-vdisk disk-name volume@primary-vds0 domain-name
   ```

Remarque - Bien que le backend du disque virtuel soit exporté plusieurs fois via différents domaines de service, vous n'assignez qu'un seul disque virtuel au domaine invité et l'associez avec le backend du disque virtuel via l'un des domaines de service.

Exemple 34 Ajout d'un numéro d'unité logique (LUN) au service de disque virtuel à l'aide d'un Mpgroup sur les domaines principal et secondaire

L'exemple suivant indique comment créer un LUN et l'ajouter au service de disque virtuel des domaines principal et secondaire à l'aide du même mpgroup :

Afin de déterminer quel domaine à utiliser en premier lors de l'accès au LUN, indiquez le chemin d'accès lorsque vous ajoutez le disque au domaine.

- Créez les périphériques de disque virtuel :

  ```
  primary# ldm add-vdsdev mpgroup=ha lun1@primary-vds0
  primary# ldm add-vdsdev mpgroup=ha lun1@alternate-vds0
  ```

- Pour utiliser le LUN de `primary-vds0` en premier, exécutez les commandes suivantes :

  ```
  primary# ldm add-vdisk disk1 lun1@primary-vds0 gd0
  ```

- Pour utiliser le LUN de `alternate-vds0` en premier, exécutez les commandes suivantes :

```
primary# ldm add-vdisk disk1 lun1@alternate-vds0 gd0
```

Résultat de la fonctionnalité multipathing de disque virtuel

Après avoir configuré le disque virtuel avec la fonctionnalité multipathing et démarré le domaine invité, le disque virtuel accède à son backend via l'un des domaines de service auxquels il a été associé. Si ce domaine de service devient indisponible, le disque virtuel tente d'accéder à son backend via un autre domaine de service faisant partie du même groupe multipathing.

Attention - Lors de la définition d'un groupe multipathing (`mpgroup`), vérifiez que les backends de disque virtuel faisant partie du même `mpgroup` sont effectivement le même backend de disque virtuel. Si vous ajoutez des moteurs de traitement différents dans le même `mpgroup`, vous risquez de voir certains comportements inattendus et vous pouvez éventuellement perdre ou endommager les données stockées sur les moteurs de traitement.

Sélection du chemin dynamique

Vous pouvez sélectionner dynamiquement le chemin à utiliser pour un disque virtuel sur des domaines invités exécutant au moins le SE Oracle Solaris 11.2 SRU 1.

La sélection du chemin dynamique se produit lorsque le premier chemin d'un disque mpgroup est modifié à l'aide de la commande `ldm set-vdisk` permettant de définir la propriété `volume` sous une valeur prenant la forme *volume-name@service-name*. Un domaine actif prenant en charge la sélection de chemins dynamiques peut basculer vers le chemin sélectionné uniquement. Si les pilotes mis à jour ne sont pas en cours d'exécution, ce chemin est sélectionné quand le SE Oracle Solaris recharge l'instance de disque ou lors de la prochaine réinitialisation du domaine.

La sélection du chemin dynamique permet de suivre dynamiquement les étapes ci-après lorsque le disque est en cours d'utilisation :

- Indiquez le chemin de disque à essayer en premier par le domaine invité lorsque vous attachez le disque
- Modifiez le chemin d'accès actuel et choisissez celui qui est déjà indiqué pour les disques de chemins d'accès multiples déjà attachés

L'utilisation de la commande `ldm add-vdisk` avec un disque mpgroup définit désormais le chemin indiqué par *volume-name@service-name* comme chemin sélectionné permettant l'accès au disque.

Le chemin sélectionné apparaît en premier dans l'ensemble de chemins fourni pour le domaine invité, et ce quel que soit son rang lorsque le mpgroup a été créé.

Vous pouvez utiliser la commande `ldm set-vdisk` sur des domaines liés, inactifs et actifs. Pour l'utilisation sur des domaines actifs, cette commande permet de choisir uniquement le chemin d'accès sélectionné du disque mpgroup.

La commande `ldm list-bindings` affiche les informations suivantes :

- La colonne `STATE` de chaque chemin de mpgroup indique l'une des valeurs suivantes :
  - `active` – Chemin actif du mpgroup
  - `standby` – Chemin non utilisé actuellement
  - `unknown` – Le domaine ne prend pas en charge la sélection du chemin dynamique, le périphérique n'est pas attaché ou une erreur empêche le statut du chemin d'être retrouvé
- Les chemins de disque sont répertoriés dans l'ordre de choix du chemin actif
- Le volume associé au disque est le chemin sélectionné au mpgroup et figure en haut de la liste.

  L'exemple suivant montre que le chemin sélectionné est `vol-ldg2@opath-ldg2` et que le chemin actif en cours d'utilisation passe par le domaine `ldg1`. Cette situation est possible si le chemin sélectionné n'a pas pu être utilisé et que le deuxième chemin possible a été utilisé à la place. Même si le chemin sélectionné est en ligne, le chemin non sélectionné continue d'être utilisé. Pour faire en sorte que le premier chemin soit de nouveau actif, exécutez de nouveau la commande `ldm set-vdisk` de sorte à définir la propriété `volume` sur le nom du chemin de votre choix.

```
DISK

NAME            VOLUME                  TOUT ID DEVICE SERVER  MPGROUP
disk            disk-ldg4@primary-vds0       0  disk@0 primary
tdiskgroup      vol-ldg2@opath-ldg2          1  disk@1 ldg2    testdiskgroup
    PORT MPGROUP VOLUME        MPGROUP SERVER   STATE
    2    vol-ldg2@opath-ldg2   ldg2             standby
    0    vol-ldg1@opath-vds    ldg1             active
    1    vol-prim@primary-vds0 primary          standby
```

Si vous utilisez la commande `ldm set-vdisk` sur un disque mpgroup d'un domaine lié qui ne prend pas en charge au moins le SE Oracle Solaris 11.2 SRU 1, l'opération modifie l'ordre des priorités de chemin et le nouveau chemin peut être utilisé en premier lorsque le disque est de nouveau attaché ou réinitialisé, ou encore si l'OBP doit y accéder.

CD, DVD et images ISO

Vous pouvez exporter un CD ou un DVD de la même manière que vous exportez un disque normal. Pour exporter un CD ou un DVD vers un domaine invité, exportez la tranche 2 du périphérique CD ou DVD en tant que disque complet, c'est-à-dire sans l'option `slice`.

Remarque - Vous ne pouvez pas exporter l'unité de CD ou de DVD elle-même. Vous pouvez uniquement exporter le CD ou le DVD qui se trouve à l'intérieur de l'unité de CD ou de DVD. Par conséquent, un CD ou un DVD doit se trouver dans l'unité avant que vous ne puissiez l'exporter. Pour pouvoir exporter un CD ou un DVD, ceux-ci ne doivent pas être en cours d'utilisation dans le domaine de service. Plus précisément, le service du système de fichiers Volume Management `volfs` ne doit pas utiliser le CD ou le DVD. Reportez-vous à la section "Procédure d'exportation d'un CD ou d'un DVD à partir du domaine de service vers le domaine invité" à la page 207 pour savoir comment empêcher l'utilisation du périphérique par `volfs`.

Si vous disposez d'une image ISO (International Organisation for Standardization) d'un CD ou d'un DVD stockée dans un fichier ou sur un volume et que vous exportez ce fichier ou ce volume en tant que disque complet, il apparaît comme un CD ou un DVD dans le domaine invité.

Lorsque vous exportez un CD, un DVD ou une image ISO, il apparaît automatiquement comme un périphérique en lecture seule sur le domaine invité. Cependant, vous ne pouvez pas effectuer d'opérations de contrôle du CD à partir du domaine invité, c'est-à-dire que vous ne pouvez pas démarrer, arrêter ou éjecter le CD du domaine invité. Si l'image ISO, le CD ou le DVD exporté est amorçable, le domaine invité peut être initialisé sur le disque virtuel correspondant.

Par exemple, si vous exportez le DVD d'installation du SE Oracle Solaris, vous pouvez initialiser le domaine invité sur le disque virtuel qui correspond à ce DVD et installer le domaine invité à partir de ce DVD. Pour ce faire, lorsque le domaine invité atteint l'invite `ok`, utilisez la commande suivante.

```
ok boot /virtual-devices@100/channel-devices@200/disk@n:f
```

Où *n* est l'index du disque virtuel représentant le DVD exporté.

Remarque - Si vous exportez un DVD d'installation du SE Oracle Solaris et initialisez un domaine invité sur le disque virtuel qui correspond à ce DVD pour installer le domaine invité ; vous ne pouvez pas changer le DVD au cours de l'installation. Vous devrez peut-être sauter les étapes d'installation demandant un CD/DVD différent ou vous devrez fournir un autre chemin pour accéder au média demandé.

▼ Procédure d'exportation d'un CD ou d'un DVD à partir du domaine de service vers le domaine invité

1. **A partir du domaine de service, assurez-vous que le démon de gestion des volumes `vold` est en cours d'exécution et en ligne.**

```
service# svcs volfs
STATE          STIME    FMRI
online         12:28:12 svc:/system/filesystem/volfs:default
```

2. **Si le démon de gestion des volumes est en cours d'exécution et en ligne, comme dans l'exemple de l'étape 1, procédez comme suit :**

   a. **Editez le fichier `/etc/vold.conf` et placez en commentaire la ligne commençant par les mots suivants :**

   ```
   use cdrom drive....
   ```

   Reportez-vous à la page de manuel `vold.conf`(4).

   b. **Insérez le CD ou le DVD dans l'unité de CD ou de DVD.**

   c. **A partir du domaine de service, redémarrez le service du système de fichiers de gestion des volumes.**

   ```
   service# svcadm refresh volfs
   service# svcadm restart volfs
   ```

3. **A partir du domaine de service, recherchez le chemin de disque pour le périphérique CD-ROM.**

```
service# cdrw -l
Looking for CD devices...
   Node                  Connected Device                Device type
- - - - - - - - - - - - - - - - - - - - +- - - - - - - - - - - - - - - - - - - - - - - - - - - - -
+- - - - - - - - - - - - - -
/dev/rdsk/c1t0d0s2    | MATSHITA CD-RW  CW-8124   DZ13 | CD Reader/Writer
```

4. **Exportez le périphérique de CD ou de DVD comme un disque complet.**

```
primary# ldm add-vdsdev /dev/dsk/c1t0d0s2 cdrom@primary-vds0
```

5. **Assignez le CD ou le DVD exporté au domaine invité.**

La commande suivante indique comment assigner le CD ou le DVD exporté au domaine `ldg1` :

```
primary# ldm add-vdisk cdrom cdrom@primary-vds0 ldg1
```

Exportation multiple d'un CD ou d'un DVD

Un CD ou un DVD peut être exporté plusieurs fois et attribué à différents domaines invités. Reportez-vous à la section "Procédure d'exportation multiple du backend d'un disque virtuel" à la page 187 pour plus d'informations.

▼ Procédure d'exportation d'une image ISO à partir d'un domaine de contrôle pour installer un domaine hôte

Avant de commencer

Cette procédure suppose que le domaine `primary` et le domaine invité soient configurés.

Par exemple, `ldm list` montre que les deux domaines `primary` et `ldg1` sont configurés :

```
primary# ldm list
NAME             STATE    FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active   -n-cv   SP      8     8G       0.3%  15m
ldom1            active   -t--    5000    4     1G         25%  8m
```

1. **Ajoutez un périphérique de serveur de disque virtuel pour exporter l'image ISO.**

   Dans cet exemple, l'image ISO est `/export/images/sol-10-u8-ga-sparc-dvd.iso`.

   ```
   primary# ldm add-vdsdev /export/images/sol-10-u8-ga-sparc-dvd.iso dvd-iso@primary-vds0
   ```

2. **Arrêtez le domaine invité.**

   Dans cet exemple, le domaine logique est `ldom1`.

   ```
   primary# ldm stop-domain ldom1
   LDom ldom1 stopped
   ```

3. **Ajoutez le disque virtuel pour l'image ISO sur le domaine logique.**

   Dans cet exemple, le domaine logique est `ldom1`.

   ```
   primary# ldm add-vdisk s10-dvd dvd-iso@primary-vds0 ldom1
   ```

4. **Redémarrez le domaine invité.**

   Dans cet exemple, le domaine logique est `ldom1`.

   ```
   primary# ldm start-domain ldom1
   LDom ldom1 started
   # ldm list
   NAME             STATE    FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
   primary          active   -n-cv   SP      8     8G       0.4%  25m
   ldom1            active   -t--    5000    4     1G        0.0%  0s
   ```

   Dans cet exemple, la commande `ldm list` affiche que le domaine `ldom1` vient d'être démarré.

5. **Connectez-vous au domaine invité.**

```
primary# telnet localhost 5000
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.

Connecting to console "ldom1" in group "ldom1" ....
Press ~? for control options ..
```

6. **Vérifiez l'existence de l'image ISO en tant que disque virtuel.**

```
{0} ok show-disks
a) /virtual-devices@100/channel-devices@200/disk@1
b) /virtual-devices@100/channel-devices@200/disk@0
q) NO SELECTION
Enter Selection, q to quit: q
```

Dans cet exemple, le périphérique nouvellement ajouté est `/virtual-devices@100/channel-devices@200/disk@1`.

7. **Initialisez le domaine invité pour effectuer l'installation à partir de l'image ISO.**

Dans cet exemple, initialisez à partir de la tranche `f` du disque `/virtual-devices@100/channel-devices@200/disk@1`.

```
{0} ok boot /virtual-devices@100/channel-devices@200/disk@1:f
```

Délai d'attente du disque virtuel

Par défaut, si le domaine de service fournissant l'accès au backend d'un disque virtuel est en panne, toutes les E/S du domaine invité vers le disque virtuel correspondant sont bloquées. Les E/S reprennent automatiquement lorsque le domaine de service est opérationnel et traite les demandes d'E/S vers le backend du disque virtuel.

Cependant, il existe certains cas dans lesquels les systèmes de fichiers ou les applications peuvent ne pas vouloir que l'opération d'E/S se bloque, mais plutôt qu'elle échoue et signale une erreur lorsque le domaine de service est en panne pendant une trop longue période. Vos pouvez maintenant définir une période de délai d'attente de connexion pour chaque disque virtuel, qui peut ensuite être utilisée pour établir une connexion entre le client de disque virtuel sur un domaine invité et le serveur de disque virtuel sur le domaine de service. Lorsque cette période de délai d'attente est atteinte, toutes les E/S en attente ou toutes les nouvelles E/S échoueront tant que le domaine de service est en panne et que la connexion entre le client de disque virtuel et le serveur n'est pas rétablie.

Définissez ce délai d'attente à l'aide de l'une des méthodes suivantes :

- Utilisation de la commande `ldm add-vdisk`.

```
ldm add-vdisk timeout=seconds disk-name volume-name@service-name domain-name
```

- Utilisation de la commande `ldm set-vdisk`.

```
ldm set-vdisk timeout=seconds disk-name domain-name
```

- L'ajout de la ligne suivante au fichier `/etc/system` sur le domaine invité.

```
set vdc:vdc_timeout=seconds
```

Pour plus d'informations sur la création ou la mise à jour des valeurs de propriété `/etc/system`, reportez-vous à la section "Mise à jour des valeurs de propriété dans le fichier `/etc/system`" à la page 409.

Remarque - Si cette option est définie, elle remplace le paramètre de délai d'attente défini à l'aide de la CLI `ldm`. Cette option définit également le délai d'attente pour tous les disques virtuels dans le domaine invité.

Indiquez le délai d'attente en secondes. Si le délai d'attente est défini sur `0`, il est désactivé et les E/S sont bloquées pendant la panne du domaine de service (il s'agit du paramètre et du comportement par défaut).

Disque virtuel et SCSI

Si un disque SCSI physique ou un LUN est exporté en tant que disque complet, le disque virtuel correspondant prend en charge l'interface de commande SCSI utilisateur, `uscsi`, ainsi que les opérations de contrôle de disque multihôte `mhd`. Les autres disques virtuels, notamment les disques virtuels ayant un fichier ou un volume comme backend, ne prennent pas en charge ces interfaces.

Remarque - Vous ne pouvez pas utiliser des mpgroups et des réservations SCSI ensemble.

En conséquence, les applications ou fonctions de produit utilisant les commandes SCSI (notamment la commande `metaset` de Solaris Volume Manager ou la commande `shared devices` d'cluster Oracle Solaris) peuvent uniquement être utilisés dans des domaines invités, les disques virtuels disposant d'un disque SCSI physique comme backend.

Remarque - Les opérations SCSI sont effectivement exécutées par le domaine de service qui gère le disque SCSI physique ou le LUN utilisé comme moteur de traitement du disque virtuel. En particulier, les réservations SCSI sont implémentées par des commandes SCSI sur le domaine de service. Par conséquent, les applications s'exécutant dans le domaine de service et dans les domaines invités ne doivent pas émettre de commandes SCSI vers les mêmes disques SCSI physiques, car un état de disque inattendu risque de survenir.

Disque virtuel et commande `format`

La commande `format` reconnaît tous les disques virtuels contenus dans un domaine. Cependant, pour les disques virtuels exportés en tant que disques à tranche unique, la commande `format` ne peut pas modifier la table des partitions du disque virtuel. Les commandes telles que `label` échoueront à moins que vous ne tentiez d'écrire une étiquette de disque semblable à celle qui est déjà associée au disque virtuel.

Les disques virtuels dont les moteurs de traitement sont des disques SCSI prennent en charge toutes les sous-commandes `format`(1M). Les disques virtuels dont les moteurs de traitement ne sont pas des disques SCSI ne prennent pas en charge certaines des sous-commandes `format`(1M), notamment `repair` et `defect`. Dans ce cas, le comportement de `format`(1M) est semblable au comportement des disques Integrated Drive Electronics (IDE).

Utilisation de ZFS avec les disques virtuels

Cette section décrit l'utilisation du système de fichiers Zettabyte (ZFS) pour stocker des moteurs de traitement de disque virtuel exportés sur des domaines invités. ZFS fournit une solution pratique et puissante pour créer et gérer des moteurs de traitement de disque virtuel. ZFS permet d'effectuer les opérations suivantes :

- Stockage des images de disque dans des volumes ZFS ou des fichiers ZFS
- Utilisation d'instantanés pour sauvegarder des images de disque
- Utilisation de clones pour dupliquer les images de disque et mettre à disposition des domaines supplémentaires.

Pour plus d'informations sur l'utilisation de ZFS, reportez-vous au manuel *Oracle Solaris ZFS Administration Guide*.

Dans les descriptions et les exemples suivants, le domaine primary est également le domaine de service où les images de disque sont stockées.

Configuration d'un pool ZFS dans un domaine de service

Pour stocker des images de disque, créez d'abord un pool de stockage ZFS dans le domaine de service. Par exemple, cette commande crée le pool de stockage `ldmpool` contenant le disque `c1t50d0` dans le domaine `primary`.

```
primary# zpool create ldmpool c1t50d0
```

Stockage des images de disque avec ZFS

La commande suivante crée une image de disque pour le domaine invité `ldg1`. Un système de fichiers ZFS pour ce domaine invité est créé et toutes les images de disque de ce domaine invité seront stockées sur ce système de fichiers.

```
primary# zfs create ldmpool/ldg1
```

Les images de disque peuvent être stockées sur les volumes ZFS ou les fichiers ZFS. La création d'un volume ZFS, quelle que soit sa taille, est rapide à l'aide de la commande `zfs create -V`. D'autre part, les fichiers ZFS doivent être créés à l'aide de la commande `mkfile`. L'exécution de cette commande peut prendre un certain temps, surtout si le fichier à créer est très volumineux, ce qui est souvent le cas lors de la création d'images de disque.

Les volumes ZFS et les fichiers ZFS peuvent tirer partir des fonctions ZFS telles que les fonctions d'instantané et de clone, mais un volume ZFS est un pseudopériphérique tandis qu'un fichier ZFS est un fichier normal.

Si l'image de disque doit être utilisée comme disque virtuel sur lequel un SE est installé, celle-ci doit être de taille suffisante pour s'adapter aux contraintes d'installation du SE. La taille dépend de la version du SE et du type d'installation effectuée. Si vous installez le SE Oracle Solaris, vous pouvez utiliser une taille de disque de 20 Go pour accueillir n'importe quel type d'installation de n'importe quelle version du SE Oracle Solaris.

Exemples de stockage d'images de disque avec ZFS

Les exemples suivants indiquent comment stocker des images de disque à l'aide d'un volume ZFS ou d'un fichier ZFS. La syntaxe pour exporter un volume ou un fichier ZFS est identique, mais le chemin d'accès au backend est différent.

Lorsque le domaine invité est démarré, le volume ou le fichier ZFS apparaît comme un disque virtuel sur lequel le SE Oracle Solaris peut être installé.

EXEMPLE 35 Stockage d'une image de disque à l'aide d'un volume ZFS

Créez d'abord une image de 20 Go sur un volume.

```
primary# zfs create -V 20gb ldmpool/ldg1/disk0
```

Exportez ensuite le volume ZFS en tant que disque virtuel.

```
primary# ldm add-vdsdev /dev/zvol/dsk/ldmpool/ldg1/disk0 ldg1_disk0@primary-vds0
```

Affectez le volume ZFS au domaine invité `ldg1`.

```
primary# ldm add-vdisk disk0 ldg1_disk0@primary-vds0 ldg1
```

EXEMPLE 36 Stockage d'une image de disque à l'aide d'un fichier ZFS

Créez d'abord une image de disque de 20 Go sur un volume ZFS et créez le fichier ZFS.

```
primary# zfs create ldmpool/ldg1/disk0
primary# mkfile 20g /ldmpool/ldg1/disk0/file
```

Exportez ensuite le fichier ZFS en tant que disque virtuel.

```
primary# ldm add-vdsdev /ldmpool/ldg1/disk0/file ldg1_dis0@primary-vds0
```

Assignez le fichier ZFS au domaine invité `ldg1`.

```
primary# ldm add-vdisk disk0 ldg1_disk0@primary-vds0 ldg1
```

Création d'un instantané d'une image de disque

Lorsque votre image de disque est stockée sur un volume ZFS ou un fichier ZFS, vous pouvez créer des instantanés de cette image de disque à l'aide de la commande `snapshot` ZFS.

Avant de créer un instantané de l'image de disque, vérifiez que le disque n'est pas actuellement utilisé dans le domaine invité pour garantir que les données actuellement stockées sur l'image de disque sont cohérentes. Vous pouvez vous assurer qu'aucun disque n'est en cours d'utilisation dans un domaine invité de l'une des façons suivantes :

- Arrêter et dissocier le domaine invité. C'est la solution la plus sûre et la seule disponible si vous voulez créer un instantané d'une image de disque utilisée comme disque d'initialisation d'un domaine invité.

- Démontez toutes les tranches du disque dont vous souhaitez créer un instantané dans le domaine invité et vérifiez qu'aucune tranche n'est utilisée dans le domaine invité.

Dans cet exemple, en raison de la disposition ZFS, la commande pour créer un instantané de l'image de disque est identique que l'image de disque soit stockée sur un volume ZFS ou un fichier ZFS.

EXEMPLE 37 Création d'un instantané d'une image de disque

Cet exemple crée un instantané de l'image de disque qui a été créée pour le domaine `ldg1`.

```
primary# zfs snapshot ldmpool/ldg1/disk0@version_1
```

Utilisation du clone pour mettre à disposition un nouveau domaine

Un fois que vous avez créé un instantané d'une image de disque, vous pouvez dupliquer cette image de disque à l'aide de la commande `clone` ZFS. L'image clonée est ensuite assignée à un autre domaine. Le clonage d'une image de disque d'initialisation crée un disque d'initialisation pour un nouveau domaine invité sans avoir à effectuer tout le processus d'installation du SE Oracle Solaris.

Par exemple, si le `disk0` créé était le disque d'initialisation du domaine `ldg1`, procédez comme suit pour cloner ce disque pour créer un disque d'initialisation pour le domaine `ldg2`.

```
primary# zfs create ldmpool/ldg2
primary# zfs clone ldmpool/ldg1/disk0@version_1 ldmpool/ldg2/disk0
```

Ensuite, `ldompool/ldg2/disk0` peut être exporté comme disque virtuel et assigné au nouveau domaine `ldg2`. Le domaine `ldg2` peut s'initialiser directement à partir de ce disque virtuel sans devoir suivre toute le processus d'installation du SE.

Clonage d'une image de disque d'initialisation

Si une image de disque d'initialisation est clonée, la nouvelle image est totalement identique au disque d'initialisation d'origine, et elle contient toutes les informations ayant été stockées sur le disque d'initialisation avant que l'image ne soit clonée, notamment le nom d'hôte, l'adresse IP, la table du système de fichiers monté ou la configuration et les paramètres du système.

Comme la table du système de fichiers montée est identique sur l'image du disque d'initialisation d'origine et sur l'image de disque clonée, l'image de disque clonée doit être assignée au nouveau domaine dans le même ordre que celui du domaine d'origine. Par exemple, si l'image du disque d'initialisation a été assignée comme premier disque du domaine d'origine, l'image de disque clonée doit être assignée comme premier disque du nouveau domaine. Sinon, le nouveau domaine ne peut pas s'initialiser.

Si le domaine d'origine a été configuré avec une adresse IP statique, un nouveau domaine utilisant l'image clonée comment avec la même adresse IP statique. Dans ce cas, vous pouvez modifier la configuration réseau du nouveau domaine à l'aide de la commande `sysconfig unconfigure` d'Oracle Solaris 11 ou de la commande `sys-unconfig` d'Oracle Solaris 10. Pour éviter ce problème, vous pouvez également créer un instantané d'une image de disque de système non configuré.

Si le domaine d'origine était configuré avec le protocole DHCP (Dynamic Host Configuration Protocol), un nouveau domaine utilisant l'image clonée utilise également le protocole DHCP. Dans ce cas, il est inutile de modifier la configuration réseau du nouveau domaine, car il reçoit automatiquement une adresse IP et sa configuration réseau lorsqu'il s'initialise.

Remarque - L'ID hôte d'un domaine n'est pas stocké sur le disque d'initialisation, mais est assigné par Logical Domains Manager lorsque vous créez un domaine. Par conséquent, lorsque vous clonez une image de disque, le nouveau domaine ne conserve pas l'ID hôte du domaine d'origine.

▼ Procédure de création d'un instantané d'une image de disque d'un système non configuré

1. **Associez et démarrez le domaine d'origine.**
2. **Annulez la configuration du système.**
   - **Sous Oracle Solaris 11 : exécutez la commande `sysconfig unconfigure`.**
   - **Sous Oracle Solaris 10 : exécutez la commande `sys-unconfig`.**

   Lorsque cette opération est terminée, le domaine s'arrête.
3. **Arrêtez le domaine et annulez sa configuration. Ne le réinitialisez *pas*.**
4. **Prenez un instantané de l'image du disque d'initialisation du domaine.**

   Par exemple :

```
primary# zfs snapshot ldmpool/ldg1/disk0@unconfigured
```

A ce stade, vous avez l'instantané de l'image du disque d'initialisation d'un système non configuré.

5. **Clonez cette image pour créer un nouveau domaine qui, au premier démarrage, demande la configuration du système.**

Utilisation des gestionnaires de volumes dans un environnement Oracle VM Server for SPARC

Cette section décrit l'utilisation des gestionnaires de volumes dans un environnement Oracle VM Server for SPARC.

Utilisation de disques virtuels avec des gestionnaires de volumes

Tous les volumes ZFS, Solaris Volume Manager ou Veritas Volume Manager (VxVM) peuvent être exportés à partir d'un domaine de service vers un domaine invité en tant que disque virtuel. Un volume peut être exporté comme un disque à tranche unique (si l'option `slice` est définie avec la commande `ldm add-vdsdev`) ou en tant que disque complet.

Remarque - Le reste de cette section utilise un volume Solaris Volume Manager comme exemple. Cependant, la section s'applique également aux volumes ZFS et VxVM.

Les exemples suivants montrent comment exporter un volume en tant que disque à tranche unique.

Le disque virtuel du domaine invité (par exemple, `/dev/dsk/c0d2s0`) est directement mappé sur le volume associé (par exemple, `/dev/md/dsk/d0`), et les données stockées sur le disque virtuel du domaine invité sont directement stockées sur le volume associé sans métadonnées supplémentaires. Par conséquent, les données stockées sur le disque virtuel du domaine invité sont accessibles directement à partir du domaine de service, par le biais du volume associé.

Exemples

- Si le volume Solaris Volume Manager `d0` est exporté du domaine `primary` sur `domain1`, la configuration de `domain1` nécessite des étapes supplémentaires.

```
primary# metainit d0 3 1 c2t70d0s6 1 c2t80d0s6 1 c2t90d0s6
primary# ldm add-vdsdev options=slice /dev/md/dsk/d0 vol3@primary-vds0
```

```
primary# ldm add-vdisk vdisk3 vol3@primary-vds0 domain1
```

- Une fois `domain1` lié et démarré, le volume exporté apparaît comme `/dev/dsk/c0d2s0`, par exemple, et vous pouvez l'utiliser.

```
domain1# newfs /dev/rdsk/c0d2s0
domain1# mount /dev/dsk/c0d2s0 /mnt
domain1# echo test-domain1 > /mnt/file
```

- Une fois `domain1` arrêté et dissocié, les données stockées sur le disque virtuel de `domain1` sont accessibles directement à partir du domaine primary via le volume Solaris Volume Manager `d0`.

```
primary# mount /dev/md/dsk/d0 /mnt
primary# cat /mnt/file
test-domain1
```

Utilisation des disques virtuels avec Solaris Volume Manager

Si un volume RAID ou Solaris Volume Manager en miroir est utilisé comme disque virtuel par un autre domaine, il doit être exporté sans définir l'option d'exclusivité (`excl`). Sinon, s'il y a une panne de l'un des composants du volume Solaris Volume Manager, la récupération du volume Solaris Volume Manager à l'aide de la commande `metareplace` ou à l'aide d'un disque hot spare ne démarre pas. La commande `metastat` voit le volume comme étant en resynchronisation, mais la resynchronisation ne progresse pas.

Par exemple, `/dev/md/dsk/d0` est un volume RAID Solaris Volume Manager exporté en tant que disque virtuel avec l'option `excl` sur un autre domaine et `d0` est configuré avec des périphériques hot spare. Si un composant de `d0` échoue, Solaris Volume Manager remplace le composant défaillant par un disque hot spare et resynchronise le volume Solaris Volume Manager. Cependant, la resynchronisation ne démarre pas. Le volume est signalé comme étant en resynchronisation, mais la resynchronisation ne progresse pas.

```
primary# metastat d0
d0: RAID
    State: Resyncing
    Hot spare pool: hsp000
    Interlace: 32 blocks
    Size: 20097600 blocks (9.6 GB)
Original device:
    Size: 20100992 blocks (9.6 GB)
Device                                          Start Block  Dbase   State Reloc
c2t2d0s1                                               330  No       Okay  Yes
c4t12d0s1                                              330  No       Okay  Yes
/dev/dsk/c10t600C0FF0000000000015153295A4B100d0s1  330  No  Resyncing  Yes
```

Dans ce cas, le domaine utilisant le volume Solaris Volume Manager en tant que disque virtuel doit être arrêté et dissocié pour terminer la resynchronisation. Ensuite, le volume Solaris Volume Manager peut être resynchronisé à l'aide de la commande `metasync`.

```
# metasync d0
```

Utilisation des disques virtuels lorsque VxVM est installé

Lorsque VxVM est installé sur votre système et que Veritas Dynamic Multipathing (DMP) est activé sur un disque ou une partition physique que vous voulez exporter en tant que disque virtuel, vous devez exporter ce disque ou cette partition sans définir l'option `excl`, qui n'est pas définie par défaut. Sinon, vous recevez une erreur dans `/var/adm/messages` lors de l'association du domaine utilisant un tel disque.

```
vd_setup_vd():  ldi_open_by_name(/dev/dsk/c4t12d0s2) = errno 16
vds_add_vd():  Failed to add vdisk ID 0
```

Vous pouvez vérifier que Veritas DMP est activé en consultant les informations sur la fonctionnalité multipathing dans la sortie `vxdisk list`. Par exemple :

```
# vxdisk list Disk_3
Device:    Disk_3
devicetag: Disk_3
type:      auto
info:      format=none
flags:     online ready private autoconfig invalid
pubpaths:  block=/dev/vx/dmp/Disk_3s2 char=/dev/vx/rdmp/Disk_3s2
guid:      -
udid:      SEAGATE%5FST336753LSUN36G%5FDISKS%5F3032333948303144304E0000
site:      -
Multipathing information:
numpaths:  1
c4t12d0s2  state=enabled
```

Sinon, si Veritas DMP est activé sur un disque ou une tranche que vous voulez exporter en tant que disque virtuel avec l'option `excl` définie, vous pouvez désactiver le DMP à l'aide de la commande `vxdmpadm`. Par exemple :

```
# vxdmpadm -f disable path=/dev/dsk/c4t12d0s2
```

Utilisation de gestionnaires de volumes avec les disques virtuels

Cette section décrit l'utilisation de gestionnaires de volumes avec les disques virtuels.

Utilisation de ZFS avec les disques virtuels

Tous les disques virtuels peuvent être utilisés avec ZFS. Un pool de stockage ZFS (`zpool`) peut être importé dans un domaine qui voit tous les périphériques de stockage qui font partie de ce

`zpool`, que ces domaines voient tous ces périphériques comme des périphériques virtuels ou des périphériques réels.

Utilisation de Solaris Volume Manager avec les disques virtuels

Tout disque virtuel peut être utilisé dans l'ensemble de disques locaux Solaris Volume Manager. Par exemple, un disque virtuel peut être utilisé pour stocker la base de données d'état des métapériphériques Solaris Volume Manager, `metadb`, de l'ensemble des disques locaux ou pour créer des volumes Solaris Volume Manager dans l'ensemble de disques locaux.

Tout disque virtuel dont le backend est un disque SCSI peut être utilisé dans un ensemble de disques partagés Solaris Volume Manager, `metaset`. Les disques virtuels dont les backends ne sont pas des disques SCSI ne peuvent pas être ajoutés dans un ensemble de disques partagés Solaris Volume Manager. Toute tentative pour ajouter un disque virtuel dont le backend n'est pas un disque SCSI dans un ensemble de disques partagés Solaris Volume Manager échouera avec une erreur semblable à la suivante.

```
# metaset -s test -a c2d2
metaset: domain1: test: failed to reserve any drives
```

Utilisation de VxVM avec les disques virtuels

Pour la prise en charge de VxVM dans les domaines invités, reportez-vous à la documentation VxVM de Symantec.

Problèmes liés au disque virtuel

La section suivante décrit les problèmes possibles lors de l'utilisation de disques virtuels.

Dans certaines circonstances, la configuration ou les métapériphériques Solaris Volume Manager d'un domaine invité peuvent être perdus

Si un domaine de service exécute une version du SE Oracle Solaris 10 antérieure à SE Oracle Solaris 10 1/13 et qu'il exporte une tranche de disque physique sous forme de disque virtuel

vers un domaine invité, ce disque virtuel est présenté sur le domaine invité avec un ID de périphérique incorrect. Si ce domaine de service est ensuite mis à niveau vers SE Oracle Solaris 10 1/13, la tranche de disque physique exportée sous forme de disque virtuel est présentée sur le domaine invité sans ID de périphérique.

L'absence d'ID de périphérique pour le disque virtuel peut être à l'origine de problèmes au niveau des applications qui tentent de référencer l'ID de périphérique de disques virtuels. En particulier, Solaris Volume Manager risque de ne plus pouvoir détecter sa configuration ou d'accéder à ses métapériphériques.

Après avoir mis à niveau un domaine de service vers SE Oracle Solaris 10 1/13, si un domaine invité ne parvient pas à détecter sa configuration ou ses métapériphériques Solaris Volume Manager, effectuez la procédure suivante.

▼ Détection de la configuration ou des métapériphériques Solaris Volume Manager d'un domaine invité

1. **Initialisez le domaine invité.**

2. **Désactivez la fonction `devid` de Solaris Volume Manager en ajoutant les lignes suivantes au fichier `/kernel/drv/md.conf` :**

   ```
   md_devid_destroy=1;
   md_keep_repl_state=1;
   ```

3. **Réinitialisez le domaine invité.**

   Après l'initialisation du domaine, la configuration et les métapériphériques de Solaris Volume Manager devraient être disponibles.

4. **Examinez la configuration de Solaris Volume Manager pour vous assurer qu'elle est correcte.**

5. **Réactivez la fonction `devid` de Solaris Volume Manager en supprimant du fichier `/kernel/drv/md.conf` les deux lignes que vous avez ajoutées à l'étape 2.**

6. **Réinitialisez le domaine invité.**

   Lors de la réinitialisation, des messages semblables aux suivants s'affichent

   ```
   NOTICE: mddb: unable to get devid for 'vdc', 0x10
   ```

   Il n'y a rien d'anormal à cela dans la mesure où aucun problème n'est signalé.

Compatibilité du disque d'initialisation Oracle Solaris

Par le passé, le SE Oracle Solaris était installé sur un disque d'initialisation configuré avec une étiquette de disque VTOC SMI. A partir du SE Oracle Solaris 11.1, le SE est installé sur un disque d'initialisation configuré par défaut avec une étiquette GPT (GUID partition table, table de partition GUID) de type EFI (Extensible Firmware Interface). Sauf pour les serveurs UltraSPARC T2, UltraSPARC T2 Plus et SPARC T3, les versions actuelles de microprogramme système de tous les serveurs supportés prennent en charge les étiquettes EFI.

Les serveurs suivants ne peuvent pas s'initialiser à partir d'un disque possédant une étiquette de disque GPT EFI :

- Serveurs UltraSPARC T2, UltraSPARC T2 Plus et SPARC T3, quelle que soit la version du microprogramme système utilisée
- Serveurs SPARC T4 exécutant des versions du microprogramme système antérieures à la version 8.4.0
- Serveurs SPARC T5, SPARC M5 et SPARC M6 exécutant des versions du microprogramme système antérieures à la version 9.1.0
- Serveurs de série SPARC T7 et SPARC M7 exécutant des versions du microprogramme système antérieures à la version 9.4.3

Ainsi, un disque d'initialisation Oracle Solaris 11.1 créé sur un serveur SPARC T4, SPARC T5, un serveur de série SPARC T7, un serveur SPARC M5, SPARC M6 ou un serveur de série SPARC M7 ne peut pas être utilisé sur un serveur plus ancien ou sur un serveur exécutant un microprogramme plus ancien.

Cette restriction réduit les possibilités d'effectuer des migrations à froid ou des migrations directes pour déplacer un domaine d'un serveur récent vers un serveur plus ancien. Cette restriction vous empêche également d'utiliser une image de disque d'initialisation GPT EFI sur un ancien serveur.

Pour déterminer si un disque d'initialisation Oracle Solaris 11.1 est compatible avec votre serveur et le microprogramme dont celui-ci est équipé, assurez-vous que le SE Oracle Solaris 11.1 est installé sur un disque configuré avec une étiquette de disque VTOC SMI.

Pour assurer la rétrocompatibilité avec les systèmes exécutant des microprogrammes plus anciens, effectuez l'une des procédures ci-après. Sinon, le disque d'initialisation utilise par défaut l'étiquette de disque GPT EFI. Ces procédures indiquent comment s'assurer que le SE Oracle Solaris 11.1 est installé sur un disque d'initialisation avec étiquette de disque VTOC SMI sur un serveur SPARC T4 équipé au minimum de la version 8.4.0 du microprogramme système, sur un serveur SPARC T5, SPARC M5 ou SPARC M6 équipé au minimum de la version 9.1.0 du microprogramme système et sur un serveur de série SPARC T7 ou SPARC M7 équipé au minimum de la version 9.4.3 du microprogramme système.

- **Solution 1 :** retirez la propriété `gpt` de manière à ce que le microprogramme ne signale pas qu'il prend en charge EFI.

  1. A partir de l'invite d'OpenBoot PROM, désactivez l'initialisation automatique et réinitialisez le système à installer.

     ```
     ok setenv auto-boot? false
     ok reset-all
     ```

     Après la réinitialisation du système, il revient à l'invite `ok`.
  2. Accédez au répertoire `/packages/disk-label` et supprimez la propriété `gpt`.

     ```
     ok cd /packages/disk-label
     ok " gpt" delete-property
     ```
  3. Débutez l'installation du SE Oracle Solaris 11.1.

     Effectuez par exemple une initialisation réseau :

     ```
     ok boot net - install
     ```

- **Solution 2 :** servez-vous de la commande `format -e` pour écrire une étiquette VTOC SMI sur le disque sur lequel installer le SE Oracle Solaris 11.1.

  1. Ecrivez une étiquette VTOC SMI sur le disque.

     Sélectionnez par exemple l'option `label` et spécifiez l'étiquette SMI :

     ```
     # format -e c1d0
     format> label
     [0] SMI Label
     [1] EFI Label
     Specify Label type[1]: 0
     ```
  2. Configurez le disque avec une tranche 0 et une tranche 2 couvrant la totalité du disque.

     Le disque ne doit pas comporter d'autres partitions. Par exemple :

     ```
     format> partition

     partition> print
     Current partition table (unnamed):
     Total disk cylinders available: 14087 + 2 (reserved cylinders)

     Part      Tag    Flag     Cylinders         Size            Blocks
       0       root    wm       0 - 14086      136.71GB    (14087/0/0) 286698624
       1 unassigned    wu       0                0         (0/0/0)             0
       2     backup    wu       0 - 14086      136.71GB    (14087/0/0) 286698624
       3 unassigned    wm       0                0         (0/0/0)             0
       4 unassigned    wm       0                0         (0/0/0)             0
     ```

```
  5 unassigned    wm       0               0         (0/0/0)           0
  6 unassigned    wm       0               0         (0/0/0)           0
  7 unassigned    wm       0               0         (0/0/0)           0
```

3. Réécrivez l'étiquette de disque VTOC SMI.

```
partition> label
[0] SMI Label
[1] EFI Label
Specify Label type[0]: 0
Ready to label disk, continue? y
```

4. Configurez le programme d'installation automatisée (AI) Oracle Solaris de manière à ce qu'il installe le SE Oracle Solaris sur la tranche 0 du disque d'initialisation.

   Modifiez comme indiqué ci-dessous la section `<disk>` du manifeste AI :

```
<target>
   <disk whole_disk="true">
        <disk_keyword key="boot_disk"/>
        <slice name="0" in_zpool="rpool"/>
   </disk>
[...]
</target>
```

5. Procédez à l'installation du SE Oracle Solaris 11.1.

♦ ♦ ♦ CHAPITRE 12

Utilisation des adaptateurs de bus hôte SCSI virtuels

Ce chapitre décrit comment utiliser les adaptateurs de bus hôte (HBA) SCSI virtuels avec le logiciel Oracle VM Server for SPARC.

Ce chapitre aborde les sujets suivants :

Présentation des adaptateurs de bus hôte SCSI virtuels

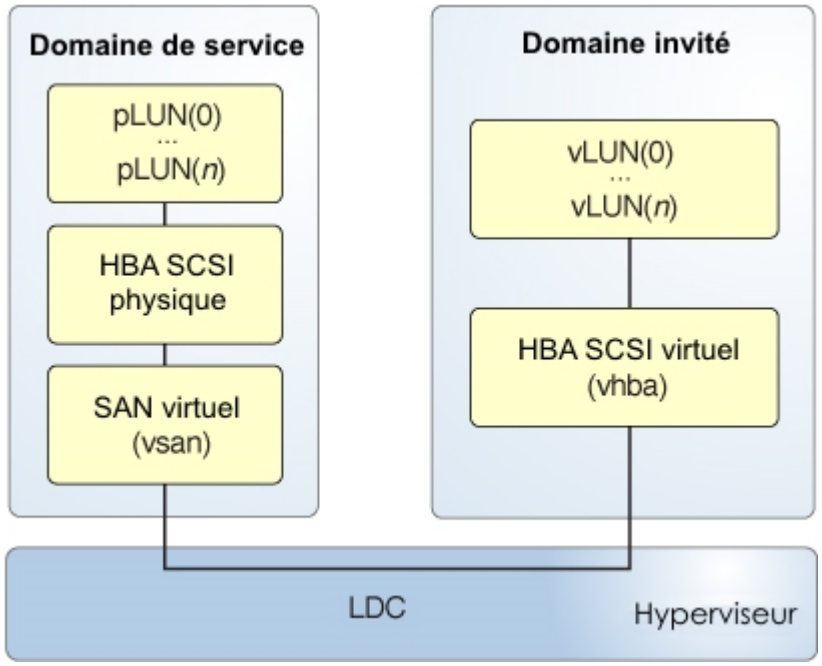
Un adaptateur de bus hôte (HBA) SCSI virtuel comprend deux composants : un HBA virtuel dans le domaine invité et un réseau de stockage (SAN) virtuel dans le domaine de service. Le HBA virtuel et les instances du SAN virtuel coopèrent pour implémenter une interface HBA SCSI pour les pilotes cible qui s'exécutent dans le domaine invité. Le service vSAN est implémenté par le pilote `vsan`, qui transmet les demandes d'E/S SCSI au pilote HBA SCSI physique s'exécutant dans le domaine de service. Le pilote `vhba` envoie des demandes d'E/S `vsan` en utilisant le canal de domaine logique géré ou de l'hyperviseur (LDC).

Une instance vHBA fournit l'accès à tous les périphériques SCSI qu'une instance vSAN peut atteindre. Un vHBA peut reconnaître tout type de périphérique SCSI tel que disque, CD, DVD ou bande. L'ensemble de périphériques SCSI accessibles change en fonction de l'ensemble de périphériques SCSI physiques qui est actuellement connu du pilote HBA physique associé au SAN virtuel. L'identité et le nombre des périphériques SCSI connus d'un vHBA ne sont pas connus avant l'exécution, comme c'est le cas avec un pilote HBA physique.

Le vHBA a des LUN virtuels (vLUN) comme périphériques enfant et ceux-ci se comportent de la même façon que les LUN physiques. Par exemple, vous pouvez utiliser la fonctionnalité multipathing d'E/S d'Oracle Solaris avec une instance vHBA et ses vLUN. Le chemin d'accès à un périphérique pour un vLUN utilise la notation complète `cXtYdZsN` : `/dev/[r]dsk/cXtYdZsN`. La partie `tY` du nom du périphérique indique le périphérique SCSI cible.

Après avoir configuré le SAN virtuel et le HBA SCSI virtuel, vous pouvez effectuer des opérations telles que l'initialisation d'un LUN virtuel à partir de l'invite OpenBoot ou l'affichage de tous les LUN virtuels à l'aide de la commande `format`.

FIGURE 8 HBA SCSI virtuels avec Oracle VM Server for SPARC



Un SAN virtuel existe dans un domaine de service et est implémenté par le module de noyau `vsan`, alors qu'un HBA SCSI virtuel existe dans un domaine invité et est implémenté par le module `vhba`. Un SAN virtuel est associé à un port initiateur de HBA SCSI physique spécifique, alors qu'un HBA SCSI virtuel est associé à un SAN virtuel spécifique.

Le module `vhba` exporte une interface compatible SCSA pour recevoir les demandes d'E/S de n'importe quel pilote SCSI cible compatible SCSA. Le module `vhba` traduit les demandes d'E/S en messages de protocole d'E/S virtuels qui sont envoyés via la LDC au domaine de service.

Le module `vsan` traduit les messages d'E/S virtuels envoyés par `vhba` en demandes d'E/S. Ces demandes sont envoyées à un pilote HBA SCSI physique compatible SCSA. Le module `vsan` renvoie les données traitées d'E/S et le statut `vhba` via la LDC. Enfin, le `vhba` transfère cette réponse d'E/S à l'initiateur de la demande d'E/S.

Modèle opérationnel pour les HBA SCSI virtuels

Le modèle opérationnel pour l'utilisation des HBA SCSI virtuels est différent de celui des autres types de périphériques virtuels Oracle VM Server for SPARC car seuls le HBA SCSI virtuel et les instances du SAN virtuel sont connus du Logical Domains Manager. Les LUN virtuels qui s'affichent dans le domaine invité et les LUN physiques qui figurent dans le domaine de service ne sont pas connus avant d'être détectés à l'exécution. Les LUN virtuels et les LUN physiques sont détectés, de manière implicite quand la connexion LDC associée est réinitialisée et, de manière explicite, à l'aide de la commande `ldm rescan-vhba`.

Bien qu'il soit possible d'exécuter la commande `ldm` pour nommer de façon explicite un disque virtuel, un LUN virtuel dans un domaine invité dérive son identité de celle du LUN physique qui lui est associé dans le domaine de service. Reportez-vous à la page de manuel `ldm`(1M).

Par exemple, le LUN physique et le LUN virtuel partagent le texte affiché en gras dans les chemins de périphérique suivants :

- LUN physique dans le domaine de service :

  ```
  /pci@0/pci@0/pci@8/pci@0/pci@2/SUNW,qlc@0/fp@0,0/ssd@w216000c0ff8089d5,0
  ```

- LUN virtuel dans le domaine invité :

  ```
  /virtual-devices@100/channel-devices@200/scsi@1/iport@0/disk@w216000c0ff8089d5,0
  ```

Remarque - Le chemin de périphérique du domaine invité n'est présent que si la fonctionnalité multipathing d'E/S d'Oracle Solaris est désactivée dans le domaine invité. Si la fonctionnalité multipathing d'E/S d'Oracle Solaris est activée, le module `scsi_vhci` crée le chemin de périphérique dans le domaine invité et a une syntaxe différente.

Notez que le composant `scsi@1` dans le chemin de périphérique du LUN virtuel indique l'instance de HBA SCSI virtuel dont ce LUN virtuel est membre.

Etant donné que l'ensemble de LUN virtuels d'un HBA SCSI virtuel est dérivé du domaine de service au moment de l'exécution, il n'est pas possible d'ajouter des LUN virtuels au domaine invité ou de les en retirer, de manière explicite. A la place, vous devez ajouter ou enlever les LUN physiques sous-jacents afin de pouvoir modifier l'appartenance des LUN virtuels au domaine invité. Un événement tel que la réinitialisation ou la migration d'un domaine peut provoquer la modification de l'appartenance des LUN virtuels à des domaines invités. Cette modification intervient car les LUN virtuels sont détectés automatiquement chaque fois que la connexion LDC du HBA SCSI virtuel est réinitialisée. Si le LUN physique sous-jacent d'un LUN virtuel est introuvable lors d'une détection future, le LUN virtuel est marqué comme étant indisponible et l'accès au LUN virtuel renvoie un message d'erreur similaire à celui-ci :

```
WARNING: .../scsi@1/iport@0/disk@w216000c0ff8089d5,0 (sd6): ... Command failed to
 complete...Device is gone
```

Une instance de HBA SCSI virtuel est gérée par le pilote `vhba`, mais un LUN virtuel est géré par le pilote SCSI cible en fonction du type de périphérique du LUN physique sous-jacent. La sortie suivante confirme que le pilote `vhba` gère l'instance du HBA SCSI virtuel et que le pilote du disque SCSI `sd` gère le LUN virtuel :

```
# prtconf -a -D /dev/dsk/c2t216000C0FF8089D5d0
SUNW,SPARC-Enterprise-T5220 (driver name: rootnex)
    virtual-devices, instance #0 (driver name: vnex)
        channel-devices, instance #0 (driver name: cnex)
            scsi, instance #0 (driver name: vhba)
                iport, instance #3 (driver name: vhba)
                    disk, instance #30 (driver name: sd)
```

Identificateur de HBA SCSI virtuel et nom de périphérique

Lorsque vous utilisez la commande `ldm add-vhba` pour ajouter un HBA SCSI virtuel à un domaine, vous pouvez spécifier son numéro de périphérique en définissant la propriété `id`.

`ldm add-vhba [id=`*vHBA-ID*`]` *vHBA-name vSAN-name domain-name*

Chaque HBA SCSI virtuel d'un domaine a un numéro de périphérique unique qui est assigné lorsque le domaine est lié. Si un HBA SCSI virtuel a été ajouté avec un numéro de périphérique explicite (au moyen de la définition de la propriété `id` sur une valeur décimale), le numéro de périphérique spécifié est utilisé. Sinon, le système assigne automatiquement le numéro de périphérique le plus petit disponible. Dans ce cas, le numéro de périphérique assigné dépend de la manière dont les HBA SCSI virtuels ont été ajoutés au domaine. Lorsqu'un domaine est lié, le numéro de périphérique qui finit par être assigné à un HBA SCSI virtuel est visible dans la sortie des commandes `ldm list-bindings` et `ldm list -o hba`.

Les commandes `ldm list-bindings`, `ldm list -o hba` et `ldm add-vhba id=`*id* illustrent et spécifient toutes la définition de la propriété `id` sur une valeur décimale. Le SE Oracle Solaris illustre la définition de la propriété `id` du HBA SCSI virtuel sur une valeur hexadécimale.

L'exemple suivant montre que `vhba@0` est le nom du périphérique pour le HBA SCSI virtuel `vh1` sur le domaine `gdom`.

```
primary# ldm list -o hba gdom
NAME
gdom

VHBA
    NAME             VSAN                    DEVICE  TOUT SERVER
    vh1              vs1                     vhba@0  0    svcdom
```

Attention - Si un numéro de périphérique n'est pas assigné explicitement à un HBA SCSI virtuel, son numéro de périphérique peut changer lorsque le domaine est dissocié, puis est réassocié ultérieurement. Dans ce cas, le nom de périphérique assigné par le SE s'exécutant dans le domaine pourrait également changer et rompre la configuration existante du système. Cela peut se produire, par exemple, lorsqu'un HBA SCSI virtuel est supprimé de la configuration du domaine.

Gestion des HBA SCSI virtuels

Cette section couvre les tâches suivantes :

Pour plus d'informations sur les commandes présentées dans cette section, reportez-vous à la page de manuel `ldm`(1M).

Obtention des informations sur un HBA SCSI physique

Avant de configurer un HBA SCSI virtuel, vous devez obtenir les informations concernant les HBA SCSI physiques liés au domaine de service. Pour plus d'informations sur la configuration des cartes HBA dans les domaines d'E/S, voir Chapitre 6, Configuration de domaines d'E/S.

Remarque - Si au moins le SE Oracle Solaris 11.3 est installé dans le domaine `primary`, le domaine de service peut être le domaine de contrôle.

La commande `ldm list-hba` répertorie les ports initiateurs de HBA SCSI physique pour le domaine actif spécifié. Après avoir identifié les ports initiateurs de HBA SCSI d'un domaine logique, vous pouvez spécifier un port initiateur donné sur la ligne de commande `ldm add-vsan` pour créer un SAN virtuel.

```
ldm list-hba [-d] [-l] [-p] [-t] domain-name
```

L'exemple suivant illustre les ports initiateurs pour les HBA SCSI qui sont liés au domaine de service `svcdom`. L'option `-l` affiche des informations détaillées.

```
primary# ldm list-hba -l svcdom
NAME                                                    VSAN
----                                                    ----
/SYS/MB/SASHBA0/HBA0/PORT1
[/pci@300/pci@1/pci@0/pci@2/scsi@0/iport@1]
/SYS/MB/SASHBA0/HBA0/PORT2
[/pci@300/pci@1/pci@0/pci@2/scsi@0/iport@2]
/SYS/MB/SASHBA0/HBA0/PORT4
[/pci@300/pci@1/pci@0/pci@2/scsi@0/iport@4]
/SYS/MB/SASHBA0/HBA0/PORT8
[/pci@300/pci@1/pci@0/pci@2/scsi@0/iport@8]
/SYS/MB/PCIE1/HBA0/PORT0,0
[/pci@300/pci@1/pci@0/pci@4/SUNW,emlxs@0/fp@0,0]
/SYS/MB/PCIE1/HBA0,1/PORT0,0
[/pci@300/pci@1/pci@0/pci@4/SUNW,emlxs@0,1/fp@0,0]
```

Si les LUN que vous vous attendez à voir pour un port initiateur ne s'affichent pas dans la sortie `ldm list-hba`, vérifiez que la fonctionnalité de chemins d'accès multiples est désactivée dans le domaine de service référencé pour le port initiateur référencé. Reportez-vous à *Managing SAN Devices and Multipathing in Oracle Solaris 11.3*.

Création d'un réseau de stockage virtuel

Après avoir obtenu le port initiateur du HBA SCSI physique, vous devez créer le réseau de stockage (SAN) virtuel sur le domaine de service. Le SAN virtuel gère tous les périphériques SCSI accessibles à partir du port initiateur du HBA SCSI spécifié.

```
ldm add-vsan [-q] iport-path vSAN-name domain-name
```

Le nom de SAN virtuel est unique pour le système, pas pour le nom de domaine indiqué. Le nom de domaine identifie le domaine dans lequel le port initiateur de HBA SCSI est configuré. Vous pouvez créer plusieurs SAN virtuels qui référencent le même chemin de port initiateur.

Vous pouvez créer plus d'un SAN virtuel à partir du même chemin de port initiateur. Plusieurs domaines invités peuvent ainsi utiliser le même port initiateur.

Remarque - Quand le SE Oracle Solaris 11.3 s'exécute sur le domaine de service, la commande `ldm add-vsan` vérifie que le chemin du port initiateur est un chemin de périphérique valide. Si le domaine de service spécifié n'est pas actif quand vous exécutez la commande `ldm add-vsan`, le chemin de port initiateur spécifié ne peut pas être vérifié par le domaine de service. Si le chemin de port initiateur ne correspond pas au port initiateur du HBA SCSI physique installé qui fait partie du domaine de service, un message d'avertissement est consigné dans le journal système du domaine de service quand celui-ci devient actif.

Dans cet exemple, vous associez le port initiateur `/SYS/MB/PCIE1/HBA0,1/PORT0,0` sur le domaine de service `svcdom` à un SAN virtuel. Vous pouvez choisir le nom du SAN virtuel. Dans cet exemple, `port0` est le nom du SAN virtuel.

```
primary# ldm add-vsan /SYS/MB/PCIE1/HBA0,1/PORT0,0 port0 svcdom
/SYS/MB/PCIE1/HBA0,1/PORT0,0 resolved to device:
/pci@300/pci@1/pci@0/pci@4/SUNW,emlxs@0,1/fp@0,0
```

Création d'un adaptateur de bus hôte SCSI virtuel

Une fois le SAN virtuel défini, vous pouvez utiliser la commande `ldm add-vhba` pour créer un HBA SCSI virtuel dans un domaine invité. Le HBA SCSI virtuel envoie les demandes d'E/S aux périphériques SCSI physiques dans le SAN virtuel.

```
ldm add-vhba [id=vHBA-ID] vHBA-name vSAN-name domain-name
```

Dans cet exemple, vous créez le HBA SCSI virtuel `port0_vhba` sur le domaine invité `gdom` qui communique avec le SAN virtuel `port0`.

```
primary# ldm add-vhba port0_vhba port0 gdom
```

Vérification de la présence d'un HBA SCSI virtuel

Utilisez la commande `ldm list` pour vérifier la présence de HBA SCSI virtuel nouvellement créé et les périphériques SAN virtuels sur le domaine de service et le domaine invité.

```
ldm list -o san,hba [domain-name ...]
```

Dans cet exemple, le domaine de service qui comporte le SAN virtuel est `svcdom` et le domaine invité qui comporte le HBA SCSI virtuel est `gdom`. Notez que l'identificateur de HBA SCSI virtuel n'est pas alloué dans cet exemple car le domaine `gdom` n'est pas encore lié.

```
primary# ldm list -o san,hba svcdom gdom
NAME
svcdom

VSAN
    NAME             TYPE   DEVICE IPORT
    port0            VSAN [/pci@300/pci@1/pci@0/pci@4/SUNW,emlxs@0,1/fp@0,0]

------------------------------------------------------------------------------
NAME
gdom

VHBA
    NAME             VSAN                        DEVICE TOUT SERVER
    port0_vhba       port0                              0    svcdom
```

Définition du délai d'attente du HBA SCSI virtuel

La commande `ldm set-vhba` permet de spécifier une valeur de délai d'attente pour le HBA SCSI virtuel sur le domaine logique spécifié. La propriété `timeout` spécifie le délai, en secondes, pendant lequel l'instance du HBA SCSI virtuel spécifié doit attendre avant de déclarer qu'une connexion LDC ne peut pas être établie avec le SAN virtuel. Reportez-vous à la section "Délai d'attente d'un HBA SCSI virtuel" à la page 244.

Une valeur de délai d'attente égale à zéro provoque une attente indéfinie du `vhba` pour la création de la connexion LDC avec le SAN virtuel.

`ldm set-vhba [timeout=`*seconds*`]` *vHBA-name domain-name*

Dans cet exemple, vous définissez un délai d'attente de 90 secondes pour le HBA SCSI virtuel `port0_vhba` sur le domaine invité `gdom`.

```
primary# ldm set-vhba timeout=90 port0_vhba gdom
```

Retrait d'un adaptateur de bus hôte SCSI virtuel

Vous pouvez exécuter la commande `ldm remove-vhba` pour retirer un HBA SCSI virtuel d'un domaine invité spécifié.

Assurez-vous que ni le SE ni aucune application actifs n'utilisent le HBA SCSI virtuel avant de tenter de le retirer. Si le HBA SCSI virtuel est en cours d'utilisation, la commande `ldm remove-vhba` échoue.

`ldm remove-vhba` *vHBA-name domain-name*

Dans cet exemple, vous retirez le HBA SCSI virtuel `port0_vhba` du domaine invité `gdom`.

```
primary# ldm remove-vhba port0_vhba gdom
```

Retrait d'un réseau de stockage virtuel

Vous pouvez exécuter la commande `ldm remove-vsan` pour enlever un SAN virtuel.

Tout d'abord, retirez le HBA SCSI virtuel qui est associé au SAN virtuel. Exécutez ensuite la commande `ldm remove-vsan` pour enlever le SAN virtuel.

`ldm remove-vsan` *vSAN-name*

Dans cet exemple, vous enlevez le SAN virtuel `port0` :

```
primary# ldm remove-vsan port0
```

Ajout ou retrait d'un LUN

Vous ne pouvez pas ajouter ou retirer un LUN virtuel directement à partir d'un HBA SCSI virtuel. Vous devez d'abord ajouter ou retirer un LUN physique avant d'exécuter la commande `ldm rescan-vhba` pour synchroniser l'ensemble des périphériques SCSI visibles par le HBA SCSI virtuel et le SAN virtuel. Les commandes permettant d'ajouter ou de retirer un LUN physique sont spécifiques de la topologie du port initiateur associé au SAN virtuel. Par exemple, si le port initiateur communique avec un SAN physique, vous devez exécuter les commandes d'administration du SAN pour ajouter un LUN ou le retirer d'un élément du SAN.

`ldm rescan-vhba` *vHBA-name domain-name*

Par exemple, la commande suivante synchronise les périphériques SCSI pour le HBA SCSI virtuel `port0_vhba` sur le domaine `gdom` :

```
primary# ldm rescan-vhba port0_vhba gdom
```

Affichage des LUN virtuels dans un domaine invité

Le ou les LUN virtuels qui sont associés à un HBA SCSI virtuel se comportent comme s'ils étaient des LUN physiques.

Un LUN virtuel représentant un disque SCSI, par exemple, s'affiche dans le domaine invité comme un disque normal sous `/dev/[r]dsk`. Le LUN virtuel est visible dans la sortie de la commande `format` car le LUN physique associé sous-jacent est du type `disk`. Le chemin de

périphérique du LUN virtuel peut être géré par d'autres commandes telles que `prtpicl` et `prtconf`.

Si le LUN virtuel représente un CD ou un DVD, son chemin de périphérique est défini dans `/dev/[r]dsk`. Si le LUN virtuel représente un périphérique de bande, son chemin de périphérique est défini dans `/dev/rmt`. Pour les commandes fonctionnant sur le LUN virtuel, reportez-vous à *Managing Devices in Oracle Solaris 11.3*.

Configurations de HBA SCSI virtuel et de SAN virtuel

La configuration des HBA SCSI virtuels et des SAN virtuels est très flexible. Le port initiateur du HBA SCSI physique qui est utilisé par la commande `ldm add-vsan` peut piloter tout type de bus qui prend en charge le SCSI, tel que Fibre Channel, SAS ou SATA. Un HBA SCSI virtuel et un SAN virtuel peuvent s'exécuter dans le même domaine. Dans la configuration la plus courante, le HBA SCSI virtuel et le SAN virtuel s'exécutent dans des domaines différents, le SAN virtuel s'exécutant dans un domaine de service qui accède directement à une carte HBA physique.

Bien qu'un SAN virtuel soit conceptuellement associé à un SAN physique, il n'a pas besoin de l'être. Par exemple, vous pouvez créer un SAN virtuel qui comprend l'ensemble des périphériques de stockage local accessibles à partir d'un HBA de carte mère.

Le sous-système HBA virtuel crée inconditionnellement un LUN virtuel pour chaque LUN physique repéré. Comme dans le cas des disques virtuels, il ne doit pas arriver que des charges conflictuelles accèdent au même LUN virtuel.

Par exemple, si un port initiateur atteint dix périphériques SCSI physiques, le sous-système HBA virtuel crée dix LUN virtuels dans le domaine invité. Si le système d'exploitation de l'invité s'initialise à partir d'un de ces LUN virtuels, vous devez vous assurer qu'aucun autre domaine invité n'accède à ce LUN virtuel et que le domaine propriétaire du périphérique SCSI physique n'accède pas au LUN physique.

Une mise en garde similaire s'applique à tout LUN virtuel susceptible d'être utilisé par un domaine invité. Vous devez contrôler strictement l'accès aux LUN virtuels des autres domaines invités et l'accès au LUN physique sous-jacent sur le domaine de service pour éviter les accès conflictuels. S'ils se produisaient, ils risqueraient de corrompre les données.

Enfin, lorsque vous configurez un SAN virtuel, retenez que seuls les périphériques SCSI cible avec un LUN 0 comportent des LUN physiques visibles dans le domaine invité. Cette contrainte est imposée par l'implémentation SE Oracle Solaris qui requiert un LUN 0 cible pour répondre à la commande REPORT LUNS du SCSI.

Configuration de la fonctionnalité de chemins d'accès multiples pour HBA SCSI virtuel

Le sous-système HBA SCSI virtuel prend en charge la fonctionnalité de chemins d'accès multiples dans le domaine invité et dans le domaine de service grâce à l'implémentation de la fonctionnalité multipathing d'E/S d'Oracle Solaris. Pour plus d'informations, reportez-vous à *Managing SAN Devices and Multipathing in Oracle Solaris 11.3*.

Comme dans la fonctionnalité multipathing d'E/S d'Oracle Solaris, un ou plusieurs chemins permettent d'accéder à un périphérique SCSI backend. Pour le sous-système HBA SCSI virtuel, chaque chemin est associé à un LUN virtuel. Le module `scsi_vhci` implémente le comportement de la fonctionnalité multipathing d'E/S d'Oracle Solaris, qui envoie des demandes d'E/S à l'ensemble de LUN virtuels en fonction des arguments transmis à la commande `mpathadm` associée. Pour plus d'informations, reportez-vous aux pages de manuel `scsi_vhci`(7D) and `mpathadm`(1M).

Quand la fonctionnalité de chemins d'accès multiples est activée dans le domaine de service, comme l'illustre la Figure 10, "Configuration de la fonctionnalité de chemins d'accès multiples pour HBA SCSI virtuel dans un domaine de service", la commande `ldm add-vsan` permet de créer une instance `vsan` qui représente tous les chemins référençant les périphériques SCSI accessibles via un port initiateur spécifié. En revanche, quand la fonctionnalité de chemins d'accès multiples est désactivée dans le domaine de service, l'instance `vsan` représente uniquement les chemins indiqués par le port initiateur et référence les périphériques SCSI.

Pour configurer la fonctionnalité de chemins d'accès multiples, vous devez configurer deux chemins d'accès distincts ou plus à partir du domaine invité ou du domaine de service pour le même périphérique backend. Notez que la fonctionnalité multipathing fonctionne même avec un seul chemin configuré. Toutefois, la configuration attendue a deux ou plusieurs chemins qui envoient leurs demandes d'E/S via des ports initiateurs de HBA SCSI physique distincts qui résident sur des domaines de service distincts.

1. Exécutez une paire de commandes `ldm add-vhba` et `ldm add-vsan` pour chaque chemin distinct au stockage backend.
2. Activez la fonctionnalité multipathing d'E/S d'Oracle Solaris dans le domaine invité pour les ports initiateurs qui sont gérés par le module HBA virtuel `vhba`.

Voici un exemple d'une configuration de fonctionnalité multipathing dans un domaine invité. Elle présente un LUN physique d'un SAN accessible par deux chemins qui sont gérés par la fonctionnalité multipathing d'E/S d'Oracle Solaris. Pour la procédure décrivant comment créer la configuration présentée dans cette figure, reportez-vous à "Procédure de configuration de la fonctionnalité multipathing pour HBA SCSI virtuel" à la page 237.

FIGURE 9 Configuration de la fonctionnalité de chemins d'accès multiples pour HBA SCSI virtuel dans un domaine invité

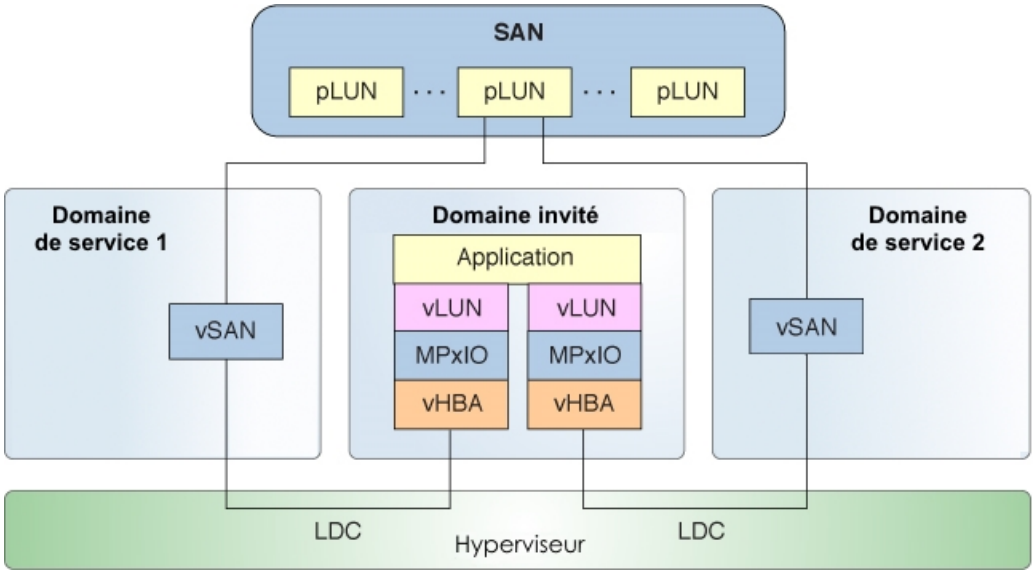
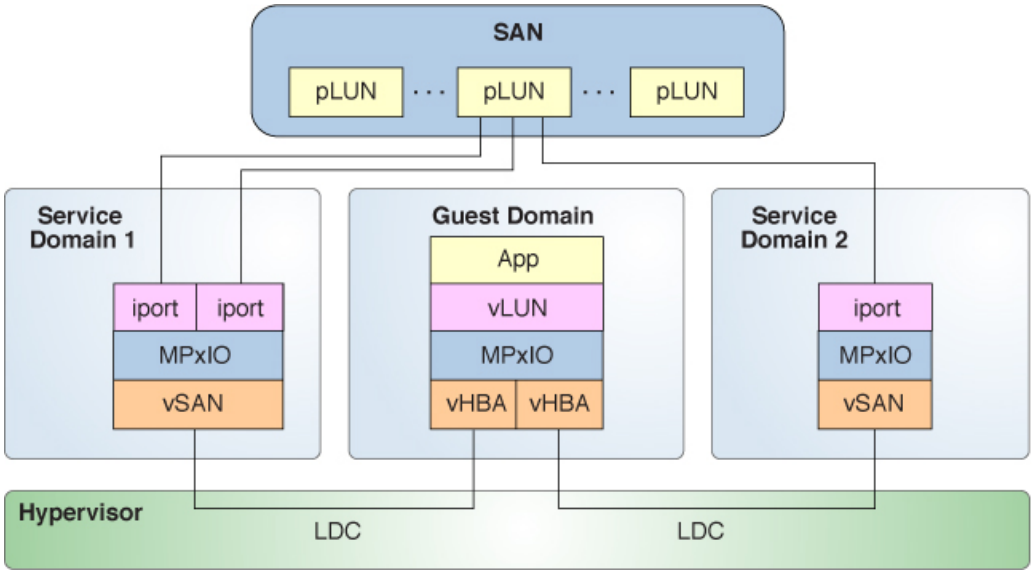


FIGURE 10 Configuration de la fonctionnalité de chemins d'accès multiples pour HBA SCSI virtuel dans un domaine de service



Procédure de configuration de la fonctionnalité multipathing pour HBA SCSI virtuel

Cette procédure décrit comment créer la configuration de la fonctionnalité de chemins d'accès multiples pour HBA SCSI virtuel illustrée dans la Figure 9, "Configuration de la fonctionnalité de chemins d'accès multiples pour HBA SCSI virtuel dans un domaine invité". Cet exemple n'est qu'une des nombreuses possibilités de configuration de la fonctionnalité de chemins d'accès multiples.

1. **Créez un domaine d'E/S avec le HBA SCSI physique qui lui est assigné.**
   Reportez-vous à la section Chapitre 6, Configuration de domaines d'E/S.

2. **Exportez le SAN virtuel pour le premier chemin de port initiateur à partir du premier domaine de service.**

   `ldm add-vsan` *vSAN-path1 vSAN-name domain-name*

vSAN-path1 est le premier chemin de port initiateur pour le SAN.

Par exemple, la commande suivante exporte le SAN virtuel `vsan-mpxio1` à partir du domaine `svcdom1` :

```
primary# ldm add-vsan /SYS/MB/RISER0/PCIE1/HBA0/PORT0 vsan-mpxio1 svcdom1
```

3. **Exportez le SAN virtuel pour le second chemin de port initiateur à partir du second domaine de service.**

   `ldm add-vsan` *vSAN-path2 vSAN-name domain-name*

   vSAN-path2 est le second chemin de port initiateur pour le SAN.

   Par exemple, la commande suivante exporte le SAN virtuel `vsan-mpxio2` à partir du domaine `svcdom2` :

   ```
   primary# ldm add-vsan /SYS/MB/RISER0/PCIE3/HBA0/PORT0 vsan-mpxio2 svcdom2
   ```

4. **Exportez les HBA SCSI virtuels vers le domaine invité.**

   `ldm add-vhba` *vHBA-name vSAN-name domain-name*

   Par exemple, les commandes suivantes exportent les HBA SCSI virtuels `vhba-mpxio1` et `vhba-mpxio2` vers le domaine `gdom` :

   ```
   primary# ldm add-vhba vhba-mpxio1 vsan-mpxio1 gdom
   primary# ldm add-vhba vhba-mpxio2 vsan-mpxio2 gdom
   ```

5. **Spécifiez la valeur de la propriété `timeout` pour les HBA SCSI virtuels sur le domaine invité.**

   `ldm set-vhba timeout=`*seconds vHBA-name domain-name*

   Par exemple, les commandes suivantes définissent la valeur de la propriété `timeout` à `30` pour les HBA SCSI virtuels `vsan-mpxio1` et `vsan-mpxio2` sur le domaine `gdom` :

   ```
   primary# ldm set-vhba timeout=30 vhba-mpxio1 gdom
   primary# ldm set-vhba timeout=30 vhba-mpxio2 gdom
   ```

6. **Réinitialisez le domaine invité.**

▼ Activation de la fonctionnalité de chemins d'accès multiples pour des HBA SCSI virtuels dans un domaine invité

1. **Copiez le fichier `/platform/sun4v/kernel/drv/vhba.conf` dans le répertoire `/etc/driver/drv`.**

```
# cp /platform/sun4v/kernel/drv/vhba.conf /etc/driver/drv
```

2. **Activez la fonctionnalité multipathing d'E/S d'Oracle Solaris en modifiant le fichier `/etc/driver/drv/vhba.conf` pour définir la propriété `mpxio-disable` sur `no`.**

3. **Réinitialisez le domaine invité.**

Désactivation de la fonctionnalité de chemins d'accès multiples pour des HBA SCSI virtuels dans un domaine invité

1. **Copiez le fichier `/platform/sun4v/kernel/drv/vhba.conf` dans le répertoire `/etc/driver/drv`.**

```
# cp /platform/sun4v/kernel/drv/vhba.conf /etc/driver/drv
```

2. **Désactivez la fonctionnalité multipathing d'E/S d'Oracle Solaris en modifiant le fichier `/etc/driver/drv/vhba.conf` pour définir la propriété `mpxio-disable` sur `yes`.**

3. **Réinitialisez le domaine invité.**

Activation de la fonctionnalité de chemins d'accès multiples pour des HBA SCSI virtuels dans un domaine de service

1. **Activez la fonctionnalité multipathing d'E/S d'Oracle Solaris pour tous les ports initiateurs dans le domaine de service.**

```
svcdom# stmsboot -e
```

Pour plus d'informations, reportez-vous à la section "Enabling and Disabling Multipathing" du manuel *Oracle Solaris SAN Configuration and Multipathing Guide*.

2. **Répertoriez les périphériques SCSI accessibles à partir de chaque port initiateur pour un domaine de service.**

Par exemple, la commande `ldm list-hba` peut afficher les informations suivantes sur le domaine de service 1, comme l'illustre la Figure 10, "Configuration de la fonctionnalité de chemins d'accès multiples pour HBA SCSI virtuel dans un domaine de service".

```
primary# ldm list-hba -d svcdom
DOMAIN
svcdom

IPORT                                                VSAN
-----                                                ----
/SYS/MB/PCIE0/HBA0/PORT1
  c0t600110D000211501010900001061ADBF4d0
  c0t600110D000211501010900000146489D34d0
/SYS/MB/PCIE1/HBA0/PORT1
  c0t600110D000211501010900001061ADBF4d0
  c0t600110D000211501010900000146489D34d0
```

3. Créez une instance de réseau de stockage SAN virtuel qui référence un port initiateur donné.

Dans la commande suivante, le port initiateur référence deux périphériques SCSI qui sont également référencés par `PCIE0`. L'un ou l'autre des ports initiateurs qui ont des chemins d'accès physiques aux même LUN peuvent être utilisés dans le cadre de la commande `ldm add-vsan` pour configurer le réseau de stockage SAN virtuel si la fonctionnalité de chemins d'accès multiples est activée.

```
primary# ldm add-vsan /SYS/MB/PCIE1/HBA0/PORT1 my_mpxio_vsan svcdom
```

4. Ajoutez le SAN virtuel au domaine invité dans un HBA SCSI virtuel.

```
primary# ldm add-vsan my_vhba my_mpxio_vsan gdom
```

5. Affichez les périphériques physiques en exécutant la commande `format` dans le domaine de service.

La sortie suivante présente deux périphériques SCSI physiques, chacun pouvant avoir un ou plusieurs chemins d'accès.

```
svcdom# format
Searching for disks...done

AVAILABLE DISK SELECTIONS:
       0. c0t600110D000211501010900001061ADBF4d0 <SANBlaze-VLUN P0T1L7-V7.3-1.00GB>
          /scsi_vhci/ssd@g600110d000211501010900001061adbf4
       1. c0t600110D000211501010900000146489D34d0 <SANBlaze-VLUN P0T1L6-V7.3-1.00GB>
          /scsi_vhci/ssd@g600110d000211501010900000146489d34
       2. c1d0 <SUN-DiskImage-10GB cyl 282 alt 2 hd 96 sec 768>
          /virtual-devices@100/channel-devices@200/disk@0
Specify disk (enter its number):
```

Cette commande indique que la configuration du domaine de service présente deux chemins d'accès à chaque périphérique physique.

```
svcdom# mpathadm list lu
        /dev/rdsk/c0t600110D000211501010900001061ADBF4d0s2
                Total Path Count: 2
                Operational Path Count: 2
        /dev/rdsk/c0t600110D000211501010900000146489D34d0s2
                Total Path Count: 2
                Operational Path Count: 2
```

Notez que la sortie pour `format` dans le domaine invité est identique car la fonctionnalité multipathing d'E/S d'Oracle Solaris s'exécute à la fois dans le domaine invité et le domaine de service. Cette fonctionnalité crée également un chemin de périphérique qui utilise le numéro d'adresse mondiale de l'unité logique, soit `g600110d000211501010900014648 9d34`, dans la sortie suivante :

```
gdom# format
Searching for disks...done

AVAILABLE DISK SELECTIONS:
       0. c0t600110D000211501010900014648 9D34d0 <SANBlaze-VLUN P0T1L6-V7.3-1.00GB>
          /scsi_vhci/disk@g600110d000211501010900014648 9d34
       1. c0t600110D000211501010900001061ADBF4d0 <SANBlaze-VLUN P0T1L7-V7.3-1.00GB>
          /scsi_vhci/disk@g600110d000211501010900001061adbf4
       2. c1d0 <SUN-DiskImage-10GB cyl 282 alt 2 hd 96 sec 768>
          /virtual-devices@100/channel-devices@200/disk@0
Specify disk (enter its number):
```

▼ Désactivation de la fonctionnalité de chemins d'accès multiples pour des HBA SCSI virtuels dans des domaines de service

- **Désactivez la fonctionnalité multipathing d'E/S d'Oracle Solaris pour tous les ports initiateurs dans le domaine de service.**

```
svcdom# stmsboot -d
```

Par ailleurs, vous pouvez activer ou désactiver la fonctionnalité multipathing d'E/S d'Oracle Solaris par port initiateur dans le domaine de service. Pour plus d'informations, reportez-vous à la section "Enabling or Disabling Multipathing on a Per-Port Basis" du manuel *Oracle Solaris SAN Configuration and Multipathing Guide*.

Initialisation à partir de périphériques SCSI

Les sections suivantes décrivent l'initialisation à partir de périphériques SCSI :

- "Initialisation à partir d'un LUN virtuel" à la page 242
- "Initialisation à partir d'un périphérique DVD SCSI" à la page 243

Initialisation à partir d'un LUN virtuel

Vous pouvez initialiser tout LUN virtuel dont le LUN physique associé référence un type de périphérique SCSI amorçable par OBP, tel qu'un CD, DVD ou disque.

Remarque - Le microprogramme système des serveurs UltraSPARC T2, UltraSPARC T2 Plus et SPARC T3 ne prend pas en charge l'initialisation d'un HBA SCSI virtuel.

Avant d'émettre la commande `boot` à l'invite OpenBoot PROM, exécutez la commande `probe-scsi-all` pour rechercher les HBA SCSI virtuels du domaine invité et les LUN virtuels associés.

L'exemple annoté suivant met en évidence les parties pertinentes de la sortie :

```
{0} ok probe-scsi-all

/virtual-devices@100/channel-devices@200/scsi@0                ligne 1

vHBA

TPORT-PHYS: w200200110d214900                                  ligne 2
  LUN: 1   Disk     VLUN          2097152 Blocks, 1073 MB
  LUN: 0   Disk     VLUN          32768000 Blocks, 16 GB       ligne 3
```

Cet exemple de sortie `probe-scsi-all` illustre une instance de HBA SCSI virtuel (`scsi@0`) ayant deux LUN de type `disk`.

Pour une initialisation à partir d'un LUN virtuel spécifique, composez manuellement le chemin du périphérique à transmettre à la commande `boot`. Le chemin du périphérique a la syntaxe suivante :

vhba-device-path`/disk@`*target-port*`,`*lun*`:`*slice*

Pour une initialisation à partir du LUN de la ligne 3, composez le chemin du périphérique comme suit :

- Prenez la valeur de *target-port* de la ligne 2
- Prenez la valeur de *vhba-device-path* de la ligne 1

Le chemin du périphérique qui en résulte est le suivant :

```
/virtual-devices@100/channel-devices@200/scsi@0/disk@w200200110d214900,0
```

Vous pouvez transmettre ce chemin de périphérique à la commande `boot` d'OBP comme suit :

```
{0} ok boot /virtual-devices@100/channel-devices@200/scsi@0/disk@w200200110d214900,0
```

Initialisation à partir d'un périphérique DVD SCSI

Vous pouvez effectuer l'initialisation à partir d'un disque DVD SCSI pour installer le domaine invité à partir de ce DVD.

L'exemple suivant illustre la configuration d'un HBA SCSI virtuel ayant un périphérique DVD SCSI associé au domaine `primary`.

```
primary# ldm list-hba -t -d primary
IPORT                                                VSAN
-----                                                ----
[...]
/SYS/MB/SASHBA1/HBA0/PORT40
        init-port w50800200008f4329
        Transport Protocol SAS
  c2t3d0s0
        lun 0
        removable media 1

primary# ldm add-vsan /SYS/MB/SASHBA1/HBA0/PORT40 dvd_vsan primary
/SYS/MB/SASHBA1/HBA0/PORT40 resolved to device: /pci@400/pci@2/pci@0/pci@4/scsi@0/iport@40
primary# ldm add-vhba dvd_vhba dvd_vsan gdom
```

A partir de la console de domaine `gdom`, sondez les périphériques SCSI et lancez l'initialisation depuis le DVD.

```
{0} ok probe-scsi-all
/virtual-devices@100/channel-devices@200/scsi@0

vHBA

TPORT-PHYS: p3
   LUN: 0   Removable Read Only device     TEAC    DV-W28SS-V 1.0B

{0} ok boot /virtual-devices@100/channel-devices@200/scsi@0/disk@p3
...
```

Installation d'un LUN virtuel

Vous pouvez installer un système d'exploitation sur tout LUN virtuel dont le LUN physique associé référence un périphérique SCSI dont le type est pris en charge par le programme d'installation. Vous pouvez alors procéder à l'initialisation à partir du LUN virtuel spécifié.

Délai d'attente d'un HBA SCSI virtuel

Par défaut, si le domaine de service fournissant l'accès au SAN virtuel est indisponible, toutes les E/S du domaine invité vers le HBA SCSI virtuel correspondant sont bloquées. Les E/S reprennent automatiquement quand le domaine de service est de nouveau opérationnel et rétablit le service pour le SAN virtuel.

Parfois, si le domaine de service est indisponible pendant trop longtemps, les systèmes de fichiers ou applications peuvent nécessiter l'échec d'une opération d'E/S et la signalisation d'une erreur. Vous pouvez définir une période de délai d'attente de connexion pour chaque HBA SCSI virtuel, qui peut ensuite être utilisée pour établir une connexion entre le HBA SCSI virtuel sur un domaine invité et le SAN virtuel sur le domaine de service. Lorsque cette période de délai d'attente est atteinte, toutes les E/S en attente ou toutes les nouvelles opérations d'E/S échouent tant que le domaine de service est indisponible et que la connexion entre le HBA SCSI virtuel et le SAN virtuel n'est pas rétablie.

Vous pouvez également spécifier une valeur de délai d'attente dans d'autres cas, notamment :

- Si vous voulez que la fonctionnalité multipathing d'E/S d'Oracle Solaris bascule vers un autre chemin configuré, vous devez définir le délai d'expiration de chaque HBA SCSI virtuel concerné.
- Si vous effectuez une migration directe, définissez la valeur de la propriété `timeout` sur `0` pour chaque HBA SCSI virtuel dans le domaine invité à migrer. Une fois la migration terminée, rétablissez la propriété de délai d'attente à la valeur d'origine pour chaque HBA SCSI virtuel.

Pour plus d'informations sur la définition de cette valeur, reportez-vous à "Définition du délai d'attente du HBA SCSI virtuel" à la page 232.

HBA SCSI virtuel et SCSI

Le module `vhba` sert de proxy pour les commandes SCSI pour le pilote de HBA SCSI qui est associé au port initiateur SCSI du SAN virtuel.

Le pilote `scsi_vhci`, qui implémente la fonctionnalité multipathing d'E/S d'Oracle Solaris gère la persistance des réservations lors du basculement des chemins des réservations SCSI-2 et SCSI-3. Le module `vhba` se connecte à la structure d'E/S d'Oracle Solaris et prend donc en charge les réservations SCSI en tirant parti du support `scsi_vhci`.

Prise en charge de tampons d'E/S très fragmentés dans le domaine invité

A l'instar des autres périphériques virtuels `sun4v`, le module `vhba` exploite un tampon d'E/S créé par les couches supérieures de la pile logicielle. Si ce tampon d'E/S est une agrégation d'un trop grand nombre de fragments de mémoire physique, le module `vhba` émet le message d'avertissement suivant lors du traitement de la demande d'E/S :

```
WARNING: ... ldc_mem_bind_hdl: ncookies(max, actual) = (8, 9)
```

Chaque fragment de mémoire physique est associé à un cookie. Si le nombre réel de cookies ne peut pas être pris en charge par le nombre maximal de cookies, la demande d'E/S échoue.

Le message d'erreur affiche le nombre réel de cookies requis. Pour résoudre cette erreur, modifiez la valeur de `vhba_desc_ncookies` dans le fichier `/etc/system`, qui spécifie le nombre de cookies par tampon d'E/S, pour qu'elle soit aussi importante que la valeur réelle. En outre, augmentez la valeur de la propriété `vhba_desc_max_ncookies`, qui spécifie le nombre maximal de cookies autorisés.

Pour plus d'informations sur la création ou la mise à jour des valeurs de propriété `/etc/system`, reportez-vous à la section "Mise à jour des valeurs de propriété dans le fichier `/etc/system`" à la page 409.

Vous pouvez ensuite recréer la connexion du HBA SCSI virtuel en exécutant une séquence de commandes `ldm remove-vhba` et `ldm add-vhba` cou en réinitialisant le domaine invité.

Par exemple, pour définir la valeur de propriété `vhba_desc_max_ncookies` sur `12`, ajoutez la ligne suivante au fichier `/etc/system` :

```
set vhba:vhba_desc_ncookies = 12
```

♦ ♦ ♦ CHAPITRE 13

Utilisation des réseaux virtuels

Ce chapitre décrit comment utiliser un réseau virtuel avec le logiciel Oracle VM Server for SPARC et aborde les rubriques suivantes :

La gestion de réseau du SE Oracle Solaris a beaucoup évolué entre le SE Oracle Solaris 10 et le SE Oracle Solaris 11. Pour plus d'informations sur les aspects à prendre en compte, reportez-vous aux sections "Présentation de la gestion de réseau dans Oracle Solaris 10" à la page 251, "Présentation de la gestion de réseau dans Oracle Solaris 11" à la page 248 et "Différences liées aux fonctions de gestion réseau Oracle Solaris 11" à la page 317.

Introduction au réseau virtuel

Un réseau virtuel permet aux domaines de communiquer les uns avec les autres sans utiliser de réseaux physiques externes. Un réseau virtuel peut également permettre aux domaines d'utiliser la même interface réseau physique pour accéder à un réseau physique et communiquer avec des systèmes distants. Un réseau virtuel est créé en ayant un commutateur virtuel auquel vous pouvez connecter des périphériques réseau virtuels.

La gestion de réseau d'Oracle Solaris varie considérablement entre le SE Oracle Solaris 10 et le SE Oracle Solaris 11. Les deux sections suivantes fournissent des informations générales sur la gestion de réseau pour chacun des systèmes d'exploitation.

Remarque - La gestion de réseau d'Oracle Solaris 10 fonctionne de la même manière que dans un domaine ou un système. Il en va de même pour la gestion de réseau d'Oracle Solaris 11. Pour plus d'informations sur la gestion de réseau dans le SE Oracle Solaris, reportez-vous à la Oracle Solaris 10 Documentation et Oracle Solaris 11.3 Documentation.

Les différences de fonctions entre la gestion de réseau dans Oracle Solaris 10 et dans Oracle Solaris 11 sont décrites à la section "Présentation de la gestion de réseau dans Oracle Solaris 11" à la page 248.

Présentation de la gestion de réseau dans Oracle Solaris 11

Le système d'exploitation Oracle Solaris 11 a introduit un grand nombre de nouvelles fonctions de gestion de réseau, lesquelles sont décrites dans la documentation relative à la gestion de réseau Oracle Solaris 11 accessible à l'adresse Oracle Solaris 11.3 Documentation.

Il est important de bien comprendre les fonctions de gestion de réseau Oracle Solaris 11 suivantes lorsque l'on utilise le logiciel Oracle VM Server for SPARC :

- L'ensemble de la configuration du réseau est effectuée à l'aide des commandes `ipadm` et `dladm`.

- La fonction “vanity name by default” (nom propre par défaut) génère des noms de lien génériques tels que `net0` pour tous les adaptateurs réseau physiques. Cette fonction génère également des noms génériques pour les commutateurs virtuels (`vsw`*n*) et les périphériques réseau virtuels (`vnet`*n*), lesquels apparaissent comme des adaptateurs réseau physiques au système d'exploitation. Pour identifier le nom de liaison générique associé à un périphérique réseau physique, utilisez la commande `dladm show-phys`.

  Dans Oracle Solaris 11, les noms de périphériques réseau physiques utilisent par défaut des noms propres génériques. Des noms génériques tels que `net0` sont utilisés à la place de noms de pilotes des périphériques tels que `nxge0` qui étaient utilisés dans Oracle Solaris 10.

  La commande suivante crée un commutateur virtuel pour le domaine `primary` en spécifiant le nom générique, `net0`, au lieu d'un nom de pilote tel que `nxge0` :

  ```
  primary# ldm add-vsw net-dev=net0 primary-vsw0 primary
  ```

- Le SE Oracle Solaris 11 utilise des cartes d'interface réseau virtuelles (VNIC) pour créer des réseaux virtuels internes.

  Une VNIC est une instanciation virtuelle d'un périphérique réseau physique pouvant être créée à partir du périphérique réseau physique et assignée à une zone.

-

  Utilisez le profil de configuration réseau (NCP) `DefaultFixed` d'Oracle Solaris 11 lors de la configuration du logiciel Oracle VM Server for SPARC.

  Pour les domaines Oracle Solaris 11, utilisez le NCP `DefaultFixed`. Vous pouvez activer ce profil pendant ou après l'installation. Pendant une installation d'Oracle Solaris 11, sélectionnez la configuration de gestion des réseaux Manuelle.

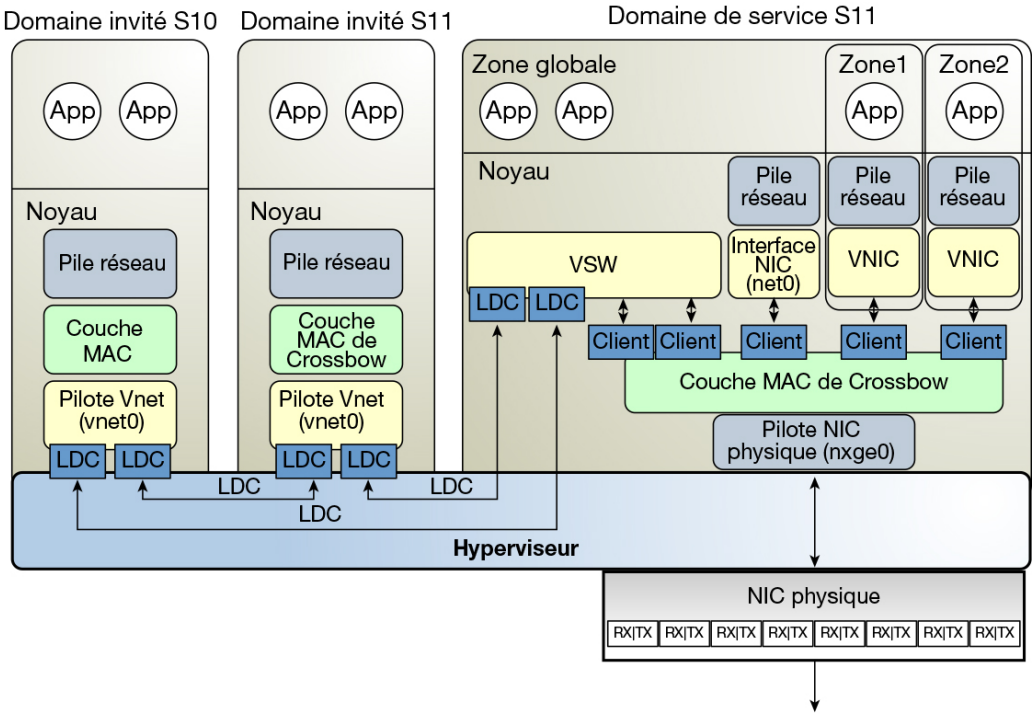
- Ne remplacez pas l'interface réseau principale par l'interface de commutateur virtuel (`vsw`). Le domaine de service peut utiliser l'interface réseau principale existante pour communiquer avec les réseaux invités possédant des périphériques réseau virtuels connectés au même commutateur virtuel.
- N'utilisez pas l'adresse MAC de l'adaptateur du réseau physique pour le commutateur virtuel, car l'utilisation de cette adresse MAC est incompatible avec l'interface réseau principale.

Remarque - Dans cette version, il est préférable d'utiliser le NCP `DefaultFixed` pour configurer les liaisons de données et les interfaces réseau sur les systèmes Oracle Solaris 11 à l'aide de la commande `dladm` ou `ipadm`.

Assurez-vous à l'aide de la commande `netadm list` que le NCP `DefaultFixed` est activé. Reportez-vous au Chapitre 7, "Using Datalink and Interface Configuration Commands on Profiles" du manuel *Oracle Solaris Administration: Network Interfaces and Network Virtualization*.

Le schéma suivant indique qu'un domaine invité exécutant le système d'exploitation Oracle Solaris 10 est parfaitement compatible avec un domaine de service Oracle Solaris 11. Les seules différences sont dues aux fonctionnalités ajoutées ou améliorées dans le SE Oracle Solaris 11.

FIGURE 11 Présentation du réseau Oracle VM Server for SPARC pour le SE Oracle Solaris 11



Le schéma montre que des noms de périphériques réseau tels que `nxge0` et `vnet0` peuvent être représentés par des noms de lien génériques tels que `net`*n* dans les domaines Oracle Solaris 11. En outre, notez les points suivants :

- Le commutateur virtuel du domaine de service est connecté aux domaines invités, ce qui permet à ces derniers de communiquer entre eux.
- Le commutateur virtuel est également connecté au périphérique réseau physique `nxge0`, ce qui permet aux domaines invités de communiquer avec le réseau physique.

  Le commutateur virtuel permet également aux domaines hôtes de communiquer avec l'interface réseau du domaine de service `net0` et avec des VNIC sur le même périphérique

réseau physique que `nxge0`. Ceci inclut la communication entre les domaines invités et le domaine de service Oracle Solaris 11. Ne configurez pas le commutateur virtuel lui-même (périphérique `vsw`*n*) en tant que périphérique réseau, car cette fonctionnalité a été abandonnée et n'est plus prise en charge dans Oracle Solaris 11.

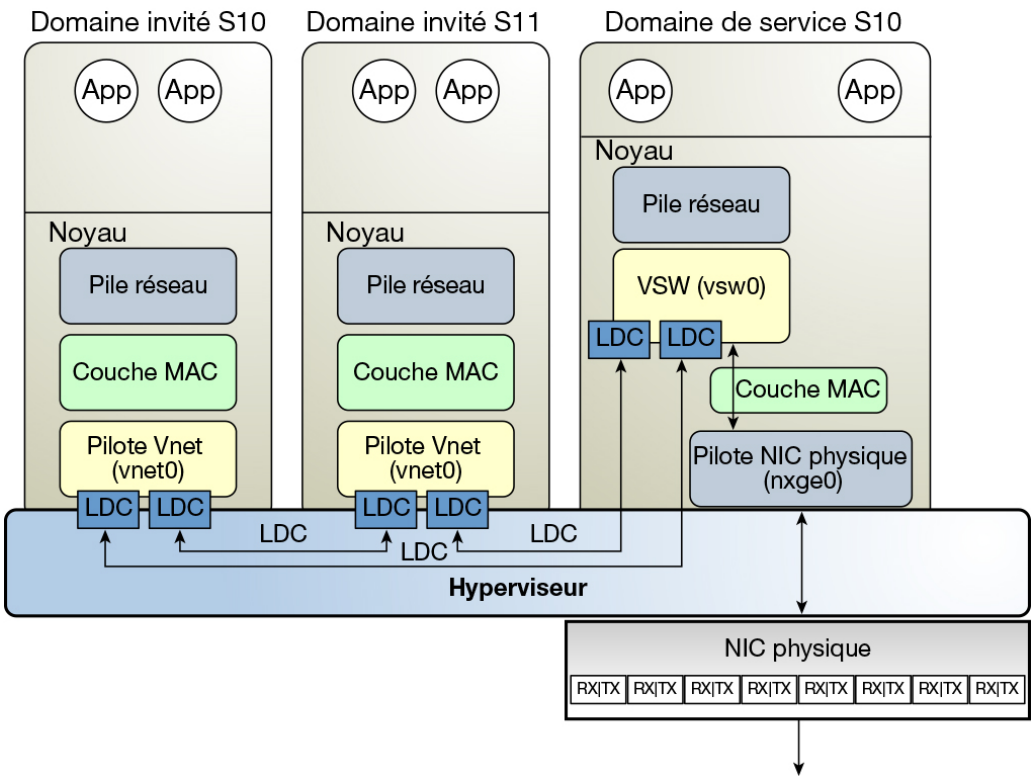
- Le périphérique de réseau virtuel `vnet0` dans un domaine invité Oracle Solaris 10 peut être configuré en tant qu'interface réseau à l'aide de la commande `ifconfig`.
- Le périphérique de réseau virtuel `vnet0` dans un domaine invité Oracle Solaris 11 peut apparaître avec un nom de lien générique tel que `net0`. Il peut être configuré en tant qu'interface réseau à l'aide de la commande `ipadm`.

Un commutateur virtuel se comporte comme un commutateur de réseau physique classique et commute les paquets du réseau entre les différents systèmes auxquels il est connecté. Un système peut être un domaine invité, un domaine de service ou un réseau physique.

Présentation de la gestion de réseau dans Oracle Solaris 10

Le schéma suivant indique qu'un domaine invité exécutant le système d'exploitation Oracle Solaris 11 est parfaitement compatible avec un domaine de service Oracle Solaris 10. Les seules différences sont dues aux fonctionnalités ajoutées ou améliorées dans le SE Oracle Solaris 11.

FIGURE 12 Présentation du réseau Oracle VM Server for SPARC pour le SE Oracle Solaris 10



Le schéma précédent affiche des noms d'interface tels que `nxge0`, `vsw0` et `vnet0` qui s'appliquent uniquement au SE Oracle Solaris 10. En outre, notez les points suivants :

- Le commutateur virtuel du domaine de service est connecté aux domaines invités, ce qui permet à ces derniers de communiquer entre eux.
- Le commutateur virtuel est également connecté à l'interface réseau physique `nxge0`, ce qui permet aux domaines invités de communiquer avec le réseau physique.
- L'interface réseau du commutateur virtuel `vsw0` est créée dans le domaine de service de façon à permettre aux deux domaines invités de communiquer avec le domaine de service.
- L'interface réseau du commutateur virtuel `vsw0` dans le domaine de service peut être configurée à l'aide de la commande `ifconfig` d'Oracle Solaris 10.
- Le périphérique de réseau virtuel `vnet0` dans un domaine invité Oracle Solaris 10 peut être configuré en tant qu'interface réseau à l'aide de la commande `ifconfig`.

- Le périphérique de réseau virtuel `vnet0` dans un domaine invité Oracle Solaris 11 peut apparaître avec un nom de lien générique tel que `net0`. Il peut être configuré en tant qu'interface réseau à l'aide de la commande `ipadm`.

Le commutateur virtuel se comporte comme un commutateur de réseau physique classique et commute les paquets du réseau entre les différents systèmes, tels que les domaines invités, le domaine de service et le réseau physique auquel il est connecté. Le pilote `vsw` fournit la fonction de périphérique réseau permettant au commutateur d'être configuré en tant qu'interface réseau.

Optimisation des performances de réseau virtuel

Vous pouvez obtenir des taux de transfert élevés pour les réseaux invités ou externes et pour les communications entre invités lorsque vous configurez votre plate-forme et les domaines comme indiqué dans cette section. La pile réseau virtuel prend en charge les LSO (large segment offload), ce qui permet d'obtenir des performances TCP élevées sans avoir besoin d'utiliser des trames géantes.

Configurations matérielle et logicielle requises

Vous devez respecter les exigences suivantes pour optimiser les performances réseau de vos domaines :

- **Configuration matérielle.** Ces améliorations de performances sont disponibles à partir du serveur SPARC T4.
- **Configuration requise du microprogramme système.** Ces systèmes SPARC doivent exécuter la dernier microprogramme du système. Voir la section "Versions complètes du microprogramme système" du manuel *Guide d'installation d'Oracle VM Server for SPARC 3.4*.
- **Configuration SE Oracle Solaris requise.** Assurez-vous que le domaine de service et le domaine invité exécutent les versions de SE Oracle Solaris suivantes.

Remarque - L'exécution de la version complète de SE Oracle Solaris vous fournit l'accès aux nouvelles fonctions. Reportez-vous à la section "Versions du Système d'exploitation Oracle Solaris complètes" du manuel *Guide d'installation d'Oracle VM Server for SPARC 3.4*.

  - **Domaine de service.** Au moins le SE Oracle Solaris 11.1 SRU 9 ou le SE Oracle Solaris 10 avec le patch 150031-03.

  - **Domaine invité.** Au moins le SE Oracle Solaris 11.1 SRU 9 ou le SE Oracle Solaris 10 avec le patch 150031-03.
- **Configuration requise de la CPU et de la mémoire.** Assurez-vous d'attribuer des ressources de CPU et de mémoire suffisantes au domaine de service et aux domaines invités.
  - **Domaine de service.** Comme le domaine de service joue un rôle de proxy de données pour les domaines invités, attribuez au moins 2 coeurs de CPU et au moins 16 Go de mémoire au domaine de service.
  - **Domaine invité.** Configurez chaque domaine invité pour garantir des performances d'au moins 10 Gbits. Attribuez au moins 2 coeurs de CPU et au moins 4 Go de mémoire à chaque domaine invité.

Configuration de vos domaines pour optimiser les performances de votre réseau virtuel

Dans les versions précédentes de Oracle VM Server for SPARC et dans SE Oracle Solaris, vous pouvez améliorer vos performances réseau en configurant les trames géantes. Cette configuration n'est plus requise et même si elle le devient pour une autre raison, il est recommandé d'utiliser la valeur MTU standard de 1500 pour vos domaines de service et invités.

Pour obtenir de meilleures performances de gestion de réseau, définissez la propriété `extended-mapin-space` sur `on` pour le domaine de service et les domaines invités. Il s'agit du comportement par défaut.

```
primary# ldm set-domain extended-mapin-space=on domain-name
```

Pour vérifier la valeur de propriété `extended-mapin-space`, exécutez la commande suivante :

```
primary# ldm list -l domain-name |grep extended-mapin
extended-mapin-space=on
```

Remarque - Une modification de la valeur de propriété `extended-mapin-space` déclenche une reconfiguration retardée sur le domaine `primary`. Cette situation requiert une réinitialisation du domaine `primary`. Vous devez également arrêter les domaines invités avant de modifier cette valeur de propriété.

Commutateur virtuel

Un commutateur virtuel (`vsw`) est un composant s'exécutant dans un domaine de service et géré par le pilote du commutateur virtuel. Un commutateur virtuel peut être connecté à certains domaines invités pour permettre les communications réseau entre ces domaines. Par ailleurs, si le commutateur virtuel est également associé à une interface réseau physique, la communication réseau entre les domaines invités et le réseau physique est autorisée via l'interface réseau physique. Lors de l'exécution d'un domaine de service Oracle Solaris 10, un commutateur virtuel a également une interface réseau, `vsw`*n*, qui permet au domaine de service de communiquer avec les autres domaines connectés à ce commutateur virtuel. Le commutateur virtuel peut être utilisé comme une interface réseau classique et configuré avec la commande `ifconfig` d'Oracle Solaris 10.

L'assignation d'un périphérique de réseau virtuel à un domaine crée une dépendance implicite sur le domaine fournissant le commutateur virtuel. Vous pouvez afficher ces dépendances ou domaines dépendant du commutateur virtuel à l'aide de la commande `ldm list-dependencies`. Voir "Liste des dépendances d'E/S de domaine" à la page 430.

Dans un domaine de service Oracle Solaris 11, le commutateur virtuel ne peut pas être utilisé en tant que simple interface réseau. Si le commutateur virtuel est connecté à une interface réseau physique, la communication avec le domaine de service est possible à l'aide de cette interface physique. Si la configuration n'a pas d'interface physique, vous pouvez activer la communication avec le domaine de service à l'aide d'un `etherstub`, comme le périphérique réseau (`net-dev`) qui est connecté avec un VNIC.

Pour identifier le périphérique réseau à utiliser en tant que périphérique backend pour le commutateur virtuel, recherchez le périphérique réseau physique dans la sortie `dladm show-phys` ou répertoriez la liste des périphériques réseau des domaines logiques à l'aide de la commande `ldm list-netdev`.

Remarque - Lorsqu'un commutateur virtuel est ajouté à un domaine de service Oracle Solaris 10, son interface réseau n'est pas créée. Par conséquent, le domaine de service est incapable par défaut de communiquer avec les domaines invités connectés à son commutateur virtuel. Pour activer les communications réseau entre les domaines invités et le domaine de service, l'interface réseau du commutateur virtuel associé doit être créée et configurée dans le domaine de service. Reportez-vous à la section "Activation de la mise en réseau entre le domaine de service Oracle Solaris 10 et les autres domaines" à la page 43 pour obtenir des instructions.

Cette situation se produit *uniquement* pour le SE Oracle Solaris 10, *pas* pour le SE Oracle Solaris 11.

Vous pouvez ajouter un commutateur virtuel à un domaine, définir des options pour un commutateur virtuel et supprimer un commutateur virtuel à l'aide des commandes `ldm add-vsw`, `ldm set-vsw` et `ldm rm-vsw`, respectivement. Reportez-vous à la page de manuel `ldm`(1M).

Lorsque vous créez un commutateur virtuel sur une instance balisée VLAN de carte d'interface réseau ou de groupement, vous devez indiquer la carte d'interface réseau (`nxge0`), le groupement `aggr3` ou le nom propre (`net0`) comme valeur pour la propriété `net-dev` lorsque vous utilisez la commande `ldm add-vsw` ou `ldm set-vsw`.

Remarque - A partir du SE Oracle Solaris 11.2 SRU 1, vous pouvez mettre à jour dynamiquement la valeur de la propriété `net-dev` à l'aide de la commande `ldm set-vsw`. Dans les précédentes versions du SE Oracle Solaris, l'utilisation de la commande `ldm set-vsw` pour mettre à jour la valeur de la propriété `net-dev` du domaine `primary` déclenche l'entrée du domaine `primary` en reconfiguration retardée.

Vous ne pouvez pas ajouter de commutateur virtuel à un périphérique réseau IPoIB (IP-over-InfiniBand) InfiniBand. Bien que les commandes `ldm add-vsw` et `ldm add-vnet` semblent fonctionner, le flux de données est nul car ces périphériques transportent des paquets d'IP au moyen de la couche de transport InfiniBand. Le commutateur virtuel ne prend en charge qu'Ethernet en tant que couche de transport. Notez que les périphériques backend IPoIB et Ethernet-over-InfiniBand (EoIB) ne sont pas pris en charge pour les commutateurs virtuels.

La commande suivante crée un commutateur virtuel sur un adaptateur réseau physique nommé `net0` :

```
primary# ldm add-vsw net-dev=net0 primary-vsw0 primary
```

L'exemple suivant utilise la commande `ldm list-netdev -b` pour n'afficher que les périphériques backend de commutateur virtuel valides pour le domaine de service `svcdom`.

```
primary# ldm list-netdev -b svcdom
DOMAIN
svcdom

NAME             CLASS MEDIA STATE   SPEED OVER   LOC
----             ----- ----- -----   ----- ----   ---
net0             PHYS  ETHER up      10000 ixgbe0 /SYS/MB/RISER1/PCIE
net1             PHYS  ETHER unknown 0     ixgbe1 /SYS/MB/RISER1/PCIE4
net2             ESTUB ETHER unknown 0     --     --
net3             ESTUB ETHER unknown 0     --     --
ldoms-estub.vsw0 ESTUB ETHER unknown 0     --     --
```

Périphérique réseau virtuel

Un périphérique réseau virtuel est un périphérique virtuel qui est défini dans un domaine connecté à un commutateur virtuel. Un périphérique réseau virtuel est géré par le pilote de

réseau virtuel et il est connecté à un réseau virtuel par l'hyperviseur à l'aide des canaux de domaines logiques (LDC).

Remarque - Un domaine invité prend en charge jusqu'à 999 périphériques de réseau virtuel.

Un périphérique réseau virtuel peut être utilisé en tant qu'interface réseau avec le nom `vnet`*n*, laquelle peut être utilisée comme n'importe quelle interface réseau classique et configurée avec la commande `ifconfig` d'Oracle Solaris 10 ou la commande `ipadm` d'Oracle Solaris 11.

Remarque - Pour Oracle Solaris 11, des noms génériques sont assignés aux périphériques, donc `vnet`*n* utiliserait un nom générique tel que `net0`.

Vous pouvez ajouter un périphérique réseau virtuel à un domaine, définir des options pour un périphérique réseau virtuel existant et supprimer un périphérique réseau virtuel à l'aide des commandes `ldm add-vnet`, `ldm set-vnet` et `ldm rm-vnet`, respectivement. Reportez-vous à la page de manuel `ldm`(1M).

Pour plus d'informations sur la mise en réseau Oracle VM Server for SPARC pour Oracle Solaris 10 et Oracle Solaris 11, voir Figure 12, "Présentation du réseau Oracle VM Server for SPARC pour le SE Oracle Solaris 10" et Figure 11, "Présentation du réseau Oracle VM Server for SPARC pour le SE Oracle Solaris 11", respectivement.

Canaux LDC inter-Vnet

Par défaut, Logical Domains Manager assigne les canaux LDC de la manière suivante :

- Un canal LDC entre les périphériques réseau virtuel et le périphérique de commutateur réseau virtuel.
- Un canal LDC entre chaque paire de périphériques réseau virtuel connectés au même périphérique de commutateur virtuel (inter-vnet).

Les canaux LDC inter-vnet sont configurés de sorte à ce que les périphériques réseau virtuel puissent communiquer directement afin d'obtenir de bonnes performances de communication d'invité à invité. Cependant, lorsque le nombre de périphériques réseau virtuel dans un périphérique de commutateur virtuel augmente, le nombre de canaux LDC requis pour les communications inter-vnet augmente de manière quadratique.

Vous pouvez choisir d'activer ou de désactiver l'allocation de canal LDC inter-vnet pour tous les périphériques réseau virtuels connectés à un périphérique de commutateur virtuel donné. En désactivant cette allocation, vous pouvez réduire l'utilisation de canaux LDC, qui sont limités en nombre.

La désactivation de cette allocation peut s'avérer utile dans les cas suivants :

- Lorsque les performances de communication entre invités n'est pas de prime importance
- Lorsqu'un grand nombre de périphériques réseau virtuel sont requis dans un périphérique de commutateur virtuel

Si vous n'affectez pas de canaux inter-vnet, davantage de canaux LDC sont disponibles et peuvent être utilisés pour ajouter davantage de périphériques d'E/S virtuelles à un domaine invité.

Remarque - Si les performances entre invités s'avèrent plus importantes que l'augmentation du nombre de périphériques virtuels du système, ne désactivez pas l'allocation de canaux LDC inter-vnet.

Vous pouvez utiliser les commandes `ldm add-vsw` et `ldm set-vsw` pour spécifier une valeur `on`, `off` ou `auto` pour la propriété `inter-vnet-link`.

Par défaut, la propriété `inter-vnet-link` est définie sur `auto`, ce qui signifie que les canaux LDC inter-vnet sont alloués à moins que le nombre de réseaux virtuels dans un commutateur virtuel donné augmente au-delà de la limite maximale par défaut spécifiée par la propriété SMF `ldmd/auto_inter_vnet_link_limit`. La valeur par défaut de `ldmd/auto_inter_vnet_link_limit` est 8. Si le nombre de réseaux virtuels présents pour un commutateur virtuel est supérieur au nombre maximal, les LDC inter-vnet sont désactivés. Voir "Identification des réseaux présents dans les domaines logiques" à la page 260.

Si suite à la liaison d'un domaine invité ou à l'ajout de réseaux virtuels à un domaine lié, le nombre de réseaux virtuels dans le commutateur virtuel dépasse la limite, les LDC inter-vnet sont désactivés automatiquement. L'inverse est également vrai. Si suite à la suppression de la liaison d'un domaine invité ou au retrait de réseaux virtuels d'un domaine lié, le nombre de réseaux virtuels dans le commutateur virtuel est inférieure à la limite, les LDC inter-vnet sont activés automatiquement.

Quand `inter-vnet-link=auto`, la sortie `ldm list` affiche la valeur comme `on/auto` ou `off/auto` selon l'état actif des liaisons inter-vnet pour le commutateur virtuel.

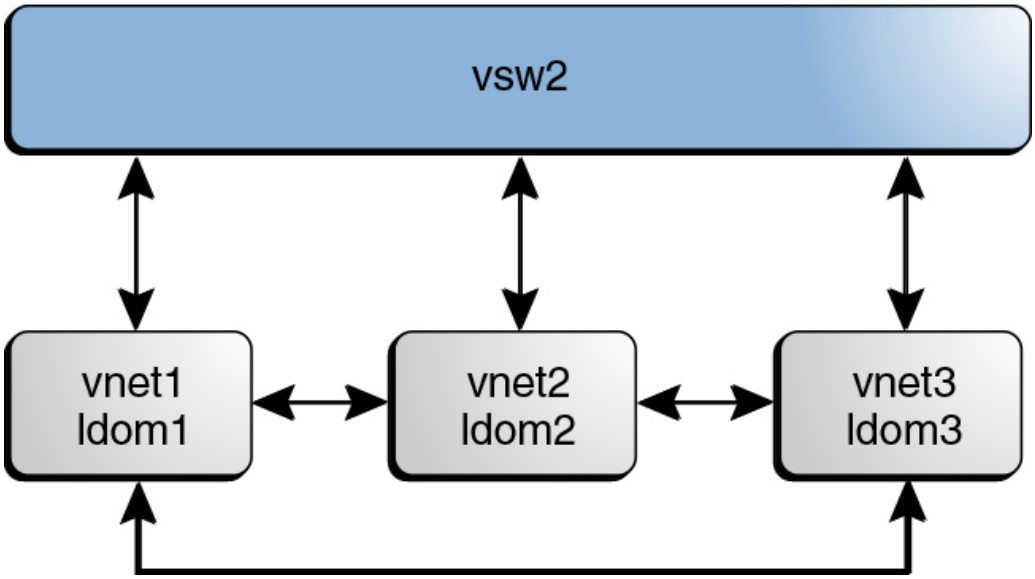
Notez que quand vous procédez à la mise à niveau de votre système vers Oracle VM Server for SPARC 3.4, la valeur `inter-vnet-link` que vous avez définie est préservée.

Les figures suivantes présentent des commutateurs virtuels type pour `inter-vnet-link=on` et `inter-vnet-link=off`, respectivement.

La figure suivante présente un commutateur virtuel typique contenant trois périphériques réseau virtuel. La propriété `inter-vnet-link` est définie sur `on`, ce qui signifie que les canaux LDC inter-vnet sont alloués. Les communications invité à invité entre `vnet1` et `vnet2` sont effectuées directement sans passer par le commutateur virtuel.

Cette figure illustre également le cas où `inter-vnet-link=auto` et le nombre de réseaux virtuels connectés au même commutateur virtuel est égal ou inférieur à la valeur maximale définie par la propriété SMF `ldmd/auto_inter_vnet_link_limit`.

FIGURE 13 Configuration du commutateur virtuel utilisant les canaux inter-vnet

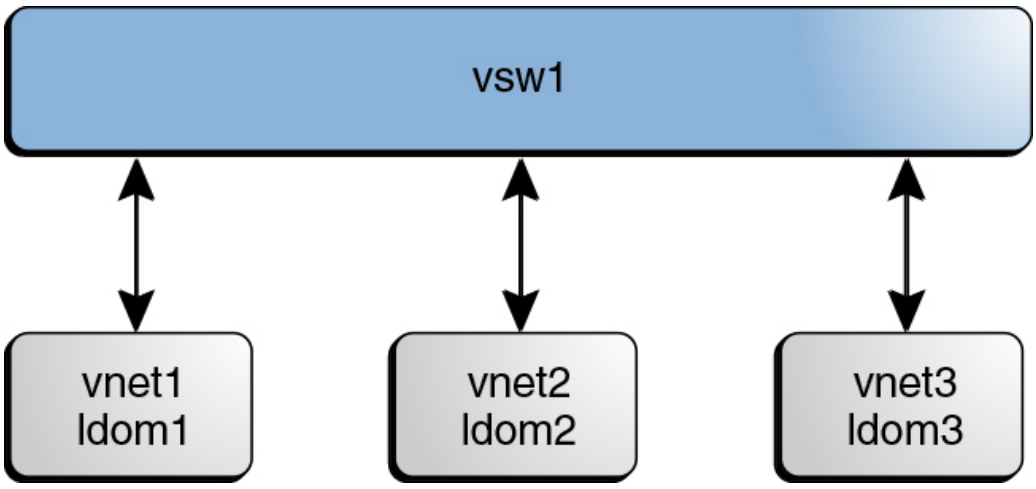


La figure suivante présente la même configuration de commutateur virtuel avec la propriété `inter-vnet-link` définie sur `off`. Les canaux LDC inter-Vnet ne sont pas alloués. Un nombre inférieur de canaux LDC est utilisé que lorsque la propriété `inter-vnet-link` est définie sur `on`. Dans cette configuration, les communications invité à invité entre `vnet1` et `vnet2` doivent passer par `vsw1`.

Cette figure illustre également le cas où `inter-vnet-link=auto` et le nombre de réseaux virtuels connectés au même commutateur virtuel dépasse la valeur maximale définie par la propriété SMF `ldmd/auto_inter_vnet_link_limit`.

Remarque - La désactivation de l'allocation de canaux LDC inter-vnet n'empêche pas la communication entre invités. Au contraire, le trafic de communication entre invités passe par le commutateur virtuel plutôt que de passer directement d'un domaine invité à un autre.

FIGURE 14 Configuration de commutateur virtuel n'utilisant pas les canaux inter-vnet



Pour plus d'informations sur les canaux LDC, reportez-vous à la section "Utilisation des canaux de domaines logiques" à la page 421.

Identification des réseaux présents dans les domaines logiques

Vous pouvez émettre des commandes à partir de l'invite OpenBoot PROM (OBP) pour répertorier les informations réseau pour les domaines logiques.

Exécutez la commande `show-nets` sur un domaine pour lister les réseaux disponibles sur ce domaine :

```
OK show-nets
```

Exécutez la commande `watch-net-all` sur le domaine de contrôle pour répertorier les réseaux disponibles et afficher le trafic réseau :

```
OK watch-net-all
```

Affichage des configurations et des statistiques des périphériques réseau

Les commandes `ldm list-netdev` et `ldm list-netstat` permettent d'afficher des informations sur les périphériques réseau figurant dans le système, ainsi que les statistiques du réseau, respectivement. Par conséquent, vous pouvez accéder à une vue centralisée des périphériques et des statistiques réseau d'un domaine physique donné.

Pour utiliser ces commandes, vous devez exécuter au moins la SRU 1 du SE Oracle Solaris 11.2 dans le domaine invité.

EXEMPLE 38 Liste des informations de configuration des périphériques réseau

L'exemple suivant présente une liste courte des périphériques réseau pour le domaine `ldg1` à l'aide de la commande `ldm list-netdev`.

```
primary# ldm list-netdev ldg1

DOMAIN
ldg1

NAME              CLASS MEDIA STATE   SPEED OVER      LOC
----              ----- ----- -----   ----- ----      ---
net0              VNET  ETHER up      0     --        primary-vsw0/vne t0_ldg1
net3              PHYS  ETHER up      10000 --        /SYS/MB/RISER1/PCIE4
net4              VSW   ETHER up      10000 --        ldg1-vsw1
net1              PHYS  ETHER up      10000 --        /SYS/MB/RISER1/PCIE4
net5              VNET  ETHER up      0     --        ldg1-vsw1/vnet1_ldg1
net6              VNET  ETHER up      0     --        ldg1-vsw1/vnet2_ldg1
aggr2             AGGR  ETHER unknown 0     net1,net3 --
ldoms-vsw0.vport3 VNIC  ETHER unknown 0     --        ldg1-vsw1/vnet2_ldg1
ldoms-vsw0.vport2 VNIC  ETHER unknown 0     --        ldg1-vsw1/vnet1_ldg1
ldoms-vsw0.vport1 VNIC  ETHER unknown 0     --        ldg1-vsw1/vnet2_ldg3
ldoms-vsw0.vport0 VNIC  ETHER unknown 0     --        ldg1-vsw1/vnet2_ldg2
```

EXEMPLE 39 Liste détaillée des informations de configuration des périphériques réseau

L'exemple suivant présente une liste détaillée des périphériques réseau pour le domaine `ldg1` à l'aide de la commande `ldm list-netdev -l`.

```
primary# ldm list-netdev -l ldg1

-------------------------------------------------------------------
DOMAIN
ldg1

NAME               CLASS    MEDIA    STATE   SPEED OVER LOC
----               -----    -----    -----   ----- ---- ---
net0               VNET     ETHER    up      0     --   primary-vsw0/vnet0_ldg1
    [/virtual-devices@100/channel-devices@200/network@0]
    MTU       : 1500 [1500-1500]
```

```
    IPADDR    : 10.129.241.200/255.255.255.0
    MAC_ADDRS : 00:14:4f:fb:9c:df

net3              PHYS     ETHER    up      10000 --   /SYS/MB/RISER1/PCIE4
    [/pci@400/pci@1/pci@0/pci@0/network@0]
    MTU       : 1500 [576-15500]
    MAC_ADDRS : a0:36:9f:0a:c5:d2

net4              VSW      ETHER    up      10000 --   ldg1-vsw1
    [/virtual-devices@100/channel-devices@200/virtual-network-switch@0]
    MTU       : 1500 [1500-1500]
    IPADDR    : 192.168.1.2/255.255.255.0
    MAC_ADDRS : 00:14:4f:fb:61:6e

net1              PHYS     ETHER    up      10000 --   /SYS/MB/RISER1/PCIE4
    [/pci@400/pci@1/pci@0/pci@0/network@0,1]
    MTU       : 1500 [576-15500]
    MAC_ADDRS : a0:36:9f:0a:c5:d2

net5              VNET     ETHER    up      0     --   ldg1-vsw1/vnet1_ldg1
    [/virtual-devices@100/channel-devices@200/network@1]
    MTU       : 1500 [1500-1500]
    IPADDR    : 0.0.0.0  /255.0.0.0
              : fe80::214:4fff:fef8:5062/ffc0::
    MAC_ADDRS : 00:14:4f:f8:50:62

net6              VNET     ETHER    up      0     --   ldg1-vsw1/vnet2_ldg1
    [/virtual-devices@100/channel-devices@200/network@2]
    MTU       : 1500 [1500-1500]
    IPADDR    : 0.0.0.0  /255.0.0.0
              : fe80::214:4fff:fef8:af92/ffc0::
    MAC_ADDRS : 00:14:4f:f8:af:92

aggr2             AGGR     ETHER    unknown  0    net1,net3 --
    MODE      : TRUNK
    POLICY    : L2,L3
    LACP_MODE : ACTIVE
    MEMBER    : net1 [PORTSTATE = attached]
    MEMBER    : net3 [PORTSTATE = attached]
    MAC_ADDRS : a0:36:9f:0a:c5:d2

ldoms-vsw0.vport3  VNIC     ETHER    unknown  0    -- ldg1-vsw1/vnet2_ldg1
    MTU       : 1500 [576-1500]
    MAC_ADDRS : 00:14:4f:f8:af:92

ldoms-vsw0.vport2  VNIC     ETHER    unknown  0    -- ldg1-vsw1/vnet1_ldg1
    MTU       : 1500 [576-1500]
    MAC_ADDRS : 00:14:4f:f8:50:62

ldoms-vsw0.vport1  VNIC     ETHER    unknown  0    -- ldg1-vsw1/vnet2_ldg3
    MTU       : 1500 [576-1500]
    MAC_ADDRS : 00:14:4f:f9:d3:88

ldoms-vsw0.vport0  VNIC     ETHER    unknown  0    -- ldg1-vsw1/vnet2_ldg2
    MTU       : 1500 [576-1500]
    MAC_ADDRS : 00:14:4f:fa:47:f4
              : 00:14:4f:f9:65:b5
              : 00:14:4f:f9:60:3f
```

EXEMPLE 40 Liste des statistiques des périphériques réseau

La commande `ldm list-netstat` affiche les statistiques réseau pour un ou plusieurs domaines du système.

L'exemple suivant présente les statistiques réseau par défaut pour tous les domaines du système.

```
primary# ldm list-netstat

DOMAIN
primary

NAME               IPACKETS     RBYTES       OPACKETS     OBYTES
----               --------     ------       --------     ------
net3               0            0            0            0
net0               2.72M        778.27M      76.32K       6.01M
net4               2.72M        778.27M      76.32K       6.01M
net6               2            140          1.30K        18.17K
net7               0            0            0            0
net2               0            0            0            0
net1               0            0            0            0
aggr1              0            0            0            0
ldoms-vsw0.vport0  935.40K      74.59M       13.15K       984.43K
ldoms-vsw0.vport1  933.26K      74.37M       11.42K       745.15K
ldoms-vsw0.vport2  933.24K      74.37M       11.46K       747.66K
ldoms-vsw1.vport1  202.26K      17.99M       179.75K      15.69M
ldoms-vsw1.vport0  202.37K      18.00M       189.00K      16.24M
------------------------------------------------------------------------
DOMAIN
ldg1

NAME               IPACKETS     RBYTES       OPACKETS     OBYTES
----               --------     ------       --------     ------
net0               5.19K        421.57K      68           4.70K
net3               0            0            2.07K        256.93K
net4               0            0            4.37K        560.17K
net1               0            0            2.29K        303.24K
net5               149          31.19K       78           17.00K
net6               147          30.51K       78           17.29K
aggr2              0            0            0            0
ldoms-vsw0.vport3  162          31.69K       52           14.11K
ldoms-vsw0.vport2  163          31.74K       51           13.76K
ldoms-vsw0.vport1  176          42.99K       25           1.50K
ldoms-vsw0.vport0  158          40.19K       45           4.42K
------------------------------------------------------------------------
DOMAIN
ldg2

NAME               IPACKETS     RBYTES       OPACKETS     OBYTES
----               --------     ------       --------     ------
net0               5.17K        418.90K      71           4.88K
net1               2.70K        201.67K      2.63K        187.01K
net2               132          36.40K       1.51K        95.07K
------------------------------------------------------------------------
DOMAIN
ldg3

NAME               IPACKETS     RBYTES       OPACKETS     OBYTES
----               --------     ------       --------     ------
net0               5.16K        417.43K      72           4.90K
```

```
net1              2.80K       206.12K      2.67K       190.36K
net2              118         35.00K       1.46K       87.78K
```

Contrôle de la quantité de bande passante de réseau physique consommée par un périphérique réseau virtuel

La fonction de contrôle des ressources de bande passante vous permet de limite la bande passante de réseau physique consommée par un périphérique réseau virtuel. Cette fonction est prise en charge sur un domaine de service qui exécute au moins le SE Oracle Solaris 11 et qui est configuré avec un commutateur virtuel. Les domaines de service Oracle Solaris 10 ignorent les paramètres de bande passante de réseau de manière silencieuse. Cette fonction s'assure qu'un domaine invité ne prend pas tout l'espace de bande passante de réseau physique disponible et n'en laisse pas pour les autres domaines.

Utilisez les commandes `ldm add-vnet` et `ldm set-vnet` pour définir la limite de bande passante en fournissant une valeur pour la propriété `maxbw`. Utilisez les commandes `ldm list-bindings` ou `ldm list-domain -o network` pour afficher la valeur de propriété `maxbw` d'un périphérique réseau virtuel existant. La limite minimum de bande passante est de 10 Mbits.

Restrictions de la bande passante réseau

Le contrôle des ressources de bande passante s'applique uniquement au trafic qui passe par le commutateur virtuel. Ainsi, le trafic inter-vnet n'est pas soumis à cette limite. Si vous n'avez pas configuré de périphérique de backend physique, vous pouvez ignorer le contrôle des ressources de bande passante.

La limite minimum de bande passante prise en charge dépend de la pile réseau d'Oracle Solaris dans le domaine de service. La limite de bande passante peut être configurée avec n'importe quelle valeur élevée souhaitée. Il n'y a pas de limite supérieure. La limite de bande passante assure uniquement que la bande passante ne dépasse pas la valeur configurée. Vous pouvez ainsi configurer une limite de bande passante avec une valeur supérieure à la vitesse de liaison du périphérique réseau physique attribué au commutateur virtuel.

Paramétrage de la limite de bande passante réseau

Utilisez la commande `ldm add-vnet` pour créer un périphérique réseau virtuel et définir la limite de bande passante en fournissant une valeur pour la propriété `maxbw`.

```
primary# ldm add-vnet maxbw=limit if-name vswitch-name domain-name
```

Utilisez la commande `ldm set-vnet` pour indiquer la limite de bande passante d'un périphérique réseau virtuel existant.

```
primary# ldm set-vnet maxbw=limit if-name domain-name
```

Vous pouvez également effacer la limite de bande passante en indiquant une valeur vide pour la propriété `maxbw` :

```
primary# ldm set-vnet maxbw= if-name domain-name
```

Les exemples suivants indiquent comment utiliser la commande `ldm` pour définir une limite de bande passante. La bande passante est indiquée comme étant un nombre entier avec une unité. `M` est l'unité pour les mégabits par seconde et `G` l'unité pour les gigabits par seconde. Par défaut l'unité est en mégabits par seconde.

EXEMPLE 41 Paramétrage de la limite de bande passante lors de la création d'un périphérique réseau virtuel

La commande suivante crée un périphérique réseau virtuel (`vnet0`) qui a une limite de bande passante de 100 Mbits.

```
primary# ldm add-vnet maxbw=100M vnet0 primary-vsw0 ldg1
```

La commande suivante envoie un message d'erreur si vous essayez de définir une limite de bande passante inférieure à la valeur minimum de 10 Mbits.

```
primary# ldm add-vnet maxbw=1M vnet0 primary-vsw0 ldg1
```

EXEMPLE 42 Paramétrage de la limite de bande passante sur un périphérique réseau virtuel existant

Les commandes suivantes définissent la limite de bande passante sur 200 Mbits sur le périphérique `vnet0` existant.

Selon le modèle de trafic du réseau en temps réel, il est possible que la quantité de bande passante n'atteigne pas la limite spécifiée de 200 Mbits. Par exemple, la bande passante peut être de 95 Mbits, ce qui ne dépasse pas la limite de 200 Mbits.

```
primary# ldm set-vnet maxbw=200M vnet0 ldg1
```

La commande suivante définit la limite de bande passante sur 2 Gbits sur le périphérique `vnet0` existant.

Comme il n'existe pas de limite supérieure de bande passante dans la couche MAC, vous pouvez définir la limite sur 2 Gbits même si la vitesse de réseau physique sous-jacente est inférieure à 2 Gbits. Dans ce cas-là, il n'y a pas d'effet pour la limite de bande passante.

```
primary# ldm set-vnet maxbw=2G vnet0 ldg1
```

EXEMPLE 43 Suppression de la limite de bande passante sur un périphérique réseau virtuel existant

La commande suivante efface la limite de bande passante sur le périphérique réseau virtuel spécifié (`vnet0`). En effaçant cette valeur, le périphérique réseau virtuel utilise la quantité maximum de bande passante disponible, fournie par le périphérique physique sous-jacent.

```
primary# ldm set-vnet maxbw= vnet0 ldg1
```

EXEMPLE 44 Affichage de la limite de bande passante d'un périphérique réseau virtuel existant

La commande `ldm list-bindings` affiche la valeur de la propriété `maxbw` pour le périphérique réseau virtuel spécifié, le cas échéant.

La commande suivante indique que le périphérique réseau virtuel `vnet3` a une limite de bande passante de 15 Mbits/s. Si aucune bande passante n'est définie, le champ `MAXBW` est vide.

```
primary# ldm ls-bindings -e -o network ldg3
NAME
ldg3

MAC
    00:14:4f:f8:5b:12

NETWORK
    NAME         SERVICE                MACADDRESS PVID|PVLAN|VIDs
    ----         -------                ---------- ---------------
    vnet3        primary-vsw0@primary   00:14:4f:fa:ba:b9 1|--|--
            DEVICE     :network@0       ID   :0
            LINKPROP   :--              MTU  :1500
            MAXBW      :15M             MODE :--
            CUSTOM     :disable
            PRIORITY   :--              COS  :--
            PROTECTION :--

        PEER                   MACADDRESS          PVID|PVLAN|VIDs
        ----                   ----------          ---------------
        primary-vsw0@primary   00:14:4f:f9:08:28   1|--|--
            LINKPROP   :--              MTU  :1500
            MAXBW      :--              LDC  :0x0
            MODE       :--
```

Vous pouvez également utiliser la commande `dladm show-linkprop` pour afficher la valeur de propriété `maxbw` de la manière suivante :

```
# dladm show-linkprop -p maxbw
LINK              PROPERTY PERM VALUE  EFFECTIVE DEFAULT POSSIBLE
...
ldoms-vsw0.vport0 maxbw    rw   15     15        --      --
```

Identificateur de périphérique virtuel et nom d'interface réseau

Si vous ajoutez un commutateur virtuel ou un périphérique réseau virtuel à un domaine, vous pouvez indiquer le numéro de périphérique en définissant la propriété `id`.

```
primary# ldm add-vsw [id=switch-id] vswitch-name domain-name
primary# ldm add-vnet [id=network-id] if-name vswitch-name domain-name
```

Chaque commutateur virtuel ou périphérique réseau virtuel d'un domaine a un numéro de périphérique unique qui est assigné lorsque le domaine est lié. Si un commutateur virtuel ou un périphérique réseau virtuel a été ajouté avec un numéro de périphérique explicite (en définissant la propriété `id`), le numéro de périphérique défini est utilisé. Sinon, le système assigne automatiquement le numéro de périphérique le plus petit disponible. Dans ce cas, le numéro de périphérique assigné dépend de la manière dont le commutateur virtuel ou les périphériques réseau virtuels ont été ajoutés au système. Le numéro de périphérique éventuellement assigné à un commutateur virtuel ou à un périphérique réseau virtuel est visible dans la sortie de la commande `ldm list-bindings` lorsqu'un domaine est lié.

L'exemple suivant montre que le domaine `primary` a une commutateur virtuel, `primary-vsw0`. Ce commutateur virtuel a un numéro de périphérique de `0` (`switch@0`).

```
primary# ldm list-bindings primary
...
VSW
    NAME         MAC               NET-DEV DEVICE   DEFAULT-VLAN-ID PVID VID MTU MODE
    primary-vsw0 00:14:4f:fb:54:f2 net0    switch@0 1               1    5,6 1500
...
```

L'exemple suivant montre que le domaine `ldg1` a deux périphériques réseau virtuels : `vnet` et `vnet1`. Le périphérique `vnet` a un numéro de périphérique de `0` (`network@0`) et le périphérique `vnet1` a un numéro de périphérique de `1` (`network@1`).

```
primary# ldm list-bindings ldg1
...
NETWORK
    NAME  SERVICE              DEVICE    MAC               MODE   PVID VID MTU
    vnet  primary-vsw0@primary network@0 00:14:4f:fb:e0:4b hybrid 1        1500
    ...
    vnet1 primary-vsw0@primary network@1 00:14:4f:f8:e1:ea        1        1500
```

...

De même, si un domaine avec un périphérique réseau virtuel exécute le SE Oracle Solaris, le périphérique réseau virtuel a une interface réseau, `vnet`*N*. Cependant, le numéro d'interface réseau du périphérique réseau virtuel, *N*, n'est pas nécessairement le même que le numéro de périphérique réseau virtuel, *n*.

Remarque - Sur les systèmes Oracle Solaris 11, des noms de lien génériques de la forme `net`*n* sont assignés aussi bien à `vsw`*n* qu'à `vnet`*n*. Utilisez la commande `dladm show-phys` pour identifier les noms `net`*n*correspondant aux périphériques `vsw`*n* et `vnet`*n*.

Attention - LE SE Oracle Solaris préserve le mappage entre le nom de l'interface réseau et un commutateur virtuel ou un périphérique réseau virtuel en fonction du numéro de périphérique. Si un numéro de périphérique n'est assigné pas explicitement à un commutateur virtuel ou à un périphérique réseau virtuel, son numéro de périphérique peut changer lorsque le domaine est dissocié, puis est réassocié ultérieurement. Dans ce cas, le nom de l'interface réseau assigné par le SE s'exécutant dans le domaine peut également changer et rendre la configuration système existante inutilisable. Cela peut se produire, par exemple, lorsque l'interface d'un commutateur virtuel ou d'un réseau virtuel est supprimée de la configuration du domaine.

Vous ne pouvez pas utiliser les commandes `ldm list-*` pour déterminer directement le nom de l'interface réseau du SE Oracle Solaris qui correspond à un commutateur virtuel ou à un périphérique réseau virtuel. Cependant, vous pouvez obtenir ces informations à l'aide d'une association de la sortie de la commande `ldm list -l` et des entrées sous `/devices` du SE Oracle Solaris.

Recherche du nom de l'interface réseau du Oracle Solaris 11

Sur les systèmes Oracle Solaris 11, vous pouvez utiliser la commande `ldm list-netdev` pour rechercher les noms d'interface réseau du SE Oracle Solaris. Pour plus d'informations, reportez-vous à la page de manuel `ldm`(1M).

L'exemple suivant illustre les commandes `ldm list-netdev` et `ldm list -o network`. La commande `ldm list -o network` affiche les périphériques réseau virtuels dans le champ `NAME`. La sortie `ldm list-netdev` affiche le nom de l'interface du SE correspondant dans la colonne `NAME`.

```
primary# ldm list -o network ldg1
....
NETWORK
    NAME        SERVICE              ID DEVICE    MAC               MODE
    PVID VID MTU   MAXBW LINKPROP
```

```
    vnet0-ldg1  primary-vsw0@primary 0  network@0 00:14:4f:fa:eb:4e 1
             1500
    vnet1-ldg1  svcdom-vsw0@svcdom   1  network@1 00:14:4f:f8:53:45 4
             1500
           PVLAN :400,community

primary# ldm list-netdev ldg1
DOMAIN
ldg1

NAME CLASS MEDIA STATE   SPEED OVER  LOC
---- ----- ----- -----   ----- ----  ---
net0 VNET  ETHER up      0     vnet0 primary-vsw0/vnet0-ldg1
net1 VNET  ETHER up      0     vnet1 svcdom-vsw0/vnet1-ldg1
net2 VNET  ETHER unknown 0     vnet2 svcdom-vsw1/vnet2-ldg1
```

Pour vérifier que la sortie `ldm list-netdev` est correcte, exécutez les commandes `dladm show-phys` et `dladm show-linkprop -p mac-address` pour `ldg1` :

```
ldg1# dladm show-phys
LINK     MEDIA     STATE      SPEED  DUPLEX         DEVICE
net0     Ethernet  up         0      unknown        vnet0
net1     Ethernet  up         0      unknown        vnet1
net2     Ethernet  unknown    0      unknown        vnet2

ldg1# dladm show-linkprop -p mac-address
LINK PROPERTY    PERM VALUE             EFFECTIVE        DEFAULT          POSSIBLE
net0 mac-address rw   0:14:4f:fa:eb:4e 0:14:4f:fa:eb:4e 0:14:4f:fa:eb:4e --
net1 mac-address rw   0:14:4f:f8:53:45 0:14:4f:f8:53:45 0:14:4f:f8:53:45 --
```

▼ Procédure d'identification du nom de l'interface réseau du SE Oracle Solaris

Cette procédure indique comment trouver le nom de l'interface réseau d'SE Oracle Solaris dans `ldg1` qui correspond à `net-c`. Cet exemple montre également les différences si vous recherchez un nom d'interface réseau d'un commutateur virtuel au lieu d'un périphérique réseau virtuel. Dans cet exemple de procédure, le domaine invité `ldg1` contient deux périphériques de réseau virtuel, `net-a` et `net-c`.

1. Utilisez la commande `ldm` pour rechercher le numéro du périphérique réseau virtuel pour `net-c`.

```
primary# ldm list -l ldg1
...
NETWORK
NAME         SERVICE                     DEVICE       MAC
net-a        primary-vsw0@primary        network@0    00:14:4f:f8:91:4f
net-c        primary-vsw0@primary        network@2    00:14:4f:f8:dd:68
...
```

Le numéro de périphérique réseau virtuel pour `net-c` est `2` (`network@2`).

Pour déterminer le nom d'interface réseau d'un commutateur virtuel, recherchez le numéro du périphérique de commutateur virtuel, *n* en tant que `switch@`*n*.

2. **Pour trouver l'interface réseau correspondante sur `ldg1`, connectez-vous à `ldg1` et recherchez l'entrée de ce numéro de périphérique sous `/devices`.**

```
ldg1# uname -n
ldg1
ldg1# find /devices/virtual-devices@100 -type c -name network@2\*
/devices/virtual-devices@100/channel-devices@200/network@2:vnet1
```

Le nom d'interface réseau est la partie de l'entrée après les deux-points, c'est-à-dire `vnet1`.

Pour déterminer le nom d'interface réseau d'un commutateur virtuel, remplacez l'argument de l'option-`name` par `virtual-network-switch@`*`n`*`\*`. Recherchez ensuite l'interface réseau avec le nom `vsw`*N*.

3. **Vérifiez que `vnet1` possède l'adresse MAC `00:14:4f:f8:dd:68` comme indiqué dans la sortie de la commande `ldm list -l` pour `net-c` à l'étape 1.**

- **Système d'exploitation Oracle Solaris 11.**

  a. **Déterminez le nom de l'interface à spécifier pour `vnet1`.**

  ```
  ldg1# dladm show-phys |grep vnet1
  net2      Ethernet      up      0      unknown   vnet1
  ```

  b. **Déterminez l'adresse MAC de `net2`.**

  ```
  # dladm show-linkprop -p mac-address net2
  LINK PROPERTY    PERM VALUE             EFFECTIVE         DEFAULT POSSIBLE
  net2 mac-address rw   00:14:4f:f8:dd:68 00:14:4f:f8:dd:68 --      --
  ```

  Cet exemple d'adresse MAC correspond à la sortie de la commande `ldm list -l` pour `net-c` dans l'étape 1.

- **Système d'exploitation Oracle Solaris 10.**

```
ldg1# ifconfig vnet1
vnet1: flags=1000842<BROADCAST,RUNNING,MULTICAST,IPv4> mtu 1500 index 3
          inet 0.0.0.0 netmask 0
          ether 0:14:4f:f8:dd:68
```

Assignation automatique et manuelle des adresses MAC

Vous devez disposer du nombre suffisant d'adresses de contrôle d'accès aux médias (MAC) pour assigner le numéro des domaines logiques, des commutateurs virtuels et des réseaux virtuels que vous allez utiliser. Logical Domains Manager peut assigner automatiquement les adresses MAC à un domaine logique, un réseau virtuel, un commutateur virtuel ; ou vous pouvez assigner manuellement des adresses MAC à partir de votre propre pool d'adresses MAC

assignées. Les sous-commandes `ldm` définissant des adresses MAC sont`add-domain`, `add-vsw`, `set-vsw`, `add-vnet` et `set-vnet`. Si vous n'indiquez aucune adresse MAC dans ces sous-commandes, Logical Domains Manager en assigne une automatiquement.

L'avantage de faire assigner les adresses MAC par Logical Domains Manager est qu'il utilise le bloc d'adresses MAC dédié à une utilisation avec les domaines logiques. Logical Domains Manager détecte et empêche également les collisions d'adresse MAC avec d'autres instances de Logical Domains Manager du même sous-réseau. Ce comportement vous libère de la gestion manuelle de votre pool d'adresses MAC.

L'assignation d'adresse MAC se produit dès qu'un domaine logique est créé ou qu'un périphérique réseau est configuré dans un domaine. Par ailleurs, l'assignation est persistance jusqu'à ce que le périphérique, ou le domaine logique lui-même, soit supprimé.

Plage d'adresses MAC assignées aux domaines

Le bloc de 512 K d'adresses MAC suivant est assigné aux domaines :

```
00:14:4F:F8:00:00 ~ 00:14:4F:FF:FF:FF
```

Les adresses 256K inférieures sont utilisées par Logical Domains Manager pour l'allocation automatique d'adresse MAC et vous ne pouvez pas demander manuellement une adresse dans cette plage :

```
00:14:4F:F8:00:00 - 00:14:4F:FB:FF:FF
```

Vous pouvez utiliser la moitié supérieure de cette plage pour l'allocation manuelle d'adresse MAC :

```
00:14:4F:FC:00:00 - 00:14:4F:FF:FF:FF
```

Remarque - Dans Oracle Solaris 11, l'allocation d'adresses MAC pour les VNIC utilise des adresses en dehors de ces plages.

Algorithme d'assignation automatique

Lorsque vous n'indiquez aucune adresse MAC lors de la création d'un domaine logique ou d'un périphérique réseau, Logical Domains Manager alloue automatiquement et assigne une adresse MAC à ce domaine logique ou périphérique réseau.

Pour obtenir cette adresse MAC, Logical Domains Manager tente itérativement de sélectionner une adresse, puis contrôle les collisions potentielles. L'adresse MAC est sélectionnée de manière aléatoire dans la plage 256K d'adresses mise de côté à cet effet. L'adresse MAC est sélectionnée de manière aléatoire pour réduire le risque de sélectionner une adresse MAC en doublon comme candidat.

L'adresse sélectionnée est contrôlée par rapport aux autres Logical Domains Managers sur les autres systèmes pour éviter aux adresses MAC en doublon d'être assignées. L'algorithme utilisé est décrit dans la section "Détection des adresses MAC en doublon" à la page 272. Si l'adresse est déjà assignée, Logical Domains Manager itère en choisissant une autre adresse et en vérifiant à nouveau les collisions. Ce processus continue jusqu'à ce qu'une adresse MAC qui n'est pas encore allouée soit trouvée ou jusqu'à ce qu'une limite de temps de 30 secondes se soit écoulée. Si la limite de temps est atteinte, la création du périphérique échoue et un message d'erreur semblable au suivant s'affiche.

```
Automatic MAC allocation failed.  Please set the vnet MAC address manually.
```

Détection des adresses MAC en doublon

Pour éviter qu'une adresse MAC identique ne soit allouée à différents périphériques, Logical Domains Manager effectue une vérification auprès des autres Logical Domains Managers sur les autres systèmes en envoyant un message multidiffusion via l'interface réseau par défaut du domaine de contrôle, en incluant l'adresse que Logical Domains Manager souhaite assigner au périphérique. Logical Domains Manager tentant d'assigner l'adresse MAC attend une réponse pendant une seconde. Si un périphérique différent sur un autre système compatible Oracle VM Server for SPARC a déjà été associé à cette adresse MAC, le gestionnaire Logical Domains Manager de ce système renvoie une réponse contenant l'adresse MAC en question. Si le gestionnaire Logical Domains Manager demandeur reçoit une réponse, il sait que l'adresse MAC choisie a déjà été allouée. Il en choisit une autre, puis recommence.

Par défaut, ces messages de multidiffusion sont uniquement envoyés aux autres gestionnaires sur le même sous-réseau. La valeur par défaut du TTL (durée de vie) est `1`. Le TTL peut être configuré à l'aide de la propriété SMF (Utilitaires de gestion des services)`ldmd/hops`.

Chaque Logical Domains Manager est responsable des éléments suivants :

- De l'écoute des messages multidiffusion
- Du suivi des adresses MAC assignées à ses domaines
- De la recherche de doublons
- De la réponse afin que les doublons ne surviennent pas

Si Logical Domains Manager sur un système est arrêté pour quelque raison que ce soit, les adresses MAC en doublon peuvent se produire pendant l'arrêt de Logical Domains Manager.

L'allocation MAC automatique se produit au moment où le domaine logique ou le périphérique réseau est créé et persiste jusqu'à ce que le périphérique ou le domaine logique soit supprimé.

Remarque - Une vérification de détection des adresses MAC en doublon est effectuée lorsque le domaine logique ou le périphérique réseau est créé et que le domaine logique est démarré.

Utilisation des adaptateurs réseau avec les domaines exécutant Oracle Solaris 10

Dans un environnement de domaines logiques Oracle Solaris 10, le service de commutateur virtuel s'exécutant sur un domaine de service peut directement interagir avec les adaptateurs réseau compatibles GLDv3. Bien que les adaptateurs réseau non compatibles GLDv3 puissent être utilisés dans ces systèmes, le commutateur virtuel ne peut pas communiquer avec eux directement. Reportez-vous à la section "Configuration d'un commutateur virtuel et du domaine de service pour NAT et le routage" à la page 274 pour plus d'informations sur l'utilisation des adaptateurs réseau non compatibles GLDv3.

Remarque - La question de la compatibilité GLDv3 ne concerne pas les environnements Oracle Solaris 11.

Pour plus d'informations sur l'utilisation du groupement de liaisons, reportez-vous à la section "Utilisation du groupement de liaisons avec un commutateur virtuel" à la page 300.

▼ Procédure de détermination de la compatibilité GLDv3 d'un adaptateur réseau

Cette procédure s'applique uniquement aux domaines Oracle Solaris 10.

- **Déterminez si l'adaptateur réseau est compatible GLDv3.**

  Dans l'exemple suivant, `bge0` correspond au nom du périphérique réseau.

```
# dladm show-link bge0
bge0            type: non-vlan   mtu: 1500      device: bge0
```

  Le champ `type:` contient l'une des valeurs suivantes :

  - Les pilotes compatibles GLDv3 sont de type `non-vlan` ou `vlan`.
  - Les pilotes non compatibles GLDv3 sont de type `legacy`.

Configuration d'un commutateur virtuel et du domaine de service pour NAT et le routage

Dans le système d'exploitation Oracle Solaris 10, le commutateur virtuel (`vsw`) est un commutateur à 2 couches pouvant également être utilisé comme périphérique réseau dans le domaine de service. Le commutateur virtuel peut être configuré pour agir uniquement comme un commutateur entre les périphériques réseau virtuel de différents domaines logiques mais sans connectivité à un réseau en dehors de la boîte via un périphérique physique. Dans ce mode, la création de `vsw` en tant que périphérique réseau et l'activation du routage IP sur le domaine de service permet aux réseaux virtuels de communiquer en dehors de la boîte en utilisant le domaine de service comme routeur. Ce mode de fonctionnement est essentiel pour fournir une connectivité externe aux domaines lorsque l'adaptateur réseau physique n'est pas compatible GLDv3.

Les avantages de cette configuration sont :

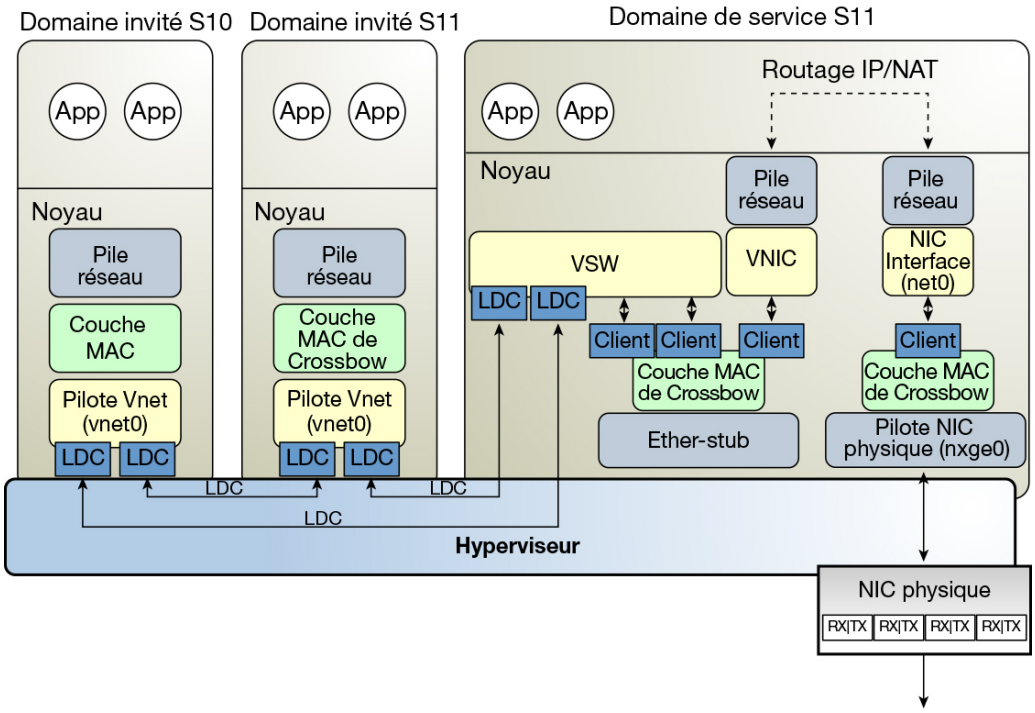
- Le commutateur virtuel ne doit pas utiliser un périphérique physique directement et peut fournir une connectivité externe même si le périphérique sous-jacent n'est pas compatible GLDv3.
- Cette configuration peut tirer parti des fonctionnalités de routage et de filtrage IP du SE Oracle Solaris.

Configuration de NAT sur un système Oracle Solaris 11

Les fonctions de virtualisation réseau d'Oracle Solaris 11 incluent `etherstub`, qui est un pseudopériphérique réseau. Ce périphérique fournit des fonctionnalités similaires à celles des périphériques réseau physiques, mais uniquement pour les communications confidentielles avec ses clients. Ce pseudopériphérique peut être utilisé en tant que périphérique backend réseau pour un commutateur virtuel assurant les communications confidentielles entre les réseaux virtuels. L'utilisation du périphérique `etherstub` en tant que périphérique backend permet également aux domaines invités de communiquer avec des VNIC sur le même périphérique `etherstub`. En utilisant le périphérique `etherstub` de cette manière, les domaines invités peuvent communiquer avec des zones dans le domaine de service. Utilisez la commande `dladm create-etherstub` pour créer un périphérique `etherstub`.

Le schéma suivant illustre l'utilisation des commutateurs virtuels, des périphériques `etherstub` et des VNIC pour configurer la traduction d'adresse réseau (NAT) dans un domaine de service.

FIGURE 15 Routage du réseau virtuel



Vous pouvez utiliser des routes permanentes. Pour plus d'informations, reportez-vous à la section "Troubleshooting Issues When Adding a Persistent Route" du manuel *Troubleshooting Network Administration Issues in Oracle Solaris 11.3* et "Creating Persistent (Static) Routes" du manuel *Configuring and Managing Network Components in Oracle Solaris 11.3*.

▼ Procédure de configuration d'un commutateur virtuel pour fournir une connectivité externe aux domaines (Oracle Solaris 11)

1. Créez un périphérique `etherstub` Oracle Solaris 11.

```
primary# dladm create-etherstub stub0
```

2. **Créez un commutateur virtuel utilisant `stub0` en tant que périphérique backend physique.**

```
primary# ldm add-vsw net-dev=stub0 primary-stub-vsw0 primary
```

3. **Créez une VNIC sur le périphérique `stub0`.**

```
primary# dladm create-vnic -l stub0 vnic0
```

4. **Configurez `vnic0` en tant qu'interface réseau.**

```
primary# ipadm create-ip vnic0
primary# ipadm create-addr -T static -a 192.168.100.1/24  vnic0/v4static
```

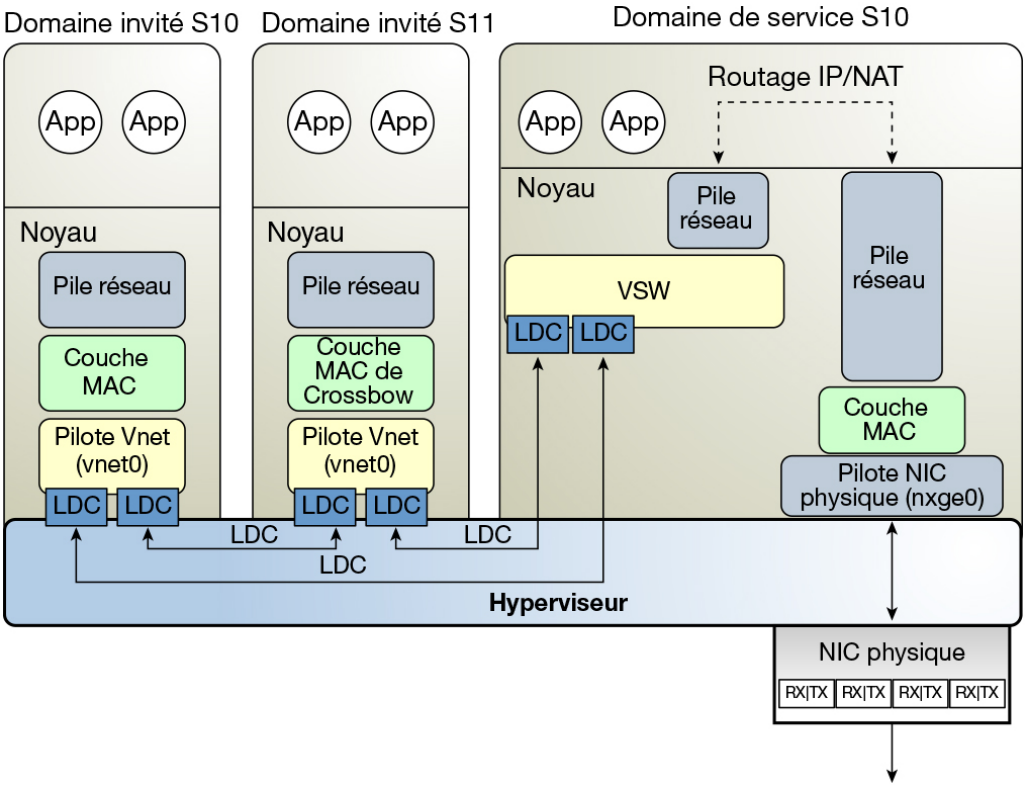
5. **Activez le transfert IPv4 et créez des règles NAT.**

   Reportez-vous à la section "Customizing IP Interface Properties and Addresses" du manuel *Configuring and Managing Network Components in Oracle Solaris 11.3* et "Packet Forwarding and Routing on IPv4 Networks" du manuel *Oracle Solaris Administration: IP Services*.

Configuration de NAT sur un système Oracle Solaris 10

Le schéma suivant illustre l'utilisation d'un commutateur virtuel pour la configuration de la translation d'adresse réseau (Network Translation Address, NAT) dans un domaine de service afin de fournir une connectivité externe aux domaines invités.

FIGURE 16 Routage du réseau virtuel



▼ Procédure de configuration d'un commutateur virtuel sur les services de domaine Oracle Solaris 10 pour fournir une connectivité externe aux domaines

1. **Créez un commutateur virtuel sans périphérique physique associé.**

   Si vous assignez une adresse, vérifiez que le commutateur virtuel a une adresse MAC unique.

   ```
   primary# ldm add-vsw [mac-addr=xx:xx:xx:xx:xx:xx] ldg1-vsw0 ldg1
   ```

2. **Créez le commutateur virtuel en tant que périphérique réseau sur le périphérique réseau physique utilisé par le domaine.**

Pour plus d'informations sur la création d'un commutateur virtuel, reportez-vous à la section "Procédure de configuration du commutateur virtuel en tant qu'interface primary" à la page 43.

3. **Configurez le périphérique du commutateur virtuel pour le DHCP, si nécessaire.**

   Reportez-vous à la section "Procédure de configuration du commutateur virtuel en tant qu'interface primary" à la page 43 pour plus d'informations sur la configuration du périphérique du commutateur virtuel pour le DHCP.

4. **Créez le fichier `/etc/dhcp.vsw`, si nécessaire.**

5. **Configurez le routage IP dans le domaine de service et définissez les tables de routage requises dans tous les domaines.**

   Pour plus d'informations sur le routage IP, reportez-vous à la section "Packet Forwarding and Routing on IPv4 Networks" du manuel *Oracle Solaris Administration: IP Services*.

Configuration d'IPMP dans un environnement Oracle VM Server for SPARC

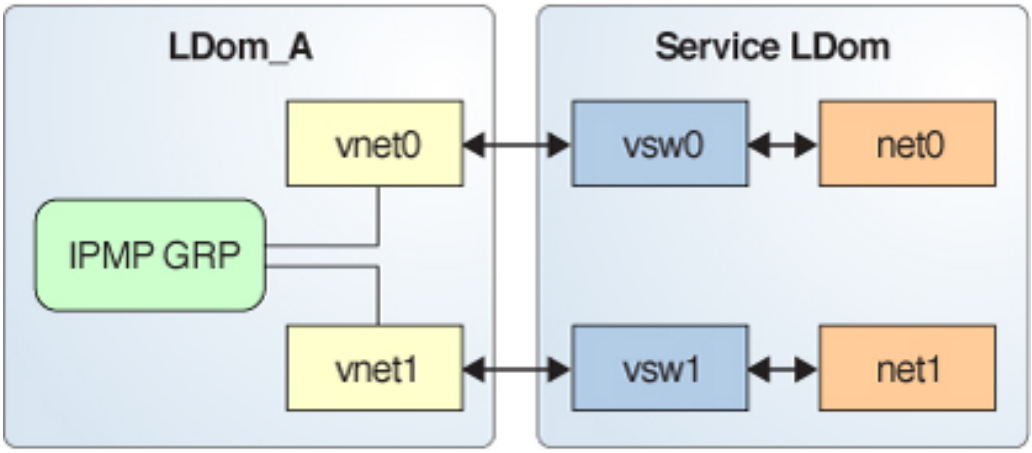
Le logiciel Oracle VM Server for SPARC prend en charge l'IPMP (IP network multipathing, multipathing sur réseau IP) basé sur liaison avec les périphériques réseau virtuels. Lors de la configuration d'un groupe IPMP avec des périphériques de réseau virtuel, configurez le groupe pour utiliser la détection de liaison. Si vous utilisez des versions antérieures du logiciel Oracle VM Server for SPARC (Logical Domains), vous ne pouvez configurer que la détection par sonde avec les périphériques de réseau virtuel.

Configuration des périphériques de réseau virtuel dans un groupe IPMP dans un domaine Oracle Solaris 11

La Figure 17, "Deux réseaux virtuels connectés à des instances de commutateur virtuel distinctes (Oracle Solaris 11)" illustre deux réseaux virtuels (`vnet0` et `vnet1`) connectés à des instances de commutateur virtuel distinctes (`vsw0` et `vsw1`) dans le domaine de service qui, à leur tour, utilisent deux interfaces physiques différentes. Les interfaces physiques sont `net0` et `net1` dans le domaine de service Oracle Solaris 11.

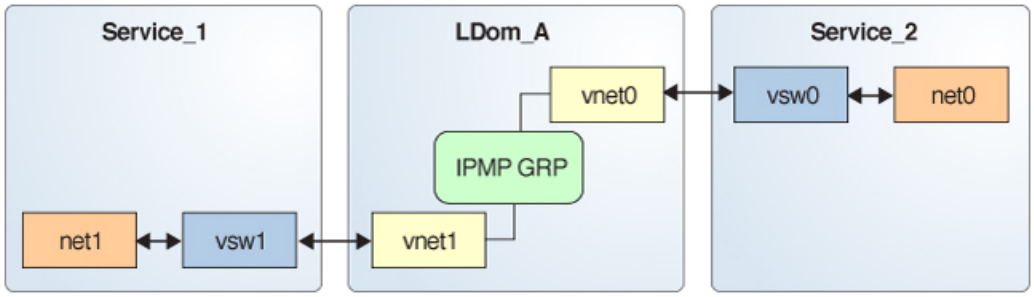
En cas de défaillance d'une liaison physique dans le domaine de service, le périphérique du commutateur virtuel qui est lié à ce périphérique physique détecte la défaillance. Ensuite, le périphérique du commutateur virtuel propage la panne au périphérique réseau virtuel correspondant lié à ce commutateur virtuel. Le périphérique réseau virtuel envoie une notification de cet événement de liaison à la couche IP dans l'invité `LDom_A`, ce qui aboutit à un basculement sur l'autre périphérique réseau virtuel dans le groupe IPMP.

FIGURE 17 Deux réseaux virtuels connectés à des instances de commutateur virtuel distinctes (Oracle Solaris 11)



La Figure 18, "Connexion de chaque périphérique réseau virtuel à des domaines de service différents (Oracle Solaris 11)" montre qu'il est possible d'obtenir une meilleure fiabilité dans le domaine logique en connectant chaque périphérique réseau virtuel (`vnet0` et `vnet1`) à des instances de commutateur virtuel dans des domaines de service différents. Dans ce cas, en plus de la panne du réseau physique, `LDom_A` peut détecter une panne du réseau virtuel et déclencher un basculement suite à une panne ou un arrêt du domaine de service.

FIGURE 18 Connexion de chaque périphérique réseau virtuel à des domaines de service différents (Oracle Solaris 11)



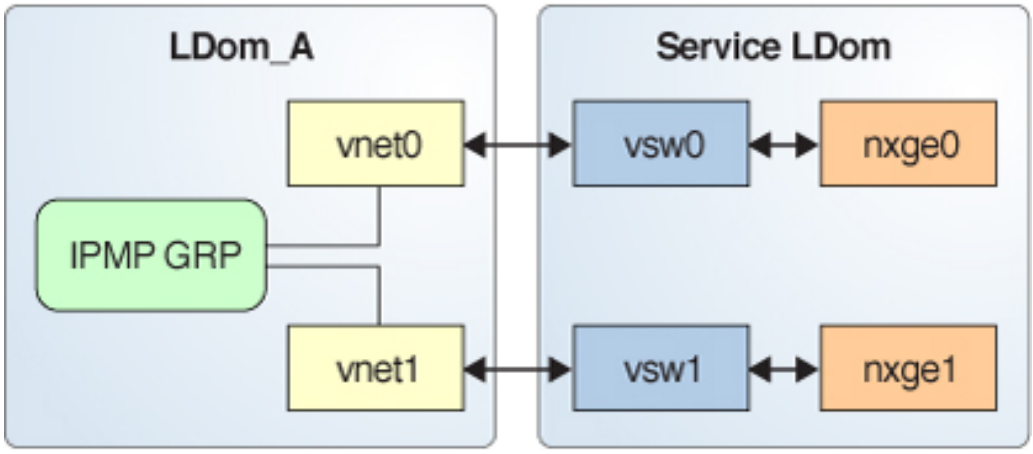
Pour plus d'informations, reportez-vous au manuel Mise en place d'un réseau Oracle Solaris dans la bibliothèque d'informations Oracle Solaris 11.3.

Configuration des périphériques de réseau virtuel dans un groupe IPMP dans un domaine Oracle Solaris 10

La Figure 19, "Deux réseaux virtuels connectés à des instances de commutateur virtuel distinctes (Oracle Solaris 10)" illustre deux réseaux virtuels (`vnet0` et `vnet1`) connectés à des instances de commutateur virtuel distinctes (`vsw0` et `vsw1`) dans le domaine de service qui, à leur tour, utilisent deux interfaces physiques différentes. Les interfaces physiques sont `nxge0` et `nxge1` dans le domaine de service Oracle Solaris 10.

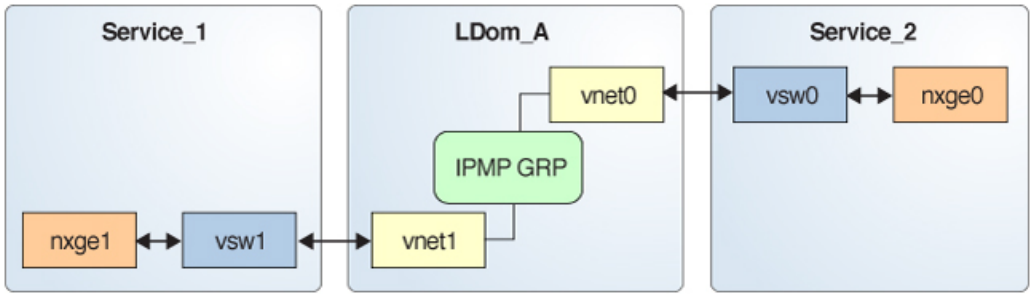
En cas de défaillance d'une liaison physique dans le domaine de service, le périphérique du commutateur virtuel qui est lié à ce périphérique physique détecte la défaillance. Ensuite, le périphérique du commutateur virtuel propage la panne au périphérique réseau virtuel correspondant lié à ce commutateur virtuel. Le périphérique réseau virtuel envoie une notification de cet événement de liaison à la couche IP dans l'invité `LDom_A`, ce qui aboutit à un basculement sur l'autre périphérique réseau virtuel dans le groupe IPMP.

FIGURE 19 Deux réseaux virtuels connectés à des instances de commutateur virtuel distinctes (Oracle Solaris 10)



La Figure 20, "Connexion de chaque périphérique réseau virtuel à des domaines de service différents (Oracle Solaris 10)" montre qu'il est possible d'obtenir une meilleure fiabilité dans le domaine logique en connectant chaque périphérique réseau virtuel (`vnet0` et `vnet1`) à des instances de commutateur virtuel dans des domaines de service différents. Dans ce cas, en plus de la panne du réseau physique, `LDom_A` peut détecter une panne du réseau virtuel et déclencher un basculement suite à une panne ou un arrêt du domaine de service.

FIGURE 20 Connexion de chaque périphérique réseau virtuel à des domaines de service différents (Oracle Solaris 10)



Pour plus d'informations, reportez-vous au manuel *Oracle Solaris Administration: IP Services*.

Configuration et utilisation d'IPMP dans le domaine de service

Sur un système Oracle Solaris 11, vous pouvez configurer IPMP dans un domaine de service en configurant des interfaces physiques dans un groupe de la même façon que sur un système qui n'a pas de réseau virtuel ou domaines. Sur un système Oracle Solaris 10, vous pouvez configurer IPMP dans le domaine de service en configurant des interfaces de commutateur virtuel dans un groupe. Les Figure 21, "Configuration de deux NIC physiques en tant que membres d'un groupe IPMP (Oracle Solaris 11)" et Figure 22, "Configuration de deux interfaces de commutateur virtuel en tant que membres d'un groupe IPMP (Oracle Solaris 10)" illustrent deux instances de commutateur virtuel (`vsw0` et `vsw1`) qui sont liées à deux périphériques physiques différents. Les deux interfaces de commutateur virtuel peuvent ensuite être créées et configurées dans un groupe IPMP. En cas de panne d'une liaison physique, le périphérique du commutateur virtuel qui est lié à ce périphérique physique détecte la panne de liaison. Le périphérique du commutateur virtuel envoie une notification de cet événement de liaison à la couche IP du domaine de service, ce qui aboutit à un basculement sur l'autre périphérique de commutateur virtuel dans le groupe IPMP. Les deux interfaces physiques sont `net0` et `net1` dans Oracle Solaris 11 et `nxge0` et `nxge1` dans Oracle Solaris 10.

FIGURE 21 Configuration de deux NIC physiques en tant que membres d'un groupe IPMP (Oracle Solaris 11)

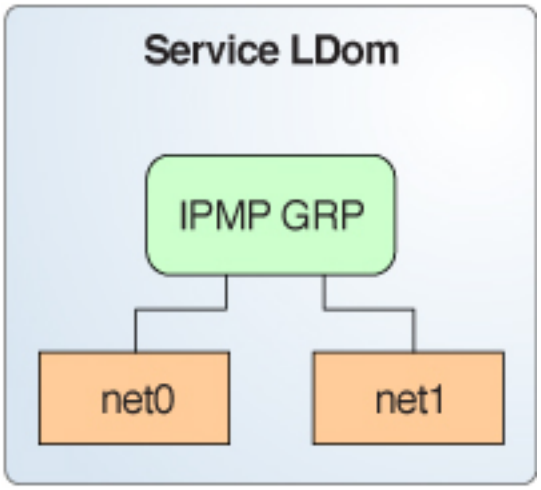
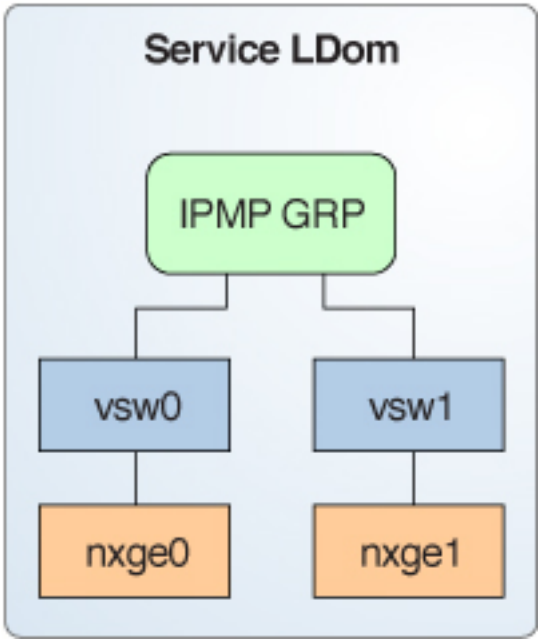


FIGURE 22 Configuration de deux interfaces de commutateur virtuel en tant que membres d'un groupe IPMP (Oracle Solaris 10)



Utilisation de l'IPMP basé sur liaison dans la mise en réseau virtuelle de Oracle VM Server for SPARC

Les périphériques de réseau virtuel et de commutateur virtuel prennent en charge les mises à jour de l'état de liaison dans la pile réseau. Par défaut, un périphérique réseau virtuel signale l'état de sa liaison virtuelle (son LDC au commutateur virtuel) et de sa liaison physique. Cette configuration est activée par défaut et ne nécessite pas d'autre étape de configuration de votre part.

Remarque - La propriété `linkprop` est définie sur `phys-state` par défaut, selon la prise en charge du périphérique de sauvegarde. Il n'est pas nécessaire d'effectuer les tâches de cette section sauf si vous avez désactivé la propriété `linkprop` manuellement et que vous tentez maintenant de définir la propriété à la valeur `phys-state`.

Vous pouvez utiliser les commandes d'administration réseau standard d'Oracle Solaris, notamment `dladm` et `ifconfig` pour vérifier l'état de la liaison. Par ailleurs, l'état de liaison est également consigné dans le fichier `/var/adm/messages`. Pour Oracle Solaris 10, reportez-vous aux pages de manuel `dladm`(1M) et `ifconfig`(1M). Pour Oracle Solaris 11, reportez-vous aux pages de manuel `dladm`(1M), `ipadm`(1M) et `ipmpstat`(1M).

Remarque - Vous pouvez exécuter les pilotes connaissant et ne connaissant pas l'état de liaison `vnet` et `vsw` simultanément sur un système Oracle VM Server for SPARC. Cependant, si vous avez l'intention de configurer un IPMP basé sur la liaison, vous devez installer un pilote connaissant l'état de liaison. Si vous avez l'intention d'activer les mises à jour de l'état de liaison physique, mettez à niveau les pilotes `vnet` et `vsw` vers SE Oracle Solaris 10 1/13 et exécutez au moins la version 1.3 de Logical Domains Manager.

▼ Procédure de configuration des mises à jour de l'état de liaison physique

Cette procédure indique comment activer les mises à jour de l'état de liaison physique pour les périphériques réseau virtuel.

Vous pouvez également activer les mises à jour de l'état de liaison physique pour un périphérique du réseau virtuel en suivant une procédure identique et en définissant l'option `linkprop=phys-state` dans les commandes `ldm add-vsw` et `ldm set-vsw`.

Remarque - Vous ne devez utiliser l'option `linkprop=phys-state` que si le périphérique de commutateur virtuel lui-même est créé en tant qu'interface. Si `linkprop=phys-state` est spécifié et que la liaison physique est interrompue, le périphérique du réseau virtuel signale son état de liaison comme interrompu, même si la connexion au commutateur virtuel fonctionne. Cette situation se produit parce que le SE Oracle Solaris ne fournit actuellement pas d'interface pour signaler deux états de liaison distincts, notamment l'état de liaison virtuelle et l'état de liaison physique.

1. **Connectez-vous en tant qu'administrateur.**
   Pour Oracle Solaris 11.3, reportez-vous au Chapitre 1, "About Using Rights to Control Users and Processes" du manuel *Securing Users and Processes in Oracle Solaris 11.3*.

2. **Activez les mises à jour de l'état de liaison physique pour le périphérique virtuel.**

Vous pouvez activer les mises à jour de l'état de liaison physique pour un périphérique réseau virtuel en procédant comme suit :

- Créez un périphérique réseau virtuel en indiquant`linkprop=phys-state` lors de l'exécution de la commande `ldm add-vnet`.

  La définition de l'option `linkprop=phys-state` configure le périphérique du réseau virtuel pour qu'il obtienne les mises à jour de l'état de liaison physique et les signale à la pile.

  Remarque - Si `linkprop=phys-state` est définie et que la liaison physique est interrompue (même si la connexion au commutateur virtuel fonctionne), le périphérique du réseau virtuel signale son état de liaison comme `down`. Cette situation se produit parce que le SE Oracle Solaris ne fournit actuellement pas d'interface pour signaler deux états de liaison distincts, notamment l'état de liaison virtuelle et l'état de liaison physique.

  ```
  primary# ldm add-vnet linkprop=phys-state if-name vswitch-name domain-name
  ```

  L'exemple suivant active les mises à jour de l'état de liaison physique pour `ldom1_vnet0` connecté à `primary-vsw0` sur le domaine logique `ldom1` :

  ```
  primary# ldm add-vnet linkprop=phys-state ldom1_vnet0 primary-vsw0 ldom1
  ```

- Modifiez un périphérique réseau virtuel existant en indiquant`linkprop=phys-state` lors de l'exécution de la commande `ldm set-vnet`.

  ```
  primary# ldm set-vnet linkprop=phys-state if-name domain-name
  ```

  L'exemple suivant active les mises à jour de l'état de liaison physique pour `vnet0` sur le domaine logique `ldom1` :

  ```
  primary# ldm set-vnet linkprop=phys-state ldom1_vnet0 ldom1
  ```

Pour désactiver les mises à jour de l'état de liaison physique, définissez`linkprop=` en exécutant la commande `ldm set-vnet`.
L'exemple suivant désactive les mises à jour de l'état de liaison physique pour `ldom1_vnet0` sur le domaine logique `ldom1` :

```
primary# ldm set-vnet linkprop= ldom1_vnet0 ldom1
```

Exemple 45 Configuration d'IPMP basé sur liaison

Les exemples suivants indiquent comment configurer l'IPMP basé sur liaison, en procédant ou non aux mises à jour de l'état de liaison physique :

- L'exemple suivant configure deux périphériques de réseau virtuel sur un domaine. Chaque périphérique réseau virtuel est connecté à un périphérique de commutateur virtuel distinct sur le domaine de service pour utiliser l'IPMP basé sur liaison.

Remarque - Les adresses de test ne sont pas configurées sur ces périphériques de réseau virtuel. Il est également inutile d'effectuer une configuration supplémentaire lorsque vous utilisez la commande `ldm add-vnet` pour créer ces périphériques de réseau virtuel.

Les commandes suivantes ajoutent les périphériques de réseau virtuel au domaine. Notez que comme `linkprop=phys-state` n'est pas défini, seule la liaison au commutateur virtuel est surveillée pour les changements d'état.

```
primary# ldm add-vnet ldom1_vnet0 primary-vsw0 ldom1
primary# ldm add-vnet ldom1_vnet1 primary-vsw1 ldom1
```

Les commandes suivantes configurent les périphériques du réseau virtuel sur le domaine invité et les assignent à un groupe IPMP. Notez que les adresses de test ne sont pas configurées sur ces périphériques de réseau virtuel, car la détection des pannes de liaison est en cours d'utilisation.

- **Sous Oracle Solaris 10 :** utilisez la commande `ifconfig`.

  ```
  # ifconfig vnet0 plumb
  # ifconfig vnet1 plumb
  # ifconfig vnet0 group ipmp0
  # ifconfig vnet1 group ipmp0
  ```

  Les deuxième et troisième commandes configurent l'interface `ipmp0` avec l'adresse IP, en fonction des besoins.

- **Sous Oracle Solaris 11 :** utilisez la commande `ipadm`.

  Notez que `net0` et `net1` sont respectivement les noms propres Oracle Solaris 11 de `vnet0` et `vnet1`.

  ```
  # ipadm create-ip net0
  # ipadm create-ip net1
  # ipadm create-ipmp ipmp0
  # ipadm add-ipmp -i net0 -i net1 ipmp0
  ```

- L'exemple suivant configure deux périphériques de réseau virtuel sur un domaine. Chaque domaine est connecté à un périphérique de commutateur virtuel distinct sur le domaine de service pour utiliser l'IPMP basé sur liaison. Les périphériques de réseau virtuel sont également configurés pour obtenir les mises à jour de l'état de la liaison physique.

  ```
  primary# ldm add-vnet linkprop=phys-state ldom1_vnet0 primary-vsw0 ldom1
  primary# ldm add-vnet linkprop=phys-state ldom1_vnet1 primary-vsw1 ldom1
  ```

Remarque - Le commutateur virtuel doit disposer d'un périphérique de réseau physique assigné pour que le domaine s'associe correctement. Si le domaine est déjà lié et que le commutateur virtuel ne dispose pas d'un périphérique réseau physique assigné, la commande `ldm add-vnet` échouera.

Les commandes suivantes créent les périphériques réseau virtuel et les assignent à un groupe IPMP :

- **Sous Oracle Solaris 10 :** utilisez la commande `ifconfig`.

```
# ifconfig vnet0 plumb
# ifconfig vnet1 plumb
# ifconfig vnet0 192.168.1.1/24 up
# ifconfig vnet1 192.168.1.2/24 up
# ifconfig vnet0 group ipmp0
# ifconfig vnet1 group ipmp0
```

- **Sous Oracle Solaris 11 :** utilisez la commande `ipadm`.

  Notez que `net0` et `net1` sont respectivement les noms propres de `vnet0` et `vnet1`.

```
# ipadm create-ip net0
# ipadm create-ip net1
# ipadm create-ipmp ipmp0
# ipadm add-ipmp -i net0 -i net1 ipmp0
# ipadm create-addr -T static -a 192.168.1.1/24 ipmp0/v4addr1
# ipadm create-addr -T static -a 192.168.1.2/24 ipmp0/v4addr2
```

Utilisation du balisage VLAN

Le logiciel Oracle VM Server for SPARC prend en charge le balisage VLAN 802.1Q dans l'infrastructure réseau.

Les périphériques de commutateur virtuel (`vsw`) et de réseau virtuel (`vnet`) prennent en charge la commutation des paquets Ethernet en fonction de l'identificateur du réseau local virtuel (VLAN) et traitent la balise ou le non balisage nécessaire des cadres Ethernet.

Vous pouvez créer plusieurs interfaces VLAN sur un périphérique réseau virtuel dans un domaine invité. Utilisez la commande `ifconfig` d'Oracle Solaris 10 ou les commandes `dladm` et `ipadm` d'Oracle Solaris 11 pour créer une interface VLAN par le biais d'un périphérique réseau virtuel. La méthode de création est identique à la méthode utilisée pour configurer une interface

VLAN sur tout autre périphérique réseau physique. La contrainte supplémentaire dans un environnement Oracle VM Server for SPARC est que vous devez utiliser la commande `ldm` pour assigner le réseau virtuel aux VLAN pour un périphérique de réseau virtuel `vsw` ou `vnet`. Reportez-vous à la page de manuel `ldm`(1M).

De même, vous pouvez configurer des interfaces VLAN sur un périphérique de commutateur virtuel dans un domaine de service Oracle Solaris 10. Les ID VLAN 2 à 4094 sont valides. L'ID VLAN 1 est réservé en tant que `default-vlan-id`. Il n'est pas nécessaire de configurer des ID VLAN sur un commutateur virtuel dans un domaine de service Oracle Solaris 11.

Lorsque vous créez un périphérique réseau virtuel sur un domaine invité, vous devez l'assigner aux VLAN requis, en spécifiant un ID de VLAN du port et zéro ou plusieurs ID de VLAN pour ce réseau virtuel, à l'aide des arguments `pvid=` et `vid=` de la commande `ldm add-vnet`. Ces informations configurent le commutateur virtuel afin qu'il prenne en charge plusieurs VLAN dans le réseau Oracle VM Server for SPARC et commute les paquets à l'aide de l'adresse MAC et des ID de VLAN dans le réseau.

Tout VLAN utilisé par un domaine de service Oracle Solaris 10 doit être configuré sur le périphérique `vsw`. Utilisez la commande `ldm add-vsw` ou `ldm set-vsw` pour spécifier les valeurs de propriété `pvid` et `vid`.

Vous pouvez modifier les VLAN auxquels un périphérique appartient à l'aide de la commande `ldm set-vnet` ou `ldm set-vsw`.

ID de VLAN du port

L'ID de VLAN du port (PVID) indique le VLAN auquel le périphérique réseau virtuel doit appartenir en mode non balisé. Dans ce cas, le périphérique `vsw` fournit le balisage ou le non balisage nécessaire des cadres pour le périphérique `vnet` sur le VLAN défini par son PVID. Les cadres de sortie du réseau virtuel non balisés sont balisés avec son PVID par le commutateur virtuel. Les cadres entrants balisés par ce PVID sont non balisés par le commutateur virtuel avant leur envoi au périphérique `vnet`. Par conséquent, l'assignation d'un PVID à un réseau virtuel signifie implicitement que le port de réseau virtuel correspondant sur le commutateur virtuel est marqué comme non balisé pour le VLAN spécifié par le PVID. Il ne peut y avoir qu'un seul PVID pour un périphérique réseau virtuel.

L'interface réseau virtuel correspondante, lorsqu'elle est configurée sans ID de VLAN et uniquement à l'aide de son instance de périphérique, fait que l'interface est implicitement assignée au VLAN spécifié par le PVID du réseau virtuel.

Par exemple, si vous envisagez de créer l'instance `0` de périphérique réseau à l'aide de l'une des commandes suivantes et que l'argument `pvid=` pour `vnet` a été défini sur `10`, l'interface `vnet0` serait assignée de manière implicite afin d'appartenir au VLAN 10. Notez que les commandes

suivantes indiquent les noms d'interface `vnet0`, qui concernent Oracle Solaris 10. Pour Oracle Solaris 11, utilisez à sa place le nom générique, par exemple `net0`.

- **Sous Oracle Solaris 10 :** utilisez la commande `ifconfig`.

  ```
  # ifconfig vnet0 plumb
  ```

- **Sous Oracle Solaris 11 :** utilisez la commande `ipadm`.

  ```
  # ipadm create-ip net0
  ```

ID de VLAN

L'ID VID (VID) indique le VLAN auquel un périphérique réseau virtuel ou un commutateur virtuel doit appartenir en mode balisé. Le périphérique réseau virtuel envoie et reçoit des cadres balisés sur les VLAN spécifiés par ses VID. Le commutateur virtuel transmet les cadres balisés avec le VID spécifié entre le périphérique réseau virtuel et le réseau externe.

Assignation et utilisation des VLAN

Les exemples de périphérique dans les tâches suivantes utilisent un numéro d'instance de 0 dans les domaines. Les VLAN sont mappés pour les sous-réseaux suivants :

- Sous-réseau VLAN 20 `192.168.1.0` (masque réseau : `255.255.255.0`)
- Sous-réseau VLAN 21 `192.168.2.0` (masque réseau : `255.255.255.0`)
- Sous-réseau VLAN 22 `192.168.3.0` (masque réseau : `255.255.255.0`)

▼ Procédures d'assignation et d'utilisation des VLAN dans un domaine de service Oracle Solaris 11

1. **Assignez le commutateur virtuel (`vsw`).**

   ```
   primary# ldm add-vsw net-dev=net0 primary-vsw0 primary
   ```

2. **Créez l'interface VLAN dans le domaine de service.**

   Notez que l'option `-T static` de la commande `ipadm create-addr` est requise uniquement en cas d'exécution d'un système d'exploitation Oracle Solaris 11 postérieur au SE Oracle Solaris 11.1. A partir du SE Oracle Solaris 11, `-T static` est le comportement par défaut.

   ```
   # ipadm create-ip net0
   # ipadm create-addr -T static -a 192.169.2.100/24 net0
   # dladm create-vlan -l net0 -v 20 vlan20
   ```

```
# ipadm create-ip vlan20
# ipadm create-addr -T static -a 192.168.1.100/24 vlan20
```

Pour plus d'informations sur la procédure de configuration des interfaces VLAN dans le SE Oracle Solaris 11, reportez-vous au Chapitre 3, "Configuring Virtual Networks by Using Virtual Local Area Networks" du manuel *Managing Network Datalinks in Oracle Solaris 11.3*.

▼ Procédures d'assignation et d'utilisation de VLAN dans un domaine de service Oracle Solaris 10

1. Assignez le commutateur virtuel (`vsw`) à deux VLAN.

Par exemple, configurez le VLAN 21 comme non balisé et le VLAN 20 comme balisé. Notez que le domaine de service n'est pas configuré pour accéder à l'ID VLAN 22.

```
primary# ldm add-vsw net-dev=nxge0 pvid=21 vid=20 primary-vsw0 primary
```

2. Créez l'interface VLAN dans le domaine de service.

```
# ifconfig vsw0 plumb
# ifconfig vsw0 192.168.2.100 netmask 0xffffff00 broadcast + up
# ifconfig vsw20000 plumb
# ifconfig vsw20000 192.168.1.100 netmask 0xffffff00 broadcast + up
```

▼ Procédures d'assignation et d'utilisation des VLAN dans un domaine invité Oracle Solaris 11

Une fois cette tâche terminée, le domaine invité `ldom1` peut communiquer avec le domaine de service `primary` ainsi qu'avec les systèmes externes et distants qui utilisent l'ID VLAN 21 balisé en externe et les adresses IP sur 192.168.2.0/24. Le domaine invité `ldom1` peut aussi communiquer avec le domaine de service et les systèmes externes et distants qui utilisent l'ID VLAN 20 balisé en externe et les adresses IP sur 192.168.1.0/24. Le domaine invité `ldom1` peut communiquer uniquement avec les systèmes externes mais pas avec le service de domaine qui utilise l'ID VLAN 22 et les adresses IP sur 192.168.3.0/24.

1. Assignez le réseau virtuel (`vnet`) à deux VLAN.

Par exemple, configurez le VLAN 21 comme non balisé et le VLAN 20 comme balisé.

```
primary# ldm add-vnet pvid=21 vid=20,22 vnet0 primary-vsw0 ldom1
ldom1# ipadm create-ip net0
ldom1# ipadm create-addr -t 192.168.2.101/24 net0
```

2. Créez l'interface VLAN dans le domaine invité.

```
ldom1# dladm create-vlan -l net0 -v 20 vlan20
ldom1# ipadm create-ip vlan20
ldom1# ipadm create-addr -t 192.168.1.101/24 vlan20
```

```
ldom1# dladm create-vlan -l net0 -v 22 vlan22
ldom1# ipadm create-ip vlan22
ldom1# ipadm create-addr -t 192.168.3.101/24 vlan22
```

▼ Procédures d'assignation et d'utilisation des VLAN dans un domaine invité Oracle Solaris 10

Une fois cette tâche terminée, le domaine invité `ldom1` peut communiquer avec le domaine de service `primary` ainsi qu'avec les systèmes externes et distants qui utilisent l'ID VLAN 21 balisé en externe et les adresses IP sur 192.168.2.0/24. Le domaine invité `ldom1` peut aussi communiquer avec le domaine de service et les systèmes externes et distants qui utilisent l'ID VLAN 20 balisé en externe et les adresses IP sur 192.168.1.0/24. Le domaine invité `ldom1` peut communiquer uniquement avec les systèmes externes mais pas avec le service de domaine qui utilise l'ID VLAN 22 et les adresses IP sur 192.168.3.0/24.

1. **Assignez le réseau virtuel (`vnet`) à deux VLAN.**

   Par exemple, configurez le VLAN 21 comme non balisé et le VLAN 20 comme balisé.

   ```
   primary# ldm add-vnet pvid=21 vid=20,22 vnet0 primary-vsw0 ldom1
   ldom1# ifconfig vnet0 plumb
   ldom1# ifconfig vnet0 192.168.2.101 netmask 0xffffff00 broadcast + up
   ```

2. **Créez l'interface VLAN dans le domaine invité.**

   ```
   ldom1# ifconfig vnet20000 plumb
   ldom1# ifconfig vnet20000 192.168.1.102 netmask 0xffffff00 broadcast + up
   ldom1# ifconfig vnet22000 plumb
   ldom1# ifconfig vnet22000 192.168.3.102 netmask 0xffffff00 broadcast + up
   ```

▼ Procédure d'installation d'un domaine invité lorsque le serveur d'installation fait partie d'un VLAN

Soyez prudent lorsque vous utilisez la fonction JumpStart d'Oracle Solaris pour installer un domaine invité sur le réseau lorsque le serveur d'installation fait partie d'un VLAN. Cette fonction n'est prise en charge que sur les systèmes Oracle Solaris 10.

Pour plus d'informations sur l'utilisation de la fonction JumpStart d'Oracle Solaris afin d'installer un domaine invité, reportez-vous à la section "Procédure d'utilisation de la fonction JumpStart d'Oracle Solaris sur un domaine invité Oracle Solaris 10" à la page 55.

1. **Configurez le périphérique réseau en mode non balisé.**

Par exemple, si le serveur d'installation est en VLAN 21, configurez le réseau virtuel au départ comme suit :

```
primary# ldm add-vnet pvid=21 vnet01 primary-vsw0 ldom1
```

Ne configurez pas de VLAN balisés (`vid`) pour ce périphérique réseau virtuel. Vous devez effectuer cette opération, car l'OpenBoot PROM (OBP) ne connaît pas les VLAN et ne peut pas traiter les paquets réseau balisés VLAN.

2. **A la fin de l'installation et du démarrage du SE Oracle Solaris, configurez le réseau virtuel en mode balisé.**

```
primary# ldm set-vnet pvid= vid=21, 22, 23 vnet01 primary-vsw0 ldom1
```

Vous pouvez désormais ajouter le périphérique réseau virtuel à d'autres VLAN en mode balisé.

Utilisation des VLAN privés

Le mécanisme de VLAN privé (PVLAN) vous permet de diviser un VLAN standard en sous-VLAN pour isoler le trafic réseau. Le mécanisme PVLAN est défini dans RFC 5517 (`http://tools.ietf.org/html/rfc5517`). Généralement, un VLAN standard est un domaine de diffusion unique, mais quand il est configuré avec les propriétés PVLAN, le domaine de diffusion unique est divisé en sous-domaines de diffusion plus petits, tout en conservant la configuration de couche 3 existante. Lorsque vous configurez un PVLAN, le VLAN standard est appelé *VLAN principal* et les sous-VLAN sont appelés *VLAN secondaires*.

Lorsque deux réseaux virtuels utilisent le même ID de VLAN sur une liaison physique, l'ensemble du trafic de diffusion est transmis entre les deux réseaux virtuels. Toutefois, lorsque vous créez des réseaux virtuels qui utilisent les propriétés PVLAN, le comportement de transfert de paquets peut ne pas s'appliquer dans tous les cas.

Le tableau suivant répertorie les règles de transfert de paquets de diffusion pour les PVLAN isolés et communautaires.

TABLEAU 2 Règles de transfert de paquets de diffusion

Type PVLAN	Isolé	Communauté A	Communauté B
Isolé	Non	Non	Non
Communauté A	Non	Oui	Non
Communauté B	Non	Non	Oui

Par exemple, lorsque les deux réseaux virtuels `vnet0` et `vnet1`sont isolés sur le réseau `net0`, `net0` ne transmet pas de trafic entre deux réseaux virtuels. Toutefois, lorsque le réseau `net0` reçoit

le trafic d'un VLAN isolé, le trafic n'est pas transmis aux ports isolés liés au VLAN. Cette situation se produit parce que le réseau virtuel isolé n'accepte que le trafic du VLAN principal.

La fonction inter-vnet-link prend en charge les restrictions de communication des PVLAN isolés et communautaires. La fonction inter-vnet-link est désactivée pour les PVLAN isolés et activée uniquement pour les réseaux virtuels qui se trouvent dans la même communauté que les PVLAN communautaires. La redirection du trafic à partir d'autres réseaux virtuels en dehors de la communauté n'est pas autorisée.

Remarque - Si un domaine de service cible ne prend pas en charge la fonctionnalité PVLAN, la migration d'un domaine invité configuré pour un PVLAN peut échouer.

Conditions d'utilisation des PVLAN

Vous pouvez configurer les PVLAN à l'aide des commandes `ldm add-vnet` et `ldm set-vnet`. Ces commandes permettent de définir la propriété `pvlan`. Notez que vous devez également spécifier la propriété `pvid` pour configurer correctement les PVLAN.

Cette fonction requiert au minimum le système d'exploitation Oracle Solaris 11.2 SRU 4.

Pour configurer un PVLAN, vous devez spécifier les informations suivantes :

- **ID de VLAN principal.** L'ID de VLAN principal est l'ID de VLAN du port (PVID) utilisé pour configurer un PVLAN pour un périphérique réseau virtuel unique. Cette configuration permet de s'assurer qu'un domaine invité ne reçoit pas de paquets VLAN. Notez que vous ne pouvez pas configurer de VID avec un PVLAN. Cette valeur est représentée par la propriété `pvid`.
- **ID de VLAN secondaire.** La fonctionnalité PVLAN est fournie grâce à un VLAN particulier utilisant un ID de VLAN secondaire. Vous précisez ces informations comme étant la partie *secondary-vid* de la valeur `pvlan`. *secondary-vid* est une valeur entière située entre 1 et 4094. Un VLAN principal peut avoir de nombreux VLAN secondaires, avec les restrictions suivantes :
  - Ni l'ID de VLAN principal ni l'ID de VLAN secondaire peuvent être identiques à l'ID de VLAN par défaut.
  - L'ID de VLAN principal et l'ID de VLAN secondaire ne peuvent pas avoir les même valeurs pour les types de PVLAN isolés et communautaires.
  - Chaque VLAN principal ne peut configurer qu'un PVLAN isolé. Vous ne pouvez donc pas créer deux PVLAN isolés utilisant le même ID de VLAN principal.
  - Un VLAN principal peut avoir plusieurs VLAN communautaires, avec les restrictions suivantes :

- Un ID de VLAN ne peut être utilisé car l'ID de VLAN secondaire crée un autre PVLAN communautaire.

  Par exemple, si vous disposez d'un PVLAN communautaire avec un ID de VLAN principal de `3` et un ID de VLAN secondaire de `100`, vous ne pouvez pas créer un autre PVLAN communautaire qui utilise `3` en tant qu'ID de VLAN secondaire.
- Un ID de VLAN secondaire ne peut être utilisé en tant qu'ID de VLAN principal pour créer un PVLAN communautaire.

  Par exemple, si vous disposez d'un PVLAN communautaire avec un ID de VLAN principal de `3` et un ID de VLAN secondaire de `100`, vous ne pouvez pas créer un autre PVLAN communautaire qui utilise `100` comme ID de VLAN principal.
- L'ID de VLAN secondaire ne peut pas encore être utilisé en tant qu'ID de VLAN pour les réseaux virtuels ou les VNIC standard.

Attention - Le Logical Domains Manager peut uniquement valider la configuration des réseaux virtuels sur un commutateur virtuel particulier. Si une configuration du PVLAN est définie pour les VNIC Oracle Solaris sur le même périphérique backend, assurez-vous que les mêmes exigences sont respectées sur toutes les VNIC et tous les réseaux virtuels.

- **Type de PVLAN.** Vous précisez ces informations comme étant la partie *pvlan-type* de la valeur `pvlan`. *pvlan-type* correspond à l'une des valeurs suivantes :
  - `isolated`**.** Les ports associés avec un PVLAN isolé sont isolés de tous les réseaux virtuels homologues et de toutes les cartes NIC virtuelles Oracle Solaris sur le périphérique réseau backend. Les paquets atteignent uniquement le réseau externe en fonction des valeurs spécifiées pour le PVLAN.
  - `community`**.** Les ports associés à un PVLAN communautaire peuvent communiquer avec d'autres ports qui se trouvent dans le même PVLAN communautaire mais sont isolés de tous les autres ports. Les paquets atteignent le réseau externe en fonction des valeurs spécifiées pour le PVLAN.

Configuration des PVLAN

Cette section inclut des tâches décrivant la création de PVLAN, ainsi que des informations y relatives.

Création d'un PVLAN

Vous pouvez configurer les PVLAN en définissant la valeur de propriété `pvlan` à l'aide des commandes `ldm add-vnet` ou `ldm set-vnet`. Reportez-vous à la page de manuel `ldm`(1M).

Vous pouvez utiliser les commandes suivantes pour créer ou supprimer un PVLAN :

- Utilisez `ldm add-vnet` pour créer un PVLAN :

  ```
  ldm add-vnet pvid=port-VLAN-ID pvlan=secondary-vid,pvlan-type \
  if-name vswitch-name domain-name
  ```

  La commande indique comment créer un réseau virtuel avec un PVLAN qui possède un *VLAN-ID* principal de `4`, un *vlan-id* secondaire de `200` et un *pvlan-type* `isolated`.

  ```
  primary# ldm add-vnet pvid=4 pvlan=200,isolated vnet1 primary-vsw0 ldg1
  ```

- Utilisez `ldm set-vnet` pour créer un PVLAN :

  ```
  ldm set-vnet pvid=port-VLAN-ID pvlan=secondary-vid,pvlan-type if-name domain-name
  ```

  La commande indique comment créer un réseau virtuel avec un PVLAN qui possède un *vlan-id* principal de `3`, un *vlan-id* secondaire de `300` et un *pvlan-type* `community`.

  ```
  primary# ldm set-vnet pvid=3 pvlan=300,community vnet2 ldg1
  ```

- Utilisez `ldm set-vnet` pour supprimer un PVLAN :

  ```
  ldm set-vnet pvlan= if-name domain-name
  ```

  La commande suivante supprime la configuration du PVLAN pour le réseau virtuel `vnet0`. Pour rétablir un ID de VLAN normal pour le réseau virtuel `vnet0`, vous devez d'abord supprimer l'ID de PVLAN.

  ```
  primary# ldm set-vnet pvlan= vnet0 ldg1
  ```

Visualisation des informations du PVLAN

Vous pouvez visualiser les informations sur un PVLAN grâce aux différentes sous-commandes de liste du Logical Domains Manager. Reportez-vous à la page de manuel `ldm`(1M).

Vous pouvez utiliser la commande suivante pour afficher les informations du PVLAN.

- Utilisez `ldm list-domain -o network` pour répertorier les informations du PVLAN :

  ```
  ldm list-domain [-e] [-l] -o network [-p] [domain-name...]
  ```

Les exemples suivants présentent des informations sur la configuration du PVLAN sur le domaine `ldg1` à l'aide de la commande `ldm list-domain -o network`.

- La commande `ldm list-domain` suivante affiche des informations sur la configuration du PVLAN sur le domaine `ldg1`.

```
primary# ldm list-domain -o network ldg1
NAME
ldg1

MAC
     00:14:4f:fb:22:79

NETWORK
     NAME         SERVICE                MACADDRESS PVID|PVLAN|VIDs
     ----         -------                ---------- ---------------
     vnet0        primary-vsw0@primary   00:14:4f:f8:6e:d9 2|300,community|--
             DEVICE           :network@0       ID   :0
             LINKPROP       :--                  MTU  :1500
             MAXBW          :--                 MODE :--
             CUSTOM        :disable
             PRIORITY       :--                    COS :--
             PROTECTION :--
```

- La commande `ldm list-domain` suivante affiche des informations sur la configuration du PVLAN sous une forme analysable pour le domaine `ldg1`.

```
primary# ldm list-domain -o network -p ldg1
VERSION 1.19
DOMAIN|name=ldg1|
MAC|mac-addr=00:14:4f:fb:22:79
VNET|name=vnet0|dev=network@0|service=primary-vsw0@primary|mac-addr=00:14:4f:f8:
6e:d9|mode=|pvid=2|vid=|mtu=1500|linkprop=|id=0|alt-mac-addrs=|maxbw=|pvlan=300,
community|protection=|priority=|cos=|custom=disable|max-mac-addrs=|max-vlans=
```

- Utilisez `ldm list-bindings` pour répertorier les informations du PVLAN :

```
ldm list-bindings [-e] [-p] [domain-name...]
```

Les exemples suivants présentent des informations sur la configuration du PVLAN sur le domaine `ldg1` à l'aide de la commande `ldm list-bindingsnetwork`.

- La commande `ldm list-bindings` suivante affiche des informations sur la configuration du PVLAN sur le domaine `ldg1`.

```
primary# ldm list-bindings -o network ldg1
NAME
ldg1
```

```
MAC
     00:14:4f:fb:22:79

NETWORK
     NAME         SERVICE                MACADDRESS PVID|PVLAN|VIDs
     ----         -------                ---------- ---------------
     vnet0        primary-vsw0@primary   00:14:4f:f8:6e:d9 2|300,community|--

         PEER                   MACADDRESS          PVID|PVLAN|VIDs
         ----                   ----------          ---------------
         primary-vsw0@primary   00:14:4f:f9:08:28   1|--|--
```

- La commande `ldm list-bindings` suivante affiche des informations sur la configuration du PVLAN sous une forme analysable pour le domaine `ldg1`.

```
primary# ldm list-bindings -o network -p ldg1
VERSION 1.19
DOMAIN|name=ldg1|
MAC|mac-addr=00:14:4f:fb:22:79
VNET|name=vnet0|dev=network@0|service=primary-vsw0@primary|mac-addr=00:14:4f:f8:
6e:d9|mode=|pvid=2|vid=|mtu=1500|linkprop=|id=0|alt-mac-addrs=|maxbw=|pvlan=300,
community|protection=|priority=|cos=|custom=disable|max-mac-addrs=|max-vlans=
|peer=primary-vsw0@primary|mac-addr=00:14:4f:f9:08:28|mode=|pvid=1|vid=|mtu=1500|
maxbw=
```

- Utilisez la commande `ldm list-constraints` pour répertorier les informations du PVLAN :

```
ldm list-constraints [-x] [domain-name...]
```

La section suivante présente la sortie générée en exécutant la commande `ldm list-constraints` :

```
primary# ldm list-constraints -x ldg1
...
<Section xsi:type="ovf:VirtualHardwareSection_Type">
  <Item>
    <rasd:OtherResourceType>network</rasd:OtherResourceType>
    <rasd:Address>auto-allocated</rasd:Address>
    <gprop:GenericProperty key="vnet_name">vnet0</gprop:GenericProperty>
    <gprop:GenericProperty key="service_name">primary-vsw0</gprop:GenericProperty>
    <gprop:GenericProperty key="pvid">1</gprop:GenericProperty>
    <gprop:GenericProperty key="vid">3</gprop:GenericProperty>
    <gprop:GenericProperty key="pvlan">200,community</gprop:GenericProperty>
    <gprop:GenericProperty key="maxbw">1700000000</gprop:GenericProperty>
    <gprop:GenericProperty key="device">network@0</gprop:GenericProperty>
    <gprop:GenericProperty key="id">0</gprop:GenericProperty>
```

```
</Item>
```

Réglage des performances du débit des paquets

Vous pouvez utiliser les commandes `ldm add-vnet` et `ldm set-vnet` pour régler les valeurs de propriété des liaisons de données et régler les performances du débit des paquets.

`priority`	Permet de définir le niveau de priorité de traitement des paquets de la CPU
`cos`	Indique la classe de service de liaison IEEE 802.1p de la liaison
`protection`	Indique le type de sécurité du trafic des paquets

Pour plus d'informations sur les valeurs de propriété par défaut et les valeurs valides, reportez-vous à la page du manuel `ldm`(1M).

EXEMPLE 46 Définition et affichage des propriétés des paquets de liaisons de données

L'exemple suivant illustre comment utiliser la commande `ldm set-vnet` permettant de définir les valeurs de propriété `priority`, `protection` et `cos` dans une seule commande. Vous pouvez également utiliser la commande `ldm add-vnet` pour ajouter un nouveau réseau virtuel qui utilise des valeurs de propriété de liaison de données spécifiques.

```
primary# ldm set-vnet allowed-ips=192.168.100.1,192.168.100.2 \
allowed-dhcp-cids=oracle@system1.company.com, \
00:14:4f:fb:22:79,system2,00:14:4f:fb:22:56 cos=7 priority=high \
protection=restricted,mac-nospoof,ip-nospoof,dhcp-nospoof vnet3_ldg3 ldg3
```

La commande `ldm list -o network` affiche les valeurs des propriétés de liaison de données sur le domaine `ldg3` que vous définissez à l'aide de la commande `ldm set-vnet` précédente. Les valeurs de protection sont `mac-nospoof`, `restricted`, `ip-nospoof` pour l'adresse MAC `192.168.100,1,192.168.100.2` et `dhcp-nospoof` pour `system1@company.com,00:14:4f:f9:d3:88, system2,00:14:4f:fb:61:6e`. La valeur `priority` est définie sur `high` et la classe de service (`cos`) sur `7`.

```
primary# ldm list-domain -o network ldg3
NAME
ldg3

MAC
     00:14:4f:f8:5b:12
     NAME         SERVICE                MACADDRESS PVID|PVLAN|VIDs
     ----         -------                ---------- ---------------
     vnet3_ldg3   primary-vsw0@primary   00:14:4f:f8:dd:96 1|--|--
            DEVICE     :network@1       ID   :1
            LINKPROP   :phys-state      MTU  :1500
```

```
MAXBW      :--              MODE :--
CUSTOM     :disable
PRIORITY   :high            COS  :7
PROTECTION :mac-nospoof
            restricted
            ip-nospoof
            [192.168.100.1
             192.168.100.2]
            dhcp-nospoof
            [oracle@system1.company.com
             00:14:4f:fb:22:79
             system2
             00:14:4f:fb:22:56]
```

Utilisation du groupement de liaisons avec un commutateur virtuel

Il est possible de configurer un commutateur virtuel de manière à ce qu'il utilise un groupement de liaisons. Un groupement de liaisons est utilisé comme périphérique réseau du commutateur virtuel pour se connecter à un réseau physique. Cette configuration permet au commutateur virtuel d'optimiser les fonctions fournies par la norme de groupement de liaisons IEEE 802.3ad. De telles fonctions comprennent une bande passante accrue, un équilibrage de la charge et un basculement. Pour plus d'informations sur la configuration d'agrégation de liens, reportez-vous à la section "Creating a Link Aggregation" du manuel *Managing Network Datalinks in Oracle Solaris 11.3*.

Après avoir créé un groupement de liaisons, vous pouvez l'assigner à un commutateur virtuel. La réalisation de cette assignation est semblable à l'assignation d'un périphérique réseau physique à un commutateur virtuel. Utilisez la commande `ldm add-vswitch` ou `ldm set-vswitch` pour définir la propriété `net-dev`.

Lorsque le groupement de liaisons est assigné au commutateur virtuel, le trafic de et vers le réseau physique s'écoule au travers du groupement. Tout équilibrage de charge ou basculement nécessaire est traité de manière transparente par le cadre de groupement sous-jacent. Le groupement de liaisons est complètement transparent aux périphériques du réseau virtuel (`vnet`) qui se trouvent sur les domaines invités et sont liés à un commutateur virtuel utilisant un groupement.

Remarque - Vous ne pouvez pas grouper des périphériques de réseau virtuel (`vnet` et `vsw`) dan un groupement de liaisons.

Vous pouvez créer et utiliser le commutateur virtuel configuré pour utiliser un groupement de liaisons dans le domaine de service. Reportez-vous à la section "Procédure de configuration du commutateur virtuel en tant qu'interface primary" à la page 43.

Les Figure 23, "Configuration d'un commutateur virtuel pour utiliser une agrégation de liens (Oracle Solaris 11)" et Figure 24, "Configuration d'un commutateur virtuel pour utiliser une agrégation de liens (Oracle Solaris 10)" illustrent un commutateur virtuel configuré pour utiliser une agrégation, `aggr1`, sur des interfaces physiques `net0` et `net1`, et `nxge0` et `nxge1`, respectivement.

FIGURE 23 Configuration d'un commutateur virtuel pour utiliser une agrégation de liens (Oracle Solaris 11)

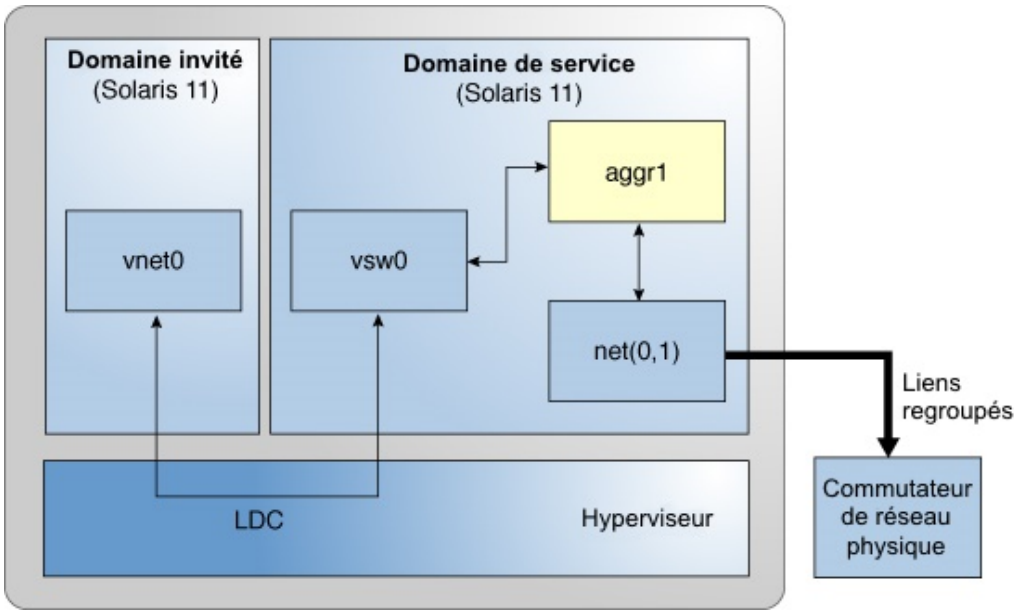
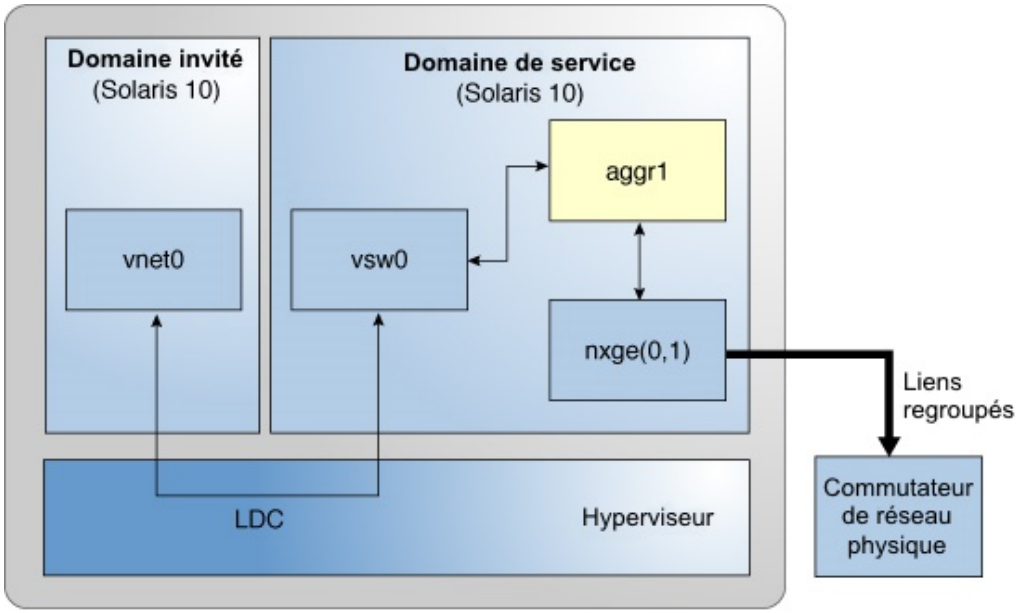


FIGURE 24 Configuration d'un commutateur virtuel pour utiliser une agrégation de liens (Oracle Solaris 10)



Configuration de trames géantes

Le commutateur virtuel de Oracle VM Server for SPARC (`vsw`) et les périphériques de réseau virtuel (`vnet`) prennent maintenant en charge les cadres Ethernet avec des tailles de charge utile supérieures à 1 500 octets. Par conséquent, ces pilotes peuvent maintenant augmenter la capacité de traitement du réseau.

Vous activez les trames géantes en définissant l'unité de transmission maximale (MTU) pour le périphérique de commutateur virtuel. Dans ce cas, le périphérique de commutateur virtuel et tous les autres périphériques du réseau virtuel sont liés au périphérique de commutateur virtuel utilisant la valeur MTU définie.

Si la valeur MTU obligatoire pour le périphérique réseau virtuel doit être inférieure à celle prise en charge par le commutateur virtuel, vous pouvez spécifier une valeur MTU directement sur un périphérique réseau virtuel.

Remarque - Sur le SE Oracle Solaris 10 5/09 uniquement, la MTU d'un périphérique physique doit être configurée pour correspondre à la MTU du commutateur virtuel. Pour plus d'informations sur la configuration de ces pilotes en particulier, reportez-vous à la page de manuel qui correspond à ce pilote dans la section 7D du manuel de référence Oracle Solaris. Par exemple, pour obtenir des informations sur le pilote `nxge` Oracle Solaris 10, reportez-vous à la page de manuel `nxge`(7D).

Dans de rares cas, vous devrez utiliser la commande `ldm add-vnet` ou `ldm set-vnet` pour spécifier une valeur MTU pour un périphérique réseau virtuel qui diffère de la valeur MTU du commutateur virtuel. Par exemple, vous pouvez modifier la valeur MTU du périphérique réseau virtuel si vous configurez des VLAN sur un périphérique réseau virtuel et la MTU du VLAN la plus grande est inférieure à la valeur MTU sur le commutateur virtuel. Un pilote `vnet` qui prend en charge les trames géantes n'est pas toujours nécessaire pour les domaines dans lesquels seule la valeur MTU par défaut est utilisée. Cependant, si les domaines ont des périphériques de réseau virtuel liés à un commutateur virtuel utilisant des trames géantes, vérifiez que le pilote `vnet` prend en charge les trames géantes.

Si vous utilisez la commande `ldm set-vnet` pour définir une valeur `mtu` sur un périphérique réseau virtuel, les mises à jour ultérieures de la valeur MTU du périphérique de commutateur virtuel ne sont pas propagées à ce périphérique de commutateur virtuel. Pour réactiver le périphérique réseau virtuel afin d'obtenir la valeur MTU du périphérique de commutateur virtuel, exécutez la commande suivante :

```
primary# ldm set-vnet mtu= vnet-name domain-name
```

Sur le domaine de contrôle, Logical Domains Manager met à jour les valeurs MTU initiées par les commandes `ldm set-vsw` et `ldm set-vnet` comme des opérations de reconfiguration retardée. Pour apporter des mises à jour MTU sur des domaines autres que le domaine de contrôle, vous devez arrêter un domaine avant d'exécuter la commande `ldm set-vsw` ou `ldm set-vnet` pour modifier une valeur MTU.

▼ Procédure de configuration du réseau virtuel et des périphériques de commutateur virtuels pour l'utilisation de trames géantes

1. **Connectez-vous au domaine de contrôle.**

2. **Connectez-vous en tant qu'administrateur.**

   Pour Oracle Solaris 11.3, reportez-vous au Chapitre 1, "About Using Rights to Control Users and Processes" du manuel *Securing Users and Processes in Oracle Solaris 11.3*.

3. **Déterminez la valeur de MTU que vous voulez utiliser pour ce réseau virtuel.**

   Vous pouvez indiquer une valeur MTU comprise entre 1 500 et 16 000 octets. La MTU indiquée doit correspondre à la MTU du périphérique du réseau physique qui est assignée au commutateur virtuel.

4. **Indiquez la valeur MTU d'un périphérique de commutateur virtuel ou d'un périphérique réseau virtuel.**

   Effectuez l'une des opérations suivantes :

   - Activez les trames géantes sur un nouveau périphérique de commutateur virtuel dans le domaine de service en définissant sa MTU comme valeur de la propriété `mtu`.

     ```
     primary# ldm add-vsw net-dev=device mtu=value vswitch-name ldom
     ```

     En plus de configurer le commutateur virtuel, cette commande met à jour la valeur MTU de chaque périphérique réseau virtuel qui sera lié au commutateur virtuel.
   - Activez les trames géantes sur un périphérique de commutateur virtuel existant dans le domaine de service en définissant sa MTU comme valeur de la propriété `mtu`.

     ```
     primary# ldm set-vsw net-dev=device mtu=value vswitch-name
     ```

     En plus de configurer le commutateur virtuel, cette commande met à jour la valeur MTU de chaque périphérique réseau virtuel qui sera lié au commutateur virtuel.

Exemple 47 Configuration des trames géantes sur des périphériques de commutateur virtuel et de réseau virtuel

- L'exemple suivant montre comment ajouter un nouveau périphérique de commutateur virtuel utilisant une valeur MTU de `9000`. Cette valeur MTU est propagée du périphérique de commutateur virtuel à tous les périphériques de réseau virtuel du client.

  En premier lieu, la commande `ldm add-vsw` crée le périphérique de commutateur virtuel, `ldg1-vsw0`, avec une valeur MTU de `9000`. Notez que l'instance 0 du périphérique réseau `net0` est définie comme une valeur de la propriété `net-dev`.

  ```
  primary# ldm add-vsw net-dev=net0 mtu=9000 ldg1-vsw0 ldg1
  ```

  Ensuite, la commande `ldm add-vnet` ajoute un périphérique réseau virtuel client à ce commutateur virtuel, `ldg1-vsw0`. Notez que la MTU du périphérique réseau virtuel est assignée implicitement à partir du commutateur virtuel auquel il est lié. En conséquence, la commande `ldm add-vnet` ne nécessite pas que vous définissiez une valeur pour la propriété `mtu`.

  ```
  primary# ldm add-vnet vnet01 ldg1-vsw0 ldg1
  ```

Selon la version du SE Oracle Solaris en cours d'exécution, procédez comme suit :

- **SE Oracle Solaris 11 :** exécutez la commande `ipadm` pour afficher la valeur de propriété `mtu` de l'interface principale.

```
# ipadm show-ifprop -p mtu net0
IFNAME PROPERTY PROTO PERM CURRENT PERSISTENT DEFAULT POSSIBLE
net0   mtu      ipv4  rw   9000    --         9000    68-9000
```

  La commande `ipadm` crée l'interface réseau virtuelle dans le domaine invité, `ldg1`. La sortie de la commande `ipadm show-ifprop` montre que la valeur de la propriété `mtu` est `9000`.

```
ldg1# ipadm create-ip net0
ldg1# ipadm create-addr -T static -a 192.168.1.101/24 net0/ipv4
ldg1# ipadm show-ifprop -p mtu net0
IFNAME PROPERTY PROTO PERM CURRENT PERSISTENT DEFAULT POSSIBLE
net0   mtu      ipv4  rw   9000    --         9000    68-9000
```

- **SE Oracle Solaris 10 :** La commande `ifconfig` crée l'interface du commutateur virtuel dans le domaine de service, `ldg1`. La sortie de la commande `ifconfig vsw0` montre que la valeur de la propriété `mtu` est `9000`.

```
ldg1# ifconfig vsw0 plumb
ldg1# ifconfig vsw0 192.168.1.100/24 up
ldg1# ifconfig vsw0
vsw0: flags=201000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4,CoS> mtu 9000 index 5
        inet 192.168.1.100 netmask ffffff00 broadcast 192.168.1.255
        ether 0:14:4f:fa:0:99
```

  La commande `ifconfig` crée l'interface réseau virtuel dans le domaine invité, `ldg1`. La sortie de la commande `ifconfig vnet0` montre que la valeur de la propriété `mtu` est `9000`.

```
ldg1# ifconfig vnet0 plumb
ldg1# ifconfig vnet0 192.168.1.101/24 up
ldg1# ifconfig vnet0
vnet0: flags=201000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4,CoS> mtu 9000 index 4
        inet 192.168.1.101 netmask ffffff00 broadcast 192.168.1.255
        ether 0:14:4f:f9:c4:13
```

- L'exemple suivant indique comment modifier la MTU de l'interface et la définir sur `4000`.

  Notez que la MTU d'une interface peut uniquement être remplacée par une valeur inférieure à la MTU du périphérique qui est assigné par Logical Domains Manager. Cette méthode est utile lorsque les VLAN sont configurés et que chaque interface de VLAN a besoin d'une MTU différente.

  - **Sous Oracle Solaris 11 :** utilisez la commande `ipadm`.

```
primary# ipadm set-ifprop -p mtu=4000 net0
primary# ipadm show-ifprop -p mtu net0
IFNAME PROPERTY PROTO PERM CURRENT PERSISTENT DEFAULT POSSIBLE
net0   mtu      ipv4  rw   4000    --         9000    68-9000
```

- **Sous Oracle Solaris 10 :** utilisez la commande `ifconfig`.

```
primary# ifconfig vnet0 mtu 4000
primary# ifconfig vnet0
vnet0: flags=1201000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4,CoS,FIXEDMTU>
mtu 4000 index 4
        inet 192.168.1.101 netmask ffffff00 broadcast 192.168.1.255
        ether 0:14:4f:f9:c4:13
```

Compatibilité avec des versions antérieures (ne connaissant pas les trames géantes) des pilotes `vnet` et `vsw` (Oracle Solaris 10)

Remarque - Cette section s'applique uniquement au SE Oracle Solaris 10.

Les pilotes prenant en charge les trames géantes peuvent interopérer avec les pilotes ne prenant pas en charge les trames géantes sur le même système. Cette interopérabilité est possible tant que la prise en charge des trames géantes n'est pas activée lorsque vous créez le commutateur virtuel.

Remarque - Ne définissez pas la propriété `mtu` si des domaines invités ou de service associés au commutateur virtuel n'utilisent pas les pilotes de Oracle VM Server for SPARC prenant en charge les trames géantes.

Les trames géantes peuvent être activés en modifiant la propriété `mtu` d'un commutateur virtuel de la valeur par défaut à 1 500. Dans cette instance, des versions antérieures des pilotes ignorent le paramètre `mtu` et continuent à utiliser la valeur par défaut. Notez que la sortie `ldm list` affiche la valeur MTU que vous définissez et non pas la valeur par défaut. Les cadres plus grands que la valeur MTU par défaut ne sont pas envoyés à ces périphériques et sont supprimés par les nouveaux pilotes. Cette situation peut provoquer un comportement incohérent du réseau sur ces invités qui utilisent toujours les anciens pilotes. Cette restriction s'applique à la fois aux domaines invités clients et au domaine de service.

Par conséquent, lorsque les trames géantes sont activées, vérifiez que tous les périphériques virtuels dans le réseau Oracle VM Server for SPARC ont été mis à niveau pour utiliser

les nouveaux pilotes prenant en charge les trames géantes. Vous devez exécuter Logical Domains 1.2 au moins pour configurer des trames géantes.

Utilisation des cartes NIC virtuelles sur les réseaux virtuels

Le système d'exploitation Oracle Solaris 11 permet de définir des réseaux virtuels composés de cartes d'interface réseau virtuelles (VNIC), de commutateurs virtuels et d'etherstubs. Les zones Oracle Solaris virtualisent les services de système d'exploitation et fournissent des environnements isolés pour les applications en cours d'exécution au sein de la même instance de SE Oracle Solaris d'un domaine logique.

Oracle Solaris 11 améliore le modèle de zone "IP partagée" d'Oracle Solaris 10 dans lequel les zones héritent des propriétés réseau de la zone globale et ne peuvent pas définir leur propre adresse réseau ou d'autres propriétés. Désormais, en utilisant des zones avec des périphériques de réseau virtuel, vous pouvez configurer plusieurs cartes NIC virtuelles isolées, associer des zones à chaque réseau virtuel et établir des règles pour l'isolement, la connectivité et la qualité de service (QoS).

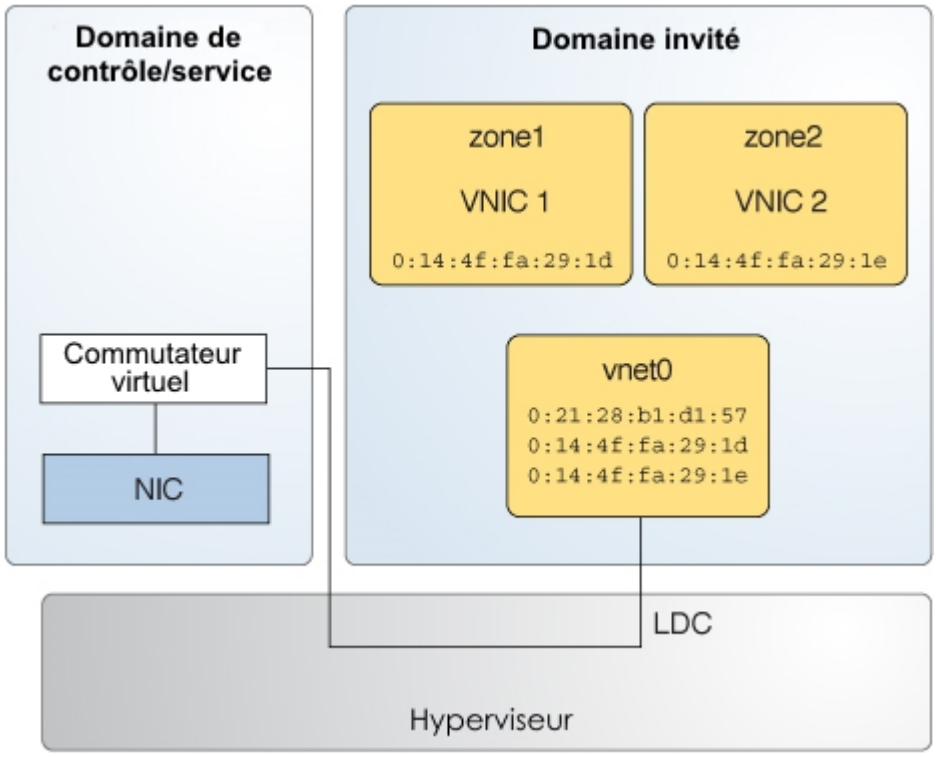
Pour plus d'informations, reportez-vous aux manuels de mise en réseau dans la bibliothèque d'informations Oracle Solaris 11.3 (`http://docs.oracle.com/cd/E53394_01/`).

Un périphérique réseau virtuel dans un domaine logique peut prendre en charge plusieurs cartes NIC virtuelles Oracle Solaris 11. Le périphérique réseau virtuel doit être configuré pour prendre en charge plusieurs adresses MAC, une pour chaque carte NIC virtuelle. Les zones Oracle Solaris du domaine logique se connectent aux cartes NIC virtuelles.

La Figure 25, "Cartes NIC virtuelles sur les périphériques de réseau virtuel" illustre un domaine logique, `domain1`, qui fournit un périphérique réseau virtuel unique `vnet1` au SE Oracle Solaris. Ce périphérique réseau virtuel peut héberger plusieurs périphériques de réseau virtuel Oracle Solaris 11, chacun ayant sa propre adresse MAC et pouvant être assigné individuellement à une zone.

Dans le domaine `domain1`, il y a deux zones Oracle Solaris 11 : `zone1` et `zone2`. Chaque zone est connectée au réseau par une carte NIC virtuelle, basée sur le périphérique réseau virtuel `vnet1`.

FIGURE 25 Cartes NIC virtuelles sur les périphériques de réseau virtuel



Les sections suivantes décrivent la configuration des cartes NIC virtuelles sur des périphériques de réseau virtuel et la création des zones dans le domaine avec les cartes NIC virtuelles :

- "Configuration des cartes NIC virtuelles sur les périphériques de réseau virtuel" à la page 309
- "Création de zones Oracle Solaris 11 dans un domaine" à la page 309

Pour plus d'informations sur l'utilisation des cartes NIC virtuelles sur les fonctions virtuelles SR-IOV Ethernet, reportez-vous aux sections suivantes :

- "Création de fonctions Ethernet virtuelles" à la page 97
- "Modification des fonctions virtuelles SR-IOV Ethernet" à la page 105
- "Création de cartes d'interface réseau virtuelles (VNIC) sur des fonctions virtuelles SR-IOV" à la page 112

Configuration des cartes NIC virtuelles sur les périphériques de réseau virtuel

Pour configurer des cartes NIC virtuelles sur les périphériques de réseau virtuel, le domaine de contrôle doit exécuter au moins le SE Oracle Solaris 11.1 SRU 4 et le domaine invité doit exécuter au moins le SE Oracle Solaris 11.1.

Pour configurer un périphérique réseau virtuel pour héberger plusieurs adresses MAC, exécutez la commande `ldm add-vnet` ou `ldm set-vnet` pour spécifier une ou plusieurs valeurs séparées par des virgules pour la propriété `alt-mac-addrs`. Les valeurs valides sont une adresse MAC à un octet et `auto`. La valeur `auto` indique que le système génère l'adresse MAC.

Par exemple, vous pouvez spécifier trois autres adresses MAC générées par le système pour un périphérique réseau virtuel de l'une des façons suivantes :

- A l'aide de la commande `ldm add-vnet`. La commande `ldm add-vnet` suivante crée le périphérique réseau virtuel `vnet1` sur le domaine `domain1` et met trois adresses MAC générées par le système à la disposition du périphérique.

  ```
  primary# ldm add-vnet alt-mac-addrs=auto,auto,auto vnet1 primary-vsw0 domain1
  ```

- A l'aide d'une combinaison de commandes `ldm add-vnet` et `ldm set-vnet`. Les commandes `ldm add-vnet` et `ldm set-vnet` suivantes montrent comment créer un périphérique réseau virtuel et comment assigner ensuite d'autres adresses MAC au périphérique réseau virtuel existant.

  La première utilise la commande `ldm add-vnet` pour créer le périphérique réseau virtuel `vnet1` sur le domaine `domain1`. La seconde utilise la commande `ldm set-vnet` pour mettre trois adresses MAC générées par le système à la disposition du périphérique réseau virtuel `vnet1`.

  ```
  primary# ldm add-vnet vnet1 primary-vsw0 domain1
  primary# ldm set-vnet alt-mac-addrs=auto,auto,auto vnet1 domain1
  ```

Création de zones Oracle Solaris 11 dans un domaine

Après avoir créé les cartes NIC virtuelles de la section "Configuration des cartes NIC virtuelles sur les périphériques de réseau virtuel" à la page 309, créez une zone associée à une adresse MAC disponible. Pour plus d'informations sur les zones Oracle Solaris, reportez-vous au manuel *Creating and Using Oracle Solaris Zones*.

Exécutez la commande `zonecfg` pour spécifier une adresse MAC à utiliser pour une zone :

`zonecfg:`*zone-name*`> set mac-address=[`*MAC-address*`,auto]`

Vous pouvez spécifier une valeur pour `auto` pour choisir l'une des adresses MAC disponibles automatiquement ou fournir une autre adresse MAC spécifique que vous avez créée à l'aide de la commande `ldm set-vnet`.

Utilisation des réseaux virtuels sécurisés

La fonctionnalité de réseau virtuel sécurisé étend des privilèges aux domaines invités sécurisés et permet l'affectation dynamique d'adresses MAC et ID VLAN alternatifs aux périphériques `vnet`. Ces adresses MAC et ID VLAN servent à configurer des périphériques virtuels. Avant l'introduction de cette fonctionnalité, il n'était possible d'effectuer ces affectations qu'à partir du Logical Domains Manager. En outre, l'affectation d'adresses MAC alternatives nécessitait que le domaine hébergeant le périphérique de réseau virtuel soit lié. Cette fonctionnalité permet la création dynamique de périphériques virtuels tels que des cartes d'interface réseau virtuelles (VNIC) et des VLAN venant s'ajouter aux périphériques de réseau virtuel.

Pour utiliser la fonctionnalité de réseau virtuel sécurisé sur un périphérique `vnet`, vous devez créer ou configurer le périphérique en mode sécurisé à l'aide du Logical Domains Manager. Par défaut, le mode sécurisé est activé quand un périphérique `vnet` est créé.

La fonctionnalité de réseau virtuel sécurisé prend en charge la migration en direct, la réinitialisation du domaine de service ainsi que plusieurs fonctions du domaine de service.

Exigences et restrictions relatives au réseau virtuel sécurisé

Vous pouvez configurer un réseau virtuel sécurisé en exécutant les commandes `ldm add-vnet` et `ldm set-vnet` pour définir la propriété `custom=enable`. Notez que vous devez indiquer des valeurs pour les propriétés `custom/max-mac-addrs` et `custom/max-vlans` afin de garantir que le nombre d'adresses MAC et de VLAN personnalisés soit limité pour le périphérique de réseau virtuel spécifié. Ces deux valeurs de propriété sont, par défaut, définies sur 4096.

La fonctionnalité de réseau virtuel sécurisé requiert au minimum le SE Oracle Solaris 11.3 SRU 8.

Le domaine invité comportant le périphérique virtuel personnalisé et le domaine de service comportant le commutateur virtuel correspondant requièrent tous les deux la version la plus récente de ce microprogramme système.

Pour configurer un réseau virtuel sécurisé, vous devez indiquer les informations suivantes :

- `custom` – Permet d'activer ou de désactiver la fonctionnalité de réseau virtuel sécurisé. Cette fonctionnalité permet à une entité sécurisée d'ajouter des ID VLAN personnalisés et des adresses MAC alternatives personnalisées en mode dynamique.
- `custom/max-mac-addrs` – Permet d'indiquer le nombre maximal d'adresses MAC alternatives personnalisées à configurer sur un périphérique de réseau virtuel sécurisé donné.
- `custom/max-vlans` – Permet d'indiquer le nombre maximal d'ID VLAN alternatifs personnalisés à configurer sur un périphérique de réseau virtuel sécurisé donné.

Les restrictions suivantes s'appliquent à la fonctionnalité de réseau virtuel sécurisé :

- Vous ne pouvez pas utiliser le Logical Domains Manager pour configurer des adresses MAC ou ID VLAN alternatifs sur un réseau virtuel sécurisé donné.
- Pour modifier des adresses MAC alternatives personnalisées ou existantes, le domaine doit être lié.
- Vous pouvez augmenter les valeurs des propriétés `custom/max-mac-addrs` et `custom/max-vlans`, de façon dynamique. En revanche, pour que vous puissiez réduire ces valeurs, le domaine doit être lié.

Remarque - Une réduction des valeurs de ces propriétés peut avoir des effets indésirables. Par conséquent, veillez à supprimer tout VNIC ou VLAN créé sur l'hôte dont vous n'avez pas besoin car vous ne contrôlez absolument pas les adresses MAC ou les ID de VLAN que le système d'exploitation va retenir.

En outre, définissez `custom=disable` sur le périphérique réseau virtuel avant d'exécuter la commande `ldm set-vnet` afin de réduire le nombre maximum d'ID de VLAN et d'adresses MAC pour le périphérique réseau virtuel personnalisé.

Attention - Cette fonctionnalité a pour objet de limiter et de contrôler ces propriétés.

- Veillez à supprimer les périphériques VNIC et VLAN qui ont été créés avant de réduire le nombre des ID VLAN personnalisés ou des adresses MAC alternatives personnalisées. Autrement, le domaine invité comportera des VNIC qui ne pourront pas être configurés et devront être supprimés manuellement.
- La commande `dladm show-vnic -m` affiche les adresses MAC et ID VLAN qui sont configurés sur le réseau virtuel indiqué. La commande `dladm show-vnic -m` affiche les adresses MAC et ID VLAN alternatifs en cours d'utilisation sur le domaine invité. Cela provient des anciennes versions où tous les adresses MAC et ID VLAN alternatifs étaient préconfigurés sur le commutateur virtuel.

- La fonctionnalité de réseau virtuel sécurisé et la fonctionnalité PVLAN s'excluent mutuellement.
- Le Logical Domains Manager tente de valider la prise en charge du domaine invité et du domaine de service pour cette fonctionnalité avant d'activer la fonctionnalité personnalisée. Si le domaine invité n'est pas en cours d'exécution, vous pouvez activer cette fonctionnalité si le domaine de service la prend en charge. En revanche, si le domaine invité ne prend pas en charge la fonctionnalité, vous devez définir `custom=disabled` avant de pouvoir réactiver les adresses MAC et ID VLAN alternatifs non personnalisés.
- Vous pouvez effectuer une migration en direct d'un domaine avec des réseaux virtuels sécurisés uniquement si le domaine de service cible prend en charge la fonctionnalité de réseau virtuel sécurisé.

Configuration des réseaux virtuels sécurisés

Cette section décrit les tâches expliquant comment créer des réseaux virtuels sécurisés et obtenir des informations les concernant.

Vous pouvez configurer un réseau virtuel sécurisé en définissant la valeur de la propriété `custom` à l'aide de la commande `ldm add-vnet` ou `ldm set-vnet`. Reportez-vous à la page de manuel `ldm`(1M).

EXEMPLE 48 Création d'un réseau virtuel sécurisé

Vous pouvez exécuter les commandes suivantes pour créer un réseau virtuel sécurisé `ldg1_vnet0` sur le commutateur virtuel `primary-vsw0` dans le domaine `ldg1`. Les valeurs des propriétés `custom/max-mac-addrs` et `custom/max-vlans` sont définies par défaut sur 4096.

```
primary# ldm add-vnet custom=enable ldg1_vnet0 primary-vsw0 ldg1
primary# ldm list -o network ldg1
...
NETWORK
    NAME         SERVICE                MACADDRESS          PVID|PVLAN|VIDs
    ----         -------                ----------          ---------------
    ldg1-vnet0   primary-vsw0@primary   00:14:4f:fa:d7:5e   1|--|--
            DEVICE     :network@1       ID   :1
            LINKPROP   :phys-state      MTU  :1500
            MAXBW      :--              MODE :--
            CUSTOM     :enable
            MAX-CUSTOM-MACS:4096        MAX-CUSTOM-VLANS:4096
            PRIORITY   :--              COS  :--
            PROTECTION :--
```

EXEMPLE 49 Activation de la fonctionnalité de réseau virtuel sécurisé sur un réseau virtuel existant

L'exemple suivant montre comment activer la fonctionnalité de réseau virtuel sécurisé en définissant `custom=enable` pour le périphérique de réseau virtuel `ldg1_vnet0` dans le domaine

`ldg1`. Les valeurs des propriétés `custom/max-mac-addrs` et `custom/max-vlans` sont définies par défaut sur 4096.

```
primary# ldm set-vnet custom=enabled ldg1_vnet0 ldg1
primary# ldm list -o network ldg1
...
NETWORK
    NAME         SERVICE                MACADDRESS          PVID|PVLAN|VIDs
    ----         -------                ----------          ---------------
    ldg1-vnet0   primary-vsw0@primary   00:14:4f:fa:d7:5e   1|--|--
            DEVICE     :network@1       ID   :1
            LINKPROP   :phys-state      MTU  :1500
            MAXBW      :--              MODE :--
            CUSTOM     :enable
            MAX-CUSTOM-MACS:4096        MAX-CUSTOM-VLANS:4096
            PRIORITY   :--              COS  :--
            PROTECTION :--
```

EXEMPLE 50 Définition des propriétés `custom/max-mac-addrs` et `custom/max-vlans`

L'exemple suivant définit la valeur de la propriété `custom/max-vlans` sur 12 et celle de `custom/max-mac-addrs` sur 13.

Comme ces nouvelles valeurs de propriété sont inférieures aux valeurs précédentes, vous ne pouvez pas modifier ces paramètres de façon dynamique. Vous ne pouvez apporter ces modifications que pour un domaine lié ou inactif.

```
primary# ldm stop ldg1
primary# ldm set-vnet custom/max-vlans=12 custom/max-mac-addrs=13 ldg1_vnet0 ldg1
primary# ldm list -o network ldg1
...
NETWORK
    NAME         SERVICE                MACADDRESS          PVID|PVLAN|VIDs
    ----         -------                ----------          ---------------
    ldg1-vnet0   primary-vsw0@primary   00:14:4f:fa:d7:5e   1|--|--
            DEVICE     :network@1       ID   :1
            LINKPROP   :phys-state      MTU  :1500
            MAXBW      :--              MODE :--
            CUSTOM     :enable
            MAX-CUSTOM-MACS:13          MAX-CUSTOM-VLANS:12
            PRIORITY   :--              COS  :--
            PROTECTION :--
```

EXEMPLE 51 Réinitialisation des propriétés `custom/max-mac-addrs` et `custom/max-vlans`

L'exemple suivant montre comment réinitialiser la valeur de la propriété `custom/max-vlans` sur sa valeur par défaut, soit 4096, en spécifiant une valeur NULL.

Quand `custom=enabled`, vous pouvez réinitialiser la valeur de la propriété `custom/max-vlans`, la valeur de la propriété `custom/max-mac-addrs`, ou les deux.

```
primary# ldm set-vnet custom/max-mac-addrs= ldg1_vnet0 ldg1
primary# ldm list -o network ldg1
...
```

```
NETWORK
    NAME         SERVICE                 MACADDRESS          PVID|PVLAN|VIDs
    ----         -------                 ----------          ---------------
    ldg1-vnet0   primary-vsw0@primary    00:14:4f:fa:d7:5e   1|--|--
            DEVICE     :network@1        ID   :1
            LINKPROP   :phys-state       MTU  :1500
            MAXBW      :--               MODE :--
            CUSTOM     :enable
            MAX-CUSTOM-MACS:4096           MAX-CUSTOM-VLANS:12
            PRIORITY   :--               COS  :--
            PROTECTION :--
```

EXEMPLE 52 Modification des valeurs des propriétés `custom/max-mac-addrs` et `custom/max-vlans`

L'exemple suivant montre comment augmenter la valeur de la propriété `custom/max-vlans` et réduire celle de `custom/max-mac-addrs`. Vous pouvez augmenter la valeur de la propriété `custom/max-vlans` à 24, de façon dynamique, car la valeur 24 est supérieure à la précédente qui était 12. Cependant, comme vous réduisez la valeur maximale pour `custom/max-mac-addrs` de 4096 à 11, vous devez d'abord arrêter le domaine.

```
primary# ldm set-vnet custom/max-vlans=24 ldg1_vnet0 ldg1
primary# ldm stop ldg1
primary# ldm set-vnet custom/max-mac-addrs=11 ldg1_vnet0 ldg1
primary# ldm list -o network ldg1
...
NETWORK
    NAME         SERVICE                 MACADDRESS          PVID|PVLAN|VIDs
    ----         -------                 ----------          ---------------
    ldg1-vnet0   primary-vsw0@primary    00:14:4f:fa:d7:5e   1|--|--
            DEVICE     :network@1        ID   :1
            LINKPROP   :phys-state       MTU  :1500
            MAXBW      :--               MODE :--
            CUSTOM     :enable
            MAX-CUSTOM-MACS:11           MAX-CUSTOM-VLANS:24
            PRIORITY   :--               COS  :--
            PROTECTION :--
```

EXEMPLE 53 Désactivation de la fonctionnalité de réseau virtuel sécurisé

L'exemple suivant montre comment désactiver la propriété `custom` pour le périphérique de réseau virtuel `ldg1_vnet0` dans le domaine `ldg1`.

```
primary# ldm set-vnet custom=disabled ldg1_vnet0 ldg1
...
NETWORK
    NAME         SERVICE                 MACADDRESS          PVID|PVLAN|VIDs
    ----         -------                 ----------          ---------------
    ldg1-vnet0   primary-vsw0@primary    00:14:4f:fa:d7:5e   1|--|--
            DEVICE     :network@1        ID   :1
            LINKPROP   :phys-state       MTU  :1500
            MAXBW      :--               MODE :--
            CUSTOM     :disable
            PRIORITY   :--               COS  :--
            PROTECTION :--
```

Affichage des informations de réseau virtuel sécurisé

Vous pouvez obtenir des informations sur les paramètres de réseau virtuel sécurisé à l'aide de plusieurs sous-commandes list du Logical Domains Manager. Reportez-vous à la page de manuel `ldm`(1M).

Les exemples suivants utilisent les commandes `ldm list-domain -o network`, `ldm list-bindings` et `ldm list-constraints` pour afficher des informations sur une configuration de réseau virtuel sécurisé.

- L'exemple suivant montre comment utiliser la commande `ldm list-domain` pour afficher des informations sur une configuration de réseau virtuel sécurisé pour le domaine `ldg1` :

```
primary# ldm list-domain -o network ldg1
...
NETWORK
    NAME         SERVICE                MACADDRESS          PVID|PVLAN|VIDs
    ----         -------                ----------          ---------------
    ldg1-vnet0   primary-vsw0@primary   00:14:4f:fa:d7:5e   1|--|--
            DEVICE     :network@1       ID   :1
            LINKPROP   :phys-state      MTU  :1500
            MAXBW      :--              MODE :--
            CUSTOM     :enable
            MAX-CUSTOM-MACS:11          MAX-CUSTOM-VLANS:24
            PRIORITY   :--              COS  :--
            PROTECTION :--
```

- L'exemple suivant montre comment utiliser la commande `ldm list-domain` pour afficher des informations sur une configuration de réseau virtuel sécurisé, sous une forme analysable, pour le domaine `ldg1` :

```
primary# ldm list-domain -o network -p ldg1
VERSION 1.19
DOMAIN|name=ldg1|
MAC|mac-addr=00:14:4f:f9:4b:d0
VNET|name=ldg1-vnet0|dev=network@1|service=primary-vsw0@primary|mac-addr=00:14:4f:fa:
d7:5e|mode=|pvid=1|vid=|mtu=1500|linkprop=phys-state|id=1|alt-mac-addrs=|maxbw=|pvlan=|
protection=|priority=|cos=|custom=enable|max-mac-addrs=11|max-vlans=24
```

- L'exemple suivant montre comment utiliser la commande `ldm list-bindings` pour afficher des informations sur une configuration de réseau virtuel sécurisé pour le domaine `ldg1` :

```
primary# ldm list-bindings -e -o network ldg1
...
NETWORK
```

```
NAME         SERVICE                 MACADDRESS          PVID|PVLAN|VIDs
----         -------                 ----------          ---------------
ldg1-vnet0   primary-vsw0@primary    00:14:4f:fa:d7:5e   1|--|--
        DEVICE     :network@1        ID   :1
        LINKPROP   :phys-state       MTU  :1500
        MAXBW      :--               MODE :--
        CUSTOM     :enable
        MAX-CUSTOM-MACS:11           MAX-CUSTOM-VLANS:24
        PRIORITY   :--               COS  :--
        PROTECTION :--

    PEER                    MACADDRESS          PVID|PVLAN|VIDs
    ----                    ----------          ---------------
    primary-vsw0@primary    00:14:4f:f9:08:28   1|--|--
        LINKPROP   :--               MTU  :1500
        MAXBW      :--               LDC  :0x5
        MODE       :--
```

- L'exemple suivant montre comment utiliser la commande `ldm list-bindings` pour afficher des informations sur une configuration de réseau virtuel sécurisé, sous une forme analysable, pour le domaine `ldg1` :

```
primary# ldm list-bindings -p ldg1
...
VNET|name=ldg1-vnet0|dev=network@1|service=primary-vsw0@primary|mac-addr=00:14:4f:fa:
d7:5e|mode=|pvid=1|vid=|mtu=1500|linkprop=phys-state|id=1|alt-mac-addrs=|maxbw=|pvlan=|
protection=|priority=|cos=|custom=enable|max-mac-addrs=11|max-vlans=24
|peer=primary-vsw0@primary|mac-addr=00:14:4f:f9:08:28|mode=|pvid=1|vid=|mtu=1500|maxbw=
```

- L'exemple suivant montre comment générer un XML en exécutant la commande `ldm list-constraints -x` :

```
primary# ldm list-constraints -x ldg1
...
<Section xsi:type="ovf:VirtualHardwareSection_Type">
  <Item>
    <rasd:OtherResourceType>network</rasd:OtherResourceType>
    <rasd:Address>auto-allocated</rasd:Address>
    <gprop:GenericProperty key="vnet_name">ldg1-vnet0</gprop:GenericProperty>
    <gprop:GenericProperty key="service_name">primary-vsw0</gprop:GenericProperty>
    <gprop:GenericProperty key="pvid">1</gprop:GenericProperty>
    <gprop:GenericProperty key="linkprop">phys-state</gprop:GenericProperty>
    <gprop:GenericProperty key="custom">enable</gprop:GenericProperty>
    <gprop:GenericProperty key="max-mac-addrs">11</gprop:GenericProperty>
    <gprop:GenericProperty key="max-vlans">24</gprop:GenericProperty>
    <gprop:GenericProperty key="device">network@1</gprop:GenericProperty>
```

```
    <gprop:GenericProperty key="id">1</gprop:GenericProperty>
  </Item>
</Section>
```

Différences liées aux fonctions de gestion réseau Oracle Solaris 11

Certaines des fonctionnalités de gestion de réseau d'Oracle VM Server for SPARC fonctionnent de manière différente selon qu'un domaine exécute le système d'exploitation Oracle Solaris 10 ou Oracle Solaris 11. Voici les différences fonctionnelles entre le périphérique réseau virtuel et le commutateur virtuel Oracle VM Server for SPARC lorsque le SE Oracle Solaris 11 est exécuté dans un domaine :

- **Configuration du périphérique `vsw`*n* en tant qu'interface réseau principale afin de permettre à un domaine de service de communiquer avec des domaines invités**

  Cette configuration est uniquement requise pour les domaines qui exécutent le SE Oracle Solaris 10. Dans Oracle Solaris 11, un commutateur virtuel utilise la pile réseau Oracle Solaris 11, ce qui permet automatiquement à ses périphériques réseau virtuels de communiquer avec l'interface réseau correspondant à son périphérique backend, telle que `net0` par exemple. La configuration du périphérique `vsw`*n* en tant qu'interface réseau sur Oracle Solaris 11 n'est pas prise en charge.
- **Utilisation d'un périphérique `etherstub` d'Oracle Solaris 11 en tant que périphérique backend pour créer un commutateur virtuel privé**

  S'il n'est pas connecté à un périphérique backend, un commutateur virtuel permet la communication entre des domaines invités uniquement et non entre les domaines invités et au domaine de service. L'utilisation d'un `etherstub` comme périphérique backend permet au domaine invité de communiquer avec une zone (y compris la zone globale) configurée dans un domaine de service Oracle Solaris 11. Cette configuration s'effectue à l'aide d'un VNIC connecté à ce `etherstub`.
- **Utilisation de noms génériques pour le commutateur virtuel et les périphériques réseau virtuels**

  Le SE Oracle Solaris 11 attribue des noms génériques pour les périphériques `vsw`*n* et `vnet`*n*. Prenez donc garde de ne pas créer de commutateur virtuel avec un périphérique backend qui serait un autre périphérique `vsw` ou `vnet`. Utilisez la commande `dladm show-phys` pour afficher les périphériques physiques réels associés à des noms de périphériques réseau génériques.
- **Utilisation d'une VNIC Oracle Solaris 11 pour créer un VLAN sur un stub Ethernet**

  Ne configurez pas les VLAN sur l'interface du commutateur virtuel pour les domaines de service Oracle Solaris 11 car cette configuration n'est pas prise en charge. A la place, créez

le VLAN sur l'interface correspondant à la valeur de la propriété `net-dev` du commutateur virtuel.

Pour Oracle Solaris 10, vous pouvez définir la propriété `net-dev` avec une valeur NULL pour créer un commutateur virtuel routé. En revanche, cette méthode n'est pas prise en charge pour Oracle Solaris 11. A la place, configurez la VNIC sur le périphérique de stub Ethernet comme partie du VLAN.

L'exemple suivant montre comment créer des VNIC sur un stub Ethernet. La commande `dladm create-etherstub` crée un stub Ethernet, `estub100`, qui est un périphérique de sauvegarde utilisé par la commande `ldm add-vsw` pour créer le commutateur virtuel. La commande `ldm add-vsw` crée le commutateur virtuel. La commande `dladm create-vnic` crée une VNIC au-dessus de l'etherstub pour créer le VLAN pour ce commutateur virtuel.

```
primary# dladm create-etherstub estub100
primary# ldm add-vsw net-dev=estub100 vid=100 inter-vnet-link=off \
primary-vsw100 primary
primary# dladm create-vnic -l estub100 -m auto -v 100 vnic100
```

La commande `ldm add-vnet` suivante crée deux VNIC activant la communication entre les domaines `ldg1` et `ldg2` sur le VLAN 100.

```
primary# ldm add-vnet vid=100 ldg1-vnet100 primary-vsw100 ldg1
primary# ldm add-vnet vid=100 ldg2-vnet100 primary-vsw100 ldg2
```

Dans l'exemple suivant, les commandes `dladm` créent des VLAN sur les domaines invités `ldg1` et `ldg2`. La commande `ipadm` crée des adresses IP pour les VNIC que vous avez créées sur les domaines `ldg1` et `ldg2`.

```
ldg1# dladm create-vlan -l net1 -v 100 vlan100
ldg1# ipadm create-ip vlan100
ldg1# ipadm create-ipaddr -T static -a 192.168.100.10/24 vlan100/v4
ldg2# dladm create-vlan -l net1 -v 100 vlan100
ldg2# ipadm create-ip vlan100
ldg2# ipadm create-ipaddr -T static -a 192.168.100.20/24 vlan100/v4
```

- **Utilisation de noms génériques pour le commutateur virtuel et les périphériques réseau virtuels**

  Le SE Oracle Solaris 11 attribue des noms génériques pour les périphériques `vsw`*n* et `vnet`*n*. Prenez donc garde de ne pas créer de commutateur virtuel avec un périphérique backend qui serait un autre périphérique `vsw` ou `vnet`. Utilisez la commande `dladm show-phys` pour afficher les périphériques physiques réels associés à des noms de périphériques réseau génériques.

- **Utilisation de VNIC sur les périphériques commutateur virtuel et réseau virtuel**

  Vous *ne pouvez pas* utiliser de VNIC sur les périphériques `vsw`*n*. Une tentative de création d'une VNIC sur `vsw`*n* a échoué.

- **Utilisation des commandes d'observabilité du réseau sur les domaines invités Oracle Solaris 11**

Vous pouvez utiliser les commandes `ldm list-netdev` et `ldm list-netstat` pour obtenir des informations sur les domaines invités Oracle Solaris 11.

♦ ♦ ♦ CHAPITRE 14

Migration des domaines

Ce chapitre décrit la procédure de migration des domaines d'une machine hôte à une autre.

Ce chapitre aborde les sujets suivants :

Remarque - Pour utiliser les fonctionnalités de migration décrites dans ce chapitre, vous devez exécutez les versions les plus récentes de Logical Domains Manager, du microprogramme du système et du SE Oracle Solaris. Pour plus d'informations sur la migration à l'aide de versions antérieures d'Oracle VM Server for SPARC, reportez-vous aux *Notes de version d'Oracle VM Server for SPARC 3.4* et aux versions appropriées du guide d'administration.

Introduction à la migration de domaines

La migration de domaines vous permet de migrer un domaine invité d'une machine hôte à une autre. La machine sur laquelle la migration est lancée est la *machine source*. La machine vers laquelle le domaine est migrée est la *machine cible*.

Au cours d'une opération de migration, le *domaine à migrer* est transféré de la machine source vers le *domaine migré* sur la machine cible.

La fonction *migration en direct* apporte une amélioration des performances permettant la migration d'un domaine actif en cours d'exécution. Outre la migration en direct, vous pouvez migrer des domaines liés ou inactifs. Il s'agit de la *migration à froid*.

Vous pouvez utiliser la migration de domaines pour effectuer les tâches suivantes, entre autres :

- Equilibrage des charges entre des machines
- Exécution de la maintenance matérielle alors qu'un domaine invité est toujours en cours d'exécution

Pour des performances de migration optimales, assurez-vous que la machine source et la machine cible exécutent la dernière version en date du Logical Domains Manager.

Présentation d'une opération de migration

Logical Domains Manager sur la machine source accepte les demandes de migration d'un domaine et établit une connexion réseau sécurisée avec Logical Domains Manager s'exécutant sur la machine cible. La migration se produit une fois la connexion établie. L'opération de migration s'exécute selon les phases suivantes :

Phase 1 : une fois la machine source connectée avec Logical Domains Manager en cours d'exécution sur la machine cible, les informations relatives à la machine source et le domaine à migrer sont transférés vers la machine cible. Ces informations sont utilisées pour effectuer une série de contrôles pour déterminer si une migration est possible. Les contrôles à effectuer sont basés sur l'état du domaine à migrer. Par exemple, si le domaine à migrer est actif, un jeu de contrôles différent de celui appliqué à un domaine lié ou inactif est réalisé.

Phase 2 : une fois tous les contrôles de la phase 1 effectués, les machines source et cible se préparent pour la migration. Sur la machine cible, un domaine est créé pour recevoir le domaine à migrer. Si le domaine à migrer est inactif ou lié, l'opération de migration passe à la phase 5.

Phase 3 : si le domaine à migrer est actif, ses informations d'état d'exécution sont transférées vers la machine cible. Le domaine à migrer reste en cours d'exécution, et Logical Domains

Manager recherche simultanément les modifications effectuées par le SE sur ce domaine. Ces informations sont récupérées à partir de l'hyperviseur sur la machine source et installées dans l'hyperviseur de la machine cible.

Phase 4 : le domaine à migrer est suspendu. A ce moment-là, toutes les informations d'état modifié restantes sont de nouveau copiées sur la machine cible. De cette manière, l'interruption du domaine est infime ou imperceptible. La quantité d'interruption dépend de la charge de travail.

Phase 5 : le transfert se produit de Logical Domains Manager sur la machine source vers Logical Domains Manager sur la machine cible. Le transfert se produit lorsque le domaine migré reprend son exécution (si le domaine à migrer était actif). Le domaine sur la machine source est ensuite détruit. A partir de ce moment-là, le domaine migré est la seule version du domaine en cours d'exécution.

Compatibilité logicielle

Pour qu'une migration ait lieu, les machines source et cible doivent exécuter un logiciel compatible, comme suit :

- La version de Logical Domains Manager exécutée sur les deux machines doit être la version actuelle ou la version antérieure la plus récente.
- Les machines source et cible doivent posséder une version compatible du microprogramme installé afin de prendre en charge la migration en direct. Les deux machines doivent au moins exécuter la version minimum du microprogramme pris en charge par cette version du logiciel Oracle VM Server for SPARC.

Pour plus d'informations, reportez-vous à la section "Restrictions de version pour la migration" à la page 328.

Sécurité pour les opérations de migration

Oracle VM Server for SPARC offre les fonctions de sécurité suivantes pour les opérations de migration :

- **Authentification.** L'opération de migration s'effectuant sur deux machines, dans certains cas, l'utilisateur doit être authentifié sur les machines source et cible. Plus précisément, un utilisateur autre que le superutilisateur doit utiliser le profil de droits `LDoms Management` (Gestion de domaines logiques). Cependant, si vous effectuez une migration avec les

certificats SSL, les utilisateurs n'ont pas besoin d'être authentifiés sur les ordinateurs cible et source et vous ne pouvez pas indiquer un autre utilisateur.

La commande `ldm migrate-domain` vous permet éventuellement de spécifier un nom d'utilisateur alternatif pour l'authentification sur la machine cible. Si cela n'est pas fait, le nom d'utilisateur de l'utilisateur exécutant la commande de migration est utilisé. Reportez-vous à la section Exemple 56, "Migration et renommage d'un domaine invité". Dans les deux cas, l'utilisateur est invité à entrer un mot de passe pour la machine cible, à moins que l'option `-p` ne soit utilisée pour initier une migration non interactive. Reportez-vous à la section "Réalisation de migrations non interactives" à la page 331.

- **Chiffrement.** Oracle VM Server for SPARC utilise SSL pour chiffrer le trafic de migration afin de protéger les données sensibles contre tout exploitation et d'éliminer les exigences de matériel supplémentaire et de réseaux dédiés.

  Sur les plates-formes équipées d'unités cryptographiques, la vitesse de l'opération de migration augmente lorsque le domaine `primary` sur les machines source et cible dispose d'unités cryptographiques qui lui sont assignées. Cette augmentation se produit car les opérations SSL peuvent être déchargées sur les unités cryptographiques

  La vitesse d'une opération de migration est automatiquement améliorée sur les plates-formes dont la CPU contient des instructions cryptographiques. Cette amélioration est due au fait que les opérations SSL peuvent être exécutées par ces instructions cryptographiques plutôt que dans un logiciel.
- **FIPS 140-2.** Vous pouvez configurer votre système pour effectuer les migrations de domaines de sorte à utiliser les bibliothèques certifiées OpenSSL FIPS 140-2 Oracle Solaris. Voir "Mode FIPS 140-2 pour la migration de domaine" à la page 326.

Configuration des certificats SSL pour la migration

Pour effectuer l'authentification par certificat, utilisez l'option `-c` avec la commande`ldm migrate-domain`. Cette option est mutuellement exclusive avec le fichier de mots de passe et les autres options utilisateur. Si l'option `-c` n'est pas spécifiée, l'opération de migration effectue l'authentification par mot de passe.

▼ Procédure de configuration des certificats SSL pour la migration

Pour configurer les certificats SSL, suivez les étapes ci-après sur le domaine de contrôle de l'ordinateur source.

1. **Créez le répertoire `/var/share/ldomsmanager/trust` s'il n'existe pas déjà.**

   ```
   src-primary# mkdir /var/share/ldomsmanager/trust
   ```

2. **Copiez le certificat `ldmd` du serveur cible vers le répertoire du certificat de confiance local.**

   Le certificat distant `ldmd` correspond à `/var/share/ldomsmanager/server.crt` sur l'hôte distant. Le répertoire du certificat de confiance local `ldmd` est `/var/share/ldomsmanager/trust`. Renommez le fichier de certificat distant *target-hostname*`.pem`, par exemple `tgt-primary.pem`.

3. **Créez un lien symbolique à partir du certificat dans le répertoire du certificat de confiance vers le répertoire `/etc/certs/CA`.**

   Définissez la variable `REMOTE` sur le nom d'hôte du serveur cible pointant vers le certificat du serveur cible, `tgt-primary.pem`.

   ```
   src-primary# ln -s /var/share/ldomsmanager/trust/tgt-primary.pem /etc/certs/CA/
   ```

4. **Redémarrez le service `svc:/system/ca-certificates`.**

   ```
   src-primary# svcadm restart svc:/system/ca-certificates
   ```

5. **Vérifiez que la configuration est correcte.**

   ```
   src-primary# openssl verify /var/share/ldomsmanager/trust/tgt-primary.pem
   /var/share/ldomsmanager/trust/tgt-primary.pem: ok
   ```

6. **Assurez-vous que le service `ca-certificates` est en ligne.**

   Redémarrez ou activez le service si nécessaire.

   ```
   src-primary# svcs ca-certificates
   /var/share/ldomsmanager/trust/tgt-primary.pem: ok
   STATE        STIME    FMRI
   online       0:22:38  svc:/system/ca-certificates:default
   ```

7. **Redémarrez le démon `ldmd`.**

   ```
   src-primary# svcadm restart ldmd
   ```

8. **Répétez ces étapes sur le serveur cible.**

Suppression du certificat SSL

Si vous enlevez un fichier `.pem` des répertoires `/var/share/ldomsmanager/trust` et `/etc/certs/CA`, vous devez redémarrer les services `svc:/system/ca-certificates` puis `ldmd`. Notez que les migrations qui utilisent ce fichier `.pem` sont autorisées jusqu'au redémarrage du service.

```
localhost# svcadm restart svc:/system/ca-certificates
localhost# svcadm restart ldmd
```

Mode FIPS 140-2 pour la migration de domaine

Vous pouvez configurer votre Logical Domains Manager pour effectuer les migrations de domaines de sorte à utiliser les bibliothèques certifiées OpenSSL FIPS 140-2 Oracle Solaris. Lorsque le Logical Domains Manager est en mode FIPS 140-2, vous pouvez ne l'utiliser que pour migrer des domaines vers un autre système dont le Logical Domains Manager exécute le mode FIPS 140-2. Les tentatives d'effectuer la migration vers un système non FIPS sont rejetées. Si le Logical Domains Manager n'est pas en mode FIPS 140-2, il ne peut pas migrer vers un Logical Domains Manager en mode FIPS 140-2.

Pour démarrer le Logical Domains Manager en mode FIPS 140-2, vous devez activer le médiateur FIPS. Pour consulter des instructions détaillées, reportez-vous à la section "Procédure d'exécution de Logical Domains Manager en mode FIPS 140-2" à la page 326.

Pour plus d'informations et d'exemples sur l'utilisation de l'implémentation OpenSSL compatible FIPS 140, reportez-vous aux sections "How to Switch to the FIPS 140-Capable OpenSSL Implementation" du manuel *Managing Encryption and Certificates in Oracle Solaris 11.3* et "Example of Enabling Two Applications in FIPS 140 Mode on an Oracle Solaris System" du manuel *Oracle SuperCluster M7-8 Owner's Guide: Overview*.

▼ Procédure d'exécution de Logical Domains Manager en mode FIPS 140-2

Avant de commencer

Avant d'exécuter Logical Domains Manager en mode FIPS 140-2, assurez-vous que la version exécutée est au minimum la version 3.2 de Logical Domains Manager et que le domaine `primary` exécute au moins le SE Oracle Solaris 11.2.

1. **Installez et activez le médiateur OpenSS FIPS 140-2.**

   a. **Installez le médiateur OpenSS FIPS 140-2 si nécessaire.**

      Ce package doit s'installer par défaut lorsque vous installez le SE Oracle Solaris 11.2.

      ```
      # pkg install openssl-fips-140
      ```

   b. **Répertoriez le médiateur OpenSSL en cours.**

      ```
      # pkg mediator openssl
      MEDIATOR VER. SRC. VERSION IMPL. SRC. IMPLEMENTATION
      openssl  vendor           local      default
      ```

c. Répertoriez les médiateurs OpenSSL disponibles.

```
# pkg mediator -a openssl
MEDIATOR VER. SRC. VERSION IMPL. SRC. IMPLEMENTATION
openssl  vendor            vendor     default
openssl  system            system     fips-140
```

Attention - L'implémentation d'OpenSSL vers laquelle vous basculez doit exister dans le système. Si vous basculez vers une implémentation qui ne figure pas dans le système, le système risque de devenir inutilisable.

d. Activez le médiateur FIPS 140-2.

```
# pkg set-mediator -I fips-140 openssl
```

e. Redémarrez.

```
# reboot
```

f. Vérifiez que le médiateur FIPS 140-2 est bien défini.

```
# pkg mediator openssl
    MEDIATOR     VER. SRC. VERSION IMPL. SRC. IMPLEMENTATION
    openssl      system            local      fips-140
```

2. Configurez le démon `ldmd` pour utiliser le mode FIPS 140-2.

a. Placez le démon `ldmd` en mode FIPS 140-2.

```
# svccfg -s ldoms/ldmd setprop ldmd/fips1402_enabled = true
```

b. Redémarrez le démon `ldmd`.

```
# svcadm refresh ldmd
# svcadm restart ldmd
```

▼ Procédure de retour au mode par défaut depuis le mode FIPS 140-2 pour Logical Domains Manager

1. Arrêtez d'utiliser le médiateur OpenSSL FIPS 140-2 en rétablissant le médiateur OpenSSL par défaut.

Suivez cette étape uniquement si le médiateur FIPS 140-2 n'est pas requis par d'autres applications.

```
# pkg set-mediator -I default openssl
```

2. **Redémarrez.**

```
# reboot
```

3. **Configurez le démon `ldmd`de sorte à utiliser le mode par défaut.**

```
# svccfg -s ldoms/ldmd setprop ldmd/fips1402_enabled = false
```

4. **Redémarrez le démon `ldmd`.**

```
# svcadm refresh ldmd
# svcadm restart ldmd
```

Restrictions liées aux migrations de domaines

Les sections suivantes décrivent les restrictions qui s'appliquent à la migration de domaine. Les versions du microprogramme du système et du logiciel Logical Domains Manager doivent être compatibles pour que les migrations soient possibles. De même, la CPU doit répondre à certaines exigences pour que la migration de domaine fonctionne.

La migration en direct n'est pas qualifiée et prise en charge sur toutes les combinaisons de plates-formes source et cible et de versions de microprogramme système. Pour ces combinaisons qui ne peuvent pas effectuer une migration en direct, vous pouvez à la place effectuer une migration à froid.

Restrictions de version pour la migration

Cette section décrit les restrictions liées aux versions pour la réalisation de migrations en direct.

- **Version de Logical Domains Manager.** Vous pouvez effectuer une migration en direct dans n'importe quelle direction lorsque l'un des systèmes exécute la version la plus récente de Logical Domains Manager et l'autre système exécute l'avant-dernière version de Logical Domains Manager.
- **Version de SE Oracle Solaris.** Vous pouvez effectuer une migration en direct d'un domaine invité qui exécute au minimum le SE Oracle Solaris 10 9/10. Vous *ne pouvez pas* effectuer de migration en direct d'un domaine invité exécutant le SE Oracle Solaris 10 10/09 ou les versions d'SE Oracle Solaris antérieures. Vous pouvez toujours initialiser ces anciennes versions de SE Oracle Solaris et effectuer des migrations à froid desdits domaines.
- **Version du microprogramme système.** En général, vous pouvez effectuer une migration en direct entre deux systèmes lorsque la machine source et la machine cible prennent toutes deux en charge les versions minimales appropriées du microprogramme système. Reportez-

vous à la section "Versions minimales du microprogramme système" du manuel *Guide d'installation d'Oracle VM Server for SPARC 3.4*.

Restrictions pour la migration entre CPU

Vous ne pouvez pas effectuer d'opérations de migration en direct entre un serveur UltraSPARC T2, UltraSPARC T2 Plus ou SPARC T3 et des serveurs de série SPARC T5 ou ultérieure.

Restrictions liées aux migrations pour le paramétrage des `compteurs de performances`

Portez une attention particulière lors de la migration de domaines ayant la valeur de propriété `perf-counters`.

Avant de réaliser la migration d'un domaine dont la valeur de propriété est `perf-counters` sur `global`, assurez-vous qu'aucun autre domaine de la machine cible n'ait la propriété `perf-counters` définie sur `global`.

Durant une opération de migration, la propriété `perf-counters` est traitée différemment en fonction de la disponibilité des capacités d'accès aux performances sur la machine source, la machine cible, ou les deux.

La valeur de la propriété `perf-counters` est traitée comme suit :

- **Machine source uniquement.** La valeur de la propriété `perf-counters` n'est pas étendue à la machine cible.
- **Machine source uniquement.** La valeur de propriété `perf-counters` sur la machine à migrer est mise à jour de sorte à correspondre à `perf-counters=`.
- **Machines source et cible.** La valeur de la propriété `perf-counters` est étendue depuis le domaine à migrer vers le domaine migré de la machine cible.

Pour plus d'informations sur la propriété `perf-counters`, reportez-vous à la section "Utilisation des propriétés Perf-Counter" à la page 383 et à la page de manuel `ldm`(1M).

Restrictions de migration pour le paramètre `linkprop=phys-state`

Vous pouvez migrer un périphérique de réseau virtuel qui a un périphérique de sauvegarde NIC physique et a `linkprop=phys-state` pour un domaine cible qui n'a pas de NIC physique en tant

que périphérique de sauvegarde (`net-dev=`). Comme la contrainte `linkprop=phys-state` n'est pas impérative, si un domaine est migré vers une machine qui n'a pas de valeur `net-dev` disponible, la contrainte est préservée mais n'est pas satisfaite. La propriété `linkprop` est toujours conservée comme `phys-state` et l'état de la liaison du périphérique réseau est une liaison montante.

Restrictions de migration pour des domaines ayant un grand nombre de périphériques virtuels

Il arrive parfois que la migration d'un domaine comportant un grand nombre de périphériques virtuels provoque un temps de réponse plus long du domaine de contrôle sur la machine cible. Pendant ce temps, les commandes `ldm` semblent bloquées et les commandes SE Oracle Solaris standard mettent plus longtemps à s'exécuter que normalement.

Cette interruption est provoquée par les serveurs virtuels qui traitent le grand nombre de périphériques virtuels entrants associés au domaine migré. Une fois ce traitement terminé, le domaine de contrôle s'exécute de nouveau normalement et les éventuelles commandes `ldm` bloquées aboutissent.

Vous pouvez minimiser ce type d'interruption en limitant le nombre de périphériques virtuels utilisés par un domaine à moins de 1 000.

Migration d'un domaine

Vous pouvez utiliser la commande `ldm migrate-domain` pour démarrer la migration d'un domaine d'une machine hôte à une autre.

Remarque - Si vous migrez un domaine, les ressources nommées que vous avez affectées à l'aide des propriétés `cid` et `mblock` sont supprimées. Le domaine utilise à la place les ressources anonymes sur le système cible.

Pour plus d'informations sur la migration d'un domaine actif en cours d'exécution, reportez-vous à la section "Migration d'un domaine actif" à la page 332. Pour plus d'informations sur la migration d'un domaine lié ou inactif, reportez-vous à la section "Migration de domaines liés ou inactifs" à la page 339.

Pour plus d'informations sur les options et les opérandes de migration, reportez-vous à la page de manuel `ldm`(1M).

Remarque - Lorsqu'une migration de domaine se termine, enregistrez une nouvelle configuration sur le SP des systèmes source et cible. En conséquence, l'état du domaine migré est correct si le système source ou le système cible est arrêté et redémarré.

Réalisation d'une simulation

Lorsque vous indiquez l'option `-n` dans la commande `ldm migrate-domain`, des contrôles de migration sont effectués, mais le domaine n'est pas migré. Toute exigence non remplie est signalée comme une erreur. Les résultats de la simulation vous permettent de corriger les erreurs de configuration avant de tenter la migration réelle.

Remarque - En raison de la nature dynamique des domaines logiques, une simulation peut aboutir et une migration réelle échouer, ou inversement.

Réalisation de migrations non interactives

Utilisez la méthode de certificat SSL pour effectuer des opérations de migration non-interactives. Bien que la commande existante `ldm migrate-domain -p` *filename* soit obsolète pour initier une migration non interactive, vous pouvez toujours l'utiliser.

Le nom de fichier que vous indiquez en tant qu'argument de l'option `-p` doit présenter les caractéristiques suivantes :

- La première ligne du fichier doit contenir le mot de passe.
- Le mot de passe doit être en texte brut.
- Le mot de passe ne doit pas dépasser une longueur de 256 caractères.

Les caractères de nouvelle ligne à la fin du mot de passe et de toutes les lignes suivantes sont ignorés.

Le fichier dans lequel vous stockez le mot de passe de la machine cible doit être correctement sécurisé. Si vous envisagez de stocker les mots de passe de cette manière, vérifiez que les autorisations du fichier sont définies de sorte que seul le propriétaire root ou un utilisateur doté de privilèges puissent accéder en lecture ou en écriture au fichier (400 ou 600).

Migration d'un domaine actif

Certaines exigences et restrictions sont imposées au domaine à migrer, à la machine source et à la machine cible lorsque vous tentez de migrer un domaine actif. Pour plus d'informations, reportez-vous à la section "Restrictions liées aux migrations de domaines" à la page 328.

Astuce - Vous pouvez réduire le temps total de migration en ajoutant d'autres CPU virtuelles au domaine `primary` sur les machines source et cible. Il est recommandé, mais pas obligatoire, de disposer d'au moins deux coeurs complets par domaine `primary`.

Un domaine "perd du temps" au cours du processus de migration. Pour limiter ce problème, synchronisez le domaine à migrer avec une source temporelle externe, un serveur NTP (Network Time Protocol, protocole d'heure réseau) par exemple. Lorsque vous configurez un domaine en tant que client NTP, la date et l'heure du domaine sont corrigés peu après la fin de la migration.

Pour configurer un domaine en tant que client NTP Oracle Solaris 10, reportez-vous à la section "Managing Network Time Protocol (Tasks)" du manuel *System Administration Guide: Network Services*. Pour configurer un domaine en tant que client NTP Oracle Solaris 11, reportez-vous à la section "Managing Network Time Protocol (Tasks)" du manuel *Introduction to Oracle Solaris 11 Network Services*.

Remarque - Au cours de la phase de suspension à la fin d'une migration, un domaine invité risque de subir un léger retard. Ce retard ne devrait pas entraîner d'interruption sensible des communications réseau, en particulier si le protocole inclut un mécanisme de nouvelle tentative, tel que TCP, ou si un mécanisme de nouvelle tentative existe au niveau de l'application, tel que NFS via UDP. Toutefois, si le domaine invité exécute une application sensible au réseau telle que RIP (Routing Information Protocol), le domaine peut subir un bref retard ou une courte interruption lors de l'exécution d'une opération. Cet éventuel retard se produit pendant le bref intervalle où l'interface réseau de l'invité est détruite puis recréée au cours de la phase de suspension.

Configuration requise des CPU pour la migration de domaines

Vous trouverez ci-dessous les contraintes et les limitations concernant les CPU lorsque vous effectuez une migration :

- La machine cible doit avoir suffisamment de CPU virtuelles libres pour accueillir le nombre de CPU virtuelles utilisées par le domaine à migrer.

- La définition de la propriété `cpu-arch` sur le domaine invité permet d'effectuer la migration de domaine entre des systèmes possédant des processeurs de type différent. Notez que le domaine invité doit se trouver dans un état lié ou inactif pour pouvoir modifier la valeur `cpu-arch`.

  Les valeurs suivantes sont prises en charge pour la propriété `cpu-arch` :

  - `native` utilise des fonctions matérielles spécifiques à la CPU pour permettre à un domaine invité de migrer *uniquement* entre des plates-formes de type de CPU identique. `native` est la valeur par défaut.
  - `migration-class1` est une famille de migration entre les CPU pour les plates-formes SPARC à partir de SPARC T4. Ces plates-formes prennent en charge la cryptographie matérielle pendant et après ces migrations afin de réduire la dépendance aux CPU prises en charge.

    Cette valeur n'est pas compatible avec les plates-formes UltraSPARC T2, UltraSPARC T2 Plus ou SPARC T3, ni les Plates-formes Fujitsu M10.
  - `sparc64-class1` est une famille de migration entre les CPU pour les plates-formes SPARC64. Etant donné que la valeur `sparc64-class1` est basée sur les instructions SPARC64, son nombre d'instructions est plus important que pour la valeur `generic`. Par conséquent, il n'y a aucun impact sur les performances par rapport à la valeur `generic`.

    Cette valeur est uniquement compatible avec les Serveurs Fujitsu M10.
  - `generic` utilise les fonctions matérielles de CPU constituant le plus petit dénominateur commun et utilisées par toutes les plates-formes pour permettre à un domaine invité d'exécuter une migration indépendante du type de CPU.

  Les commandes `isainfo -v` suivantes indiquent les instructions disponibles sur un système lorsque `cpu-arch=generic` et lorsque `cpu-arch=migration-class1`.

  - `cpu-arch=generic`

    ```
    # isainfo -v
    64-bit sparcv9 applications
            asi_blk_init vis2 vis popc
    32-bit sparc applications
            asi_blk_init vis2 vis popc v8plus div32 mul32
    ```

  - `cpu-arch=migration-class1`

    ```
    # isainfo -v
    64-bit sparcv9 applications
            crc32c cbcond pause mont mpmul sha512 sha256 sha1 md5
            camellia des aes ima hpc vis3 fmaf asi_blk_init vis2
            vis popc
    32-bit sparc applications
            crc32c cbcond pause mont mpmul sha512 sha256 sha1 md5
            camellia des aes ima hpc vis3 fmaf asi_blk_init vis2
    ```

```
            vis popc v8plus div32 mul32
```

L'utilisation de la valeur `generic` peut entraîner une détérioration des performances du domaine invité par rapport à l'utilisation de la valeur `native`. Cette détérioration éventuelle des performances est due au fait que le domaine invité utilise uniquement les fonctionnalités CPU disponibles sur tous les types de CPU, et non les fonctions matérielles natives d'une CPU particulière. En évitant d'utiliser ces fonctions, la valeur `generic` vous offre la possibilité de migrer le domaine entre des systèmes dont les CPU prennent en charge des fonctions différentes.

Pour la migration d'un domaine entre des systèmes SPARC T4 minimum, vous pouvez définir `cpu-arch=migration-class1` pour améliorer les performances du domaine invité. Les performances sont améliorées par l'utilisation de la valeur `generic` et la valeur `native` fournit encore une performance maximale pour le domaine invité.

Utilisez la commande `psrinfo -pv` lorsque la propriété `cpu-arch` est définie sur `native` pour déterminer le type de processeur, comme suit :

```
# psrinfo -pv
The physical processor has 2 virtual processors (0 1)
  SPARC-T5 (chipid 0, clock 3600 MHz)
```

Notez que lorsque la propriété `cpu-arch` est définie sur une valeur autre que `native`, la sortie `psrinfo -pv` n'affiche pas le type de plate-forme. Au lieu de cela, la commande montre que le module de CPU `sun4v-cpu` est chargé.

```
# psrinfo -pv
The physical processor has 2 cores and 13 virtual processors (0-12)
  The core has 8 virtual processors (0-7)
  The core has 5 virtual processors (8-12)
    sun4v-cpu (chipid 0, clock 3600 MHz)
```

Configuration requise pour la mémoire

Les exigences de mémoire pour la machine cible sont les suivantes :

- La machine cible doit avoir suffisamment de mémoire libre pour permettre la migration d'un domaine.
- La mémoire libre doit être disponible dans une disposition compatible.

Les exigences de compatibilité diffèrent selon la plate-forme SPARC. Toutefois, au minimum l'alignement de l'adresse réel et de l'adresse physique relatif à la taille de page maximale prise en charge doit être préservé pour chaque bloc mémoire dans le domaine migré.

Utilisez la commande `pagesize` pour déterminer la taille de page maximale prise en charge sur la machine cible.

Pour un domaine invité qui exécute au moins le SE Oracle Solaris 11.3, les blocs mémoire du domaine migré doivent être automatiquement fractionnés lors de la migration de sorte que le domaine migré peut s'adapter aux plus petits blocs de mémoire disponibles. Les blocs mémoire ne peuvent être fractionnés qu'en fonction des limites alignées sur la taille de page maximale.

D'autres exigences de disposition de mémoire pour les systèmes d'exploitation, microprogrammes ou plates-formes peuvent empêcher le fractionnement des blocs mémoire lors d'une migration donnée. Cette situation pourrait provoquer l'échec de la migration même si l'espace total de mémoire disponible est suffisant pour le domaine.

Configuration requise des périphériques d'E/S physiques pour la migration

Les domaines disposant d'un accès direct aux périphériques physiques ne peuvent pas être migrés. Par exemple, vous ne pouvez pas migrer des domaines d'E/S. Cependant, les périphériques virtuels associés à des périphériques physiques peuvent être migrés.

Pour plus d'informations, reportez-vous aux sections "Configuration requise des périphériques d'extrémité PCIe pour la migration" à la page 337 et "Configuration requise pour la migration des fonctions virtuelles SR-IOV PCIe" à la page 337.

Configuration requise des périphériques d'E/S virtuels physiques pour la migration

Tous les services d'E/S virtuels utilisés par le domaine à migrer doivent être disponibles sur la machine cible. En d'autres termes, les conditions suivantes doivent être remplies :

- Tous les disques backend virtuels utilisés dans le domaine à migrer doivent être définis sur la machine cible. Ce stockage partagé peut être un disque SAN ou un espace de stockage disponible via les protocoles NFS ou iSCSI. Le disque virtuel backend défini doit avoir les mêmes noms de service et de volume que sur la machine source. Les chemins d'accès peuvent être différents sur les machines source et cible, mais ils *doivent* pointer vers le même disque backend.

Attention - Une migration réussira même si les chemins d'accès à un disque virtuel backend sur les machines source et cible ne se réfèrent pas au même stockage. Cependant, le comportement du domaine sur la machine cible sera imprévisible, et il est possible que le domaine soit inutilisable. Pour corriger cette situation, arrêtez le domaine, corrigez le problème de configuration et redémarrez le domaine. Si vous n'exécutez pas ces étapes, le domaine risque de rester dans un état incohérent.

- Chaque périphérique réseau virtuel dans le domaine à migrer doit posséder un commutateur de réseau virtuel correspondant sur la machine cible. Les commutateurs de réseau virtuel doivent tous posséder le même nom que le commutateur de réseau virtuel auquel le périphérique est connecté sur la machine source.

  Par exemple, si `vnet0` dans le domaine à migrer est connecté à un service de commutateur virtuel appelé `switch-y`, un domaine sur la machine cible doit fournir un service de commutateur virtuel appelé `switch-y`.

  Remarque - Le réseau physique sur la machine cible doit être correctement configuré, de sorte que le domaine migré puisse accéder aux ressources réseau dont il a besoin. Sinon, certains services risquent d'être indisponibles sur le domaine une fois la migration terminée.

  Par exemple, vous pouvez souhaiter vous assurer que le domaine peut accéder au bon sous-réseau sur le réseau. Vous pouvez également vouloir vérifier que les passerelles, routeurs ou pare-feu sont correctement configurés, de sorte que le domaine puisse atteindre les systèmes distants à partir de la machine cible.

  Les adresses MAC utilisées par le domaine à migrer qui se trouvent dans la plage allouée automatiquement doivent être disponibles sur la machine cible.
- Un service de concentrateur de console virtuelle (`vcc`) doit exister sur la machine cible et disposer d'au moins un port libre. Les contraintes explicites de la console sont ignorées au cours de la migration. La console du domaine migré est créée en utilisant le nom du domaine migré comme groupe de consoles et n'importe quel port disponible sur n'importe quel périphérique `vcc` disponible du domaine de contrôle. Si aucun des ports disponibles n'est libre dans le domaine de contrôle, la console est créée à l'aide d'un port disponible sur un périphérique `vcc` disponible dans un domaine de service. La migration échoue s'il y a un conflit avec le nom de groupe par défaut.
- Chaque SAN virtuel utilisé par le domaine à migrer doit être défini sur la machine cible.

Configuration requise des périphériques d'extrémité PCIe pour la migration

Vous ne pouvez pas effectuer une migration de domaine sur un domaine d'E/S configuré avec des périphériques d'extrémité PCIe.

Pour plus d'informations sur la fonction d'E/S directes (DIO), reportez-vous à la section "Création d'un domaine d'E/S en assignant les périphériques d'extrémité PCIe" à la page 155.

Configuration requise pour la migration des fonctions virtuelles SR-IOV PCIe

Vous ne pouvez pas effectuer une migration de domaine sur un domaine d'E/S configuré avec des fonctions virtuelles SR-IOV PCIe.

Pour plus d'informations sur la fonction SR-IOV, reportez-vous à la section Chapitre 8, Création d'un domaine d'E/S à l'aide des fonctions virtuelles SR-IOV PCIe.

Configuration requise des unités cryptographiques pour la migration

Sur les plates-formes comportant des unités cryptographiques, vous pouvez migrer un domaine invité associé à des unités cryptographiques s'il exécute un système d'exploitation prenant en charge la reconfiguration dynamique (DR) des unités cryptographiques.

Au début de la migration, Logical Domains Manager détermine si le domaine à migrer prend en charge la reconfiguration dynamique des unités cryptographiques. Si tel est le cas, Logical Domains Manager tente de supprimer les unités cryptographiques du domaine. A la fin de la migration, les unités cryptographiques sont de nouveau ajoutées au domaine migré.

Remarque - Si les contraintes des unités cryptographiques ne peuvent pas être respectées sur la machine cible, l'opération de migration ne sera toutefois pas bloquée. Dans ce cas, le domaine migré peut présenter moins d'unités cryptographiques qu'avant l'opération de migration.

Reconfiguration retardée dans un domaine actif

Les opérations de reconfiguration retardée actives sur les hôtes source et cible empêchent le début de la migration. Vous n'êtes pas autorisé à démarrer une opération de reconfiguration retardée pendant qu'une migration est en cours.

Migration alors que la stratégie de gestion de l'alimentation élastique est en cours d'application sur un domaine actif

Vous pouvez effectuer une migration en direct lorsque la stratégie de gestion de d'alimentation (PM) élastique est en cours d'application sur la machine source ou sur la machine cible.

Opérations sur d'autres domaines

Pendant qu'une migration est en cours sur une machine, toute opération pouvant provoquer la modification de l'état ou la configuration du domaine en cours de migration est bloquée. Toutes les opérations sur le domaine lui-même, ainsi que les opérations de liaison et d'arrêt sur les autres domaines de la machine sont bloquées.

Migration d'un domaine à partir de la PROM OpenBoot ou un domaine en cours d'exécution dans le débogueur de noyau

L'exécution de la migration d'un domaine requiert une coordination entre Logical Domains Manager et le SE SE Oracle Solaris en cours d'exécution dans le domaine à migrer. Lorsqu'un domaine à migrer est exécuté dans OpenBoot ou dans le débogueur de noyau (`kmdb`), cette coordination est impossible. Par conséquent, la tentative de migration échoue.

Lorsqu'un domaine à migrer s'exécute en OpenBoot, le message suivant apparaît :

```
primary# ldm migrate ldg1 system2
```

```
Migration is not supported while the domain ldg1 is in the 'OpenBoot Running' state
Domain Migration of LDom ldg1 failed
```

Lorsqu'un domaine à migrer s'exécute dans le débogueur de noyau, `kmdb`, le message suivant apparaît :

```
primary# ldm migrate ldg1 system2
Migration is not supported while the domain ldg1 is in the 'Solaris debugging' state
Domain Migration of LDom ldg1 failed
```

Migration de domaines liés ou inactifs

Seules quelques limitations s'appliquent à un domaine lié ou inactif, car de tels domaines ne sont pas exécutés au moment de la migration. Par conséquent, vous pouvez effectuer la migration entre différents types de plates-formes, par exemple SPARC T3 vers SPARC T5 ou les Plates-formes Fujitsu M10, car aucun état d'exécution n'est copié d'une plate-forme à l'autre.

La migration d'un domaine lié nécessite que la machine cible soit capable de respecter les contraintes en termes de CPU, mémoire et E/S du domaine à migrer. Si elles ne sont pas satisfaites, la migration échoue.

Attention - Lors de la migration de domaines liés, les valeurs `options` et `mpgroup` du backend du disque virtuel ne sont pas cochées car aucune information d'état n'est échangée avec l'ordinateur cible. Cette vérification *est* effectuée lors de la migration d'un domaine actif.

La migration d'un domaine inactif ne présente pas de telles contraintes. Cependant, la machine cible doit satisfaire les contraintes du domaine migré lorsqu'une liaison est tentée ultérieurement, ou la liaison du domaine est vouée à l'échec.

Remarque - Lorsqu'une migration de domaine se termine, enregistrez une nouvelle configuration sur le SP des systèmes source et cible. En conséquence, l'état du domaine migré est correct si le système source ou le système cible est arrêté et redémarré.

Configuration requise des périphériques d'E/S virtuels physiques pour la migration

Pour un domaine inactif, aucun contrôle des contraintes d'ES virtuelles (VIO) n'est effectué. Par conséquent, les serveurs VIO n'ont pas besoin d'exister pour que la migration aboutisse. Tout

comme avec un domaine inactif, les serveurs VIO doivent exister et être disponibles au moment de la liaison du domaine.

Configuration requise des périphériques d'extrémité PCIe pour la migration

Vous ne pouvez pas effectuer une migration de domaine sur un domaine d'E/S configuré avec des périphériques d'extrémité PCIe. Cette condition s'applique aux domaines liés mais pas aux domaines inactifs.

Pour plus d'informations sur la fonction d'E/S directes (DIO), reportez-vous à la section "Création d'un domaine d'E/S en assignant les périphériques d'extrémité PCIe" à la page 155.

Configuration requise pour la migration des fonctions virtuelles SR-IOV PCIe

Vous ne pouvez pas effectuer une migration de domaine sur un domaine d'E/S configuré avec des fonctions virtuelles SR-IOV PCIe. Cette condition s'applique aux domaines liés mais pas aux domaines inactifs.

Pour plus d'informations sur la fonction SR-IOV, reportez-vous à la section Chapitre 8, Création d'un domaine d'E/S à l'aide des fonctions virtuelles SR-IOV PCIe.

Surveillance d'une migration en cours

Pendant la migration, le domaine en cours de migration et le domaine migré sont affichés différemment dans la sortie d'état. La sortie de la commande `ldm list` indique l'état du domaine en migration.

La sixième colonne de la zone FLAGS présente l'une des valeurs suivantes :

- `s` – Le domaine qui est la source de la migration.
- `t` – Le domaine migré qui est la cible de la migration.

- `e` – Une erreur s'est produite et nécessite une intervention de l'utilisateur.

La commande suivante indique que le domaine `ldg-src` est la source de la migration :

```
# ldm list ldg-src
NAME       STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
ldg-src    suspended  -n---s          1     1G       0.0%  2h 7m
```

La commande suivante indique que le domaine `ldg-tgt` est la cible de la migration :

```
# ldm list ldg-tgt
NAME       STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
ldg-tgt    bound      -----t  5000    1     1G
```

La forme longue de la sortie d'état affiche des informations supplémentaires relatives à la migration. Sur la machine source, la sortie d'état affiche le pourcentage de progression de l'opération, ainsi que les noms de la machine cible et du domaine migré. De même, sur la machine cible, la sortie d'état affiche le pourcentage de progression de l'opération, ainsi que les noms de la machine source et du domaine en cours de migration.

La commande suivante affiche la progression d'une opération de migration pour le domaine `ldg-src` :

```
# ldm list -o status ldg-src
NAME
ldg-src

STATUS
    OPERATION    PROGRESS    TARGET
    migration    17%         system2
```

Annulation d'une migration en cours

Une fois la migration démarrée, l'opération de migration se termine si la commande `ldm` est interrompue par un signal KILL. Lorsque l'opération de migration est interrompue, le domaine migré est détruit et le domaine à migrer est repris, s'il était actif. Si le shell de contrôle de la commande `ldm` est perdu, la migration continue à l'arrière-plan.

Si vous annulez une migration en direct, le contenu de la mémoire de l'instance de domaine créé sur la machine cible doit être "nettoyé" par l'hyperviseur. Ce processus de nettoyage est effectué pour des raisons de sécurité et doit être terminé afin que la mémoire puisse être renvoyée vers le pool de mémoire libre. Durant la progression du nettoyage, les commandes `ldm` deviennent non réactives. Par conséquent, Logical Domains Manager semble suspendu. Attendez la fin de la demande de nettoyage avant de tenter d'exécuter les autres commandes `ldm`. Ce processus peut être long. Par exemple, un domaine invité comptant 500 Go de mémoire peut mettre jusqu'à 7 minutes pour terminer le processus sur un serveur SPARC T4.

Récupération sur un échec de migration

L'opération de migration prend fin si la connexion réseau est perdue une fois que le domaine en cours de migration a fini d'envoyer toutes les informations d'état d'exécution au domaine migré, mais avant que le domaine migré ne puisse reconnaître que le domaine a été relancé.

Vous devez déterminer si la migration s'est bien terminée en effectuant les étapes suivantes :

1. Déterminez si le domaine migré a bien repris ses opérations. Le domaine migré sera dans l'un des deux états suivants :
   - Si la migration a abouti, le domaine migré est en état normal.
   - Si la migration a échoué, la machine cible nettoie et détruit le domaine migré.
2. Si le domaine migré reprend correctement ses opérations, vous pouvez en toute sécurité détruire le domaine en état d'erreur sur la machine source. Cependant, si le domaine migré n'est pas présent, le domaine sur la machine source est toujours la version maître du domaine et doit être récupérée. Pour récupérer ce domaine, exécutez la commande `ldm cancel-operation` sur la machine source. Cette commande efface l'état d'erreur et restaure le domaine à son état d'origine.

Exemples de migrations

EXEMPLE 54 Migration d'un domaine invité à l'aide des certificats SSL

Cet exemple décrit la migration du domaine `ldg1` vers une machine appelée `system2`. Avant de démarrer l'opération de migration, vous devez configurer les certificats SSL sur les machines source et cible. Voir "Procédure de configuration des certificats SSL pour la migration" à la page 324.

```
# ldm migrate-domain -c ldg1 system2
```

EXEMPLE 55 Migration d'un domaine invité

Cet exemple décrit la migration du domaine `ldg1` vers une machine appelée `system2`.

```
# ldm migrate-domain ldg1 system2
Target Password:
```

Pour effectuer cette migration sans devoir entrer de mot de passe pour la machine cible, utilisez la commande suivante :

```
# ldm migrate-domain -p pfile ldg1 system2
```

L'option -p prend un nom de fichier comme argument. Le fichier spécifié contient le mot de passe de superutilisateur pour la machine cible. Dans cet exemple, `pfile` contient le mot de passe de la machine cible, `system2`.

EXEMPLE 56 Migration et renommage d'un domaine invité

Cet exemple indique comment renommer un domaine dans le cadre de l'opération de migration. Le domaine `ldg-src` sur la machine source est renommé `ldg-tgt` sur la machine cible (`system2`) dans le cadre de la migration. En outre, l'utilisateur `ldm-admin` est utilisé pour l'authentification sur la machine cible.

```
# ldm migrate ldg-src ldm-admin@system2:ldg-tgt
Target Password:
```

EXEMPLE 57 Message d'erreur de migration

Cet exemple montre le message d'erreur qui peut apparaître si la machine cible ne prend pas en charge les dernières fonctionnalités de migration.

```
# ldm migrate ldg1 dt212-346
Target Password:
The target machine is running an older version of the domain
manager that does not support the latest migration functionality.

Upgrading to the latest software will remove restrictions on
a migrated domain that are in effect until it is rebooted.
Consult the product documentation for a full description of
these restrictions.

The target machine is running an older version of the domain manager
that is not compatible with the version running on the source machine.

Domain Migration of LDom ldg1 failed
```

EXEMPLE 58 Obtention de l'état de migration pour le domaine sur la machine cible

Cet exemple présente la procédure d'obtention de l'état sur un domaine migré lorsqu'une migration est en cours. Dans cet exemple, la machine source est `t5-sys-1`.

```
# ldm list -o status ldg-tgt
NAME
ldg-tgt

STATUS
    OPERATION    PROGRESS    SOURCE
    migration    55%         t5-sys-1
```

EXEMPLE 59 Obtention de l'état de migration analysable pour le domaine sur la machine cible

Cet exemple présente la procédure d'obtention de l'état analysable sur un domaine migré lorsqu'une migration est en cours. Dans cet exemple, la machine cible est `system2`.

```
# ldm list -o status -p ldg-src
VERSION 1.6
DOMAIN|name=ldg-src|
STATUS
|op=migration|progress=42|error=no|target=system2
```

♦ ♦ ♦ CHAPITRE 15

Gestion des ressources

Ce chapitre contient des informations sur la gestion des ressources des systèmes Oracle VM Server for SPARC.

Ce chapitre aborde les sujets suivants :

Reconfiguration des ressources

Un système exécutant le logiciel Oracle VM Server for SPARC peut configurer des ressources, notamment des CPU virtuelles, des périphériques d'E/S virtuels, des unités cryptographiques et de la mémoire. Certaines ressources peuvent être configurées de manière dynamique sur un domaine en cours d'exécution, tandis que d'autres doivent être configurées sur un domaine arrêté. Si une ressource ne peut pas être configurée de manière dynamique sur le domaine de contrôle, vous devez d'abord lancer une reconfiguration retardée. La reconfiguration retardée reporte les activités de configuration à après la réinitialisation du domaine de contrôle.

Reconfiguration dynamique

La reconfiguration dynamique (DR) permet d'ajouter ou de supprimer des ressources lorsque le système d'exploitation (SE) est en cours d'exécution. La capacité à réaliser une reconfiguration dynamique d'un type de ressource particulier est dépendante du fait que le SE s'exécute sur le domaine logique.

La reconfiguration dynamique est prise en charge pour les ressources suivantes :

- **CPU virtuelles –** Prises en charge dans toutes les versions du SE Oracle Solaris 10 et Oracle Solaris 11
- **Coeurs complets de CPU –** Voir la section "Versions du Système d'exploitation Oracle Solaris complètes" du manuel *Guide d'installation d'Oracle VM Server for SPARC 3.4*
- **Périphériques d'E/S virtuels –** Pris en charge au moins dans le SE Oracle Solaris 10 10/08 et dans le SE Oracle Solaris 11
- **Unités cryptographiques –** Prises en charge au moins dans le SE Oracle Solaris 10 1/13 et dans le SE Oracle Solaris 11
- **Mémoire –** Reportez-vous à la section "Utilisation de la reconfiguration dynamique de la mémoire" à la page 364
- **Périphériques d'E/S physiques –** Non pris en charge

Pour utiliser la fonction de DR, le démon de DR de Logical Domains `drd` doit être en cours d'exécution dans le domaine que vous souhaitez modifier. Reportez-vous à la page de manuel `drd`(1M).

Reconfiguration retardée

Au contraire des opérations de DR ayant lieu immédiatement, les opérations de reconfiguration retardée ont lieu dans les circonstances suivantes :

- Après la réinitialisation suivante du SE
- Après un arrêt ou un démarrage d'un domaine logique si aucun SE n'est en cours d'exécution

Les opérations de reconfiguration retardée sont généralement limitées au domaine de contrôle. Pour tous les autres domaines, vous devez arrêter le domaine pour modifier la configuration, à moins que les ressources puissent être reconfigurées de manière dynamique.

Les opérations de reconfiguration retardées sont limitées au domaine de contrôle. Vous pouvez exécuter un nombre limité de commandes lorsqu'une reconfiguration retardée est démarrée sur le domaine root, ce afin d'effectuer des opérations qui ne peuvent pas être effectuées de manière dynamique. Ces sous-commandes sont `add-io`, `set-io`, `remove-io`, `create-vf` et `destroy-vf`. Vous

pouvez également exécuter la commande `ldm start-reconf` sur le domaine root. Pour tous les autres domaines, vous devez arrêter le domaine pour modifier la configuration, à moins que les ressources puissent être reconfigurées de manière dynamique.

Lorsqu'une reconfiguration retardée est en cours, les autres demandes de reconfiguration du domaine sont retardées jusqu'à la réinitialisation de celui-ci, ou jusqu'à son arrêt et son redémarrage.

La sous-commande `ldm cancel-reconf` annule les opérations de reconfiguration retardée sur le domaine. Pour plus d'informations sur l'utilisation de la fonction de reconfiguration retardée, reportez-vous à la page de manuel `ldm`(1M).

Remarque - Vous ne pouvez pas utiliser la commande `ldm cancel-reconf` si d'autres commandes `ldm remove-`* ont déjà effectué une opération de reconfiguration retardée sur des périphériques d'E/S virtuels. Dans ce cas de figure, la commande `ldm cancel-reconf` échoue.

Vous pouvez utiliser la reconfiguration retardée pour diminuer les ressources sur le domaine de contrôle. Pour supprimer un grand nombre de CPU du domaine de contrôle, reportez-vous à la section "La suppression d'un grand nombre de CPU d'un domaine invité peut échouer" à la page 386. Pour supprimer de grandes quantités de mémoire du domaine de contrôle, reportez-vous à la section "Réduction de la mémoire du domaine de contrôle" à la page 366.

Remarque - Lorsqu'un domaine `primary` est en état de reconfiguration retardée, l'alimentation des ressources gérées par Oracle VM Server for SPARC est *uniquement* gérée après la réinitialisation du domaine `primary`. Les ressources gérées directement par le SE, telles que les CPU gérés par le Power Aware Dispatcher Solaris ne sont pas affectées par cet état.

Une seule opération de configuration de CPU peut être exécutée durant une reconfiguration retardée

N'essayez pas d'exécuter plusieurs opérations de configuration de CPU sur le domaine `primary` alors que sa reconfiguration est retardée. Si vous tentez d'effectuer plusieurs demandes de configuration de CPU, celles-ci seront rejetées.

Solution de contournement : effectuez l'une des opérations suivantes

- Annulez la reconfiguration retardée, lancez-en une autre, puis demandez les modifications de configuration perdues de la dernière reconfiguration retardée.
- Réinitialisez le domaine de contrôle à l'aide du nombre erroné de CPU et effectuez les corrections d'allocation nécessaires après la réinitialisation du domaine.

Allocation des ressources

Le mécanisme d'allocation des ressources utilise des contraintes d'allocation des ressources pour assigner des ressources à un domaine au moment de l'association.

Une *contrainte d'allocation de ressource* est une contrainte stricte que le système doit respecter lorsque vous assignez une ressource à un domaine. Si la contrainte ne peut pas être respectée, l'allocation de ressource et l'association du domaine échouent.

Attention - Ne créez pas de dépendance circulaire entre deux domaines où chaque domaine fournit des services à l'autre. Une telle configuration crée une condition de point de panne unique où une interruption de service dans un domaine entraîne l'indisponibilité de l'autre domaine. Les configurations de dépendance circulaire vous empêchent également de dissocier les domaines suite à leur association initiale.

Le Logical Domains Manager n'empêche pas la création de dépendances de domaines circulaires.

S'il est impossible de dissocier les domaines en raison d'une dépendance circulaire, retirez les périphériques causant la dépendance puis tentez de dissocier les domaines.

Allocation de CPU

Lorsque vous exécutez des threads à partir d'un seul coeur sur des domaines séparés, les performances peuvent être ralenties ou imprévisibles. Le logiciel Oracle VM Server for SPARC utilise la fonction d'affinité CPU pour optimiser l'allocation de CPU lors du processus de liaison du domaine logique qui doit se produire avant que vous puissiez démarrez le domaine. Cette fonction tente de conserver des threads à partir du même coeur affecté au même domaine logique : ce type d'allocation améliore le partage de cache entre les threads dans ce coeur.

L'affinité CPU tente d'éviter le partage des coeurs entre les domaines sauf s'il n'existe pas d'autres moyens. Lorsqu'un coeur partiel a été assigné à un domaine et qu'il nécessite des strands supplémentaires, les strands du coeur partiel sont d'abord associés puis, si nécessaire, un autre coeur libre est assigné au domaine pour satisfaire la demande.

Le mécanisme d'allocation de CPU utilise les contraintes suivantes pour les ressources de CPU :

- **Contrainte whole-core.** Cette contrainte spécifie que les coeurs de CPU sont assignés à un domaine plutôt qu'à des CPU virtuelles. Tant que la contrainte max-cores n'est pas activée

sur le domaine, la contrainte whole-core peut être ajoutée ou supprimée, respectivement à l'aide des commandes `ldm set-core` et `ldm set-vcpu`. Le domaine peut être inactif, associé ou actif. Toutefois, le nombre de coeurs disponibles doit être suffisant pour satisfaire la demande d'application de la contrainte. Par exemple, au pire, si un domaine partageant des coeurs avec un autre domaine demande la contrainte whole-core, il faut que des coeurs de la liste libre soient disponibles pour satisfaire la demande. Au mieux, toutes les CPU virtuelles du coeur sont déjà placées sur des frontières de coeur, si bien que la contrainte est appliquée sans modification des ressources CPU.

- **Contrainte de nombre maximal de coeurs (max-cores).** Cette contrainte définit le nombre maximal de coeurs pouvant être assignés à un domaine associé ou actif.

▼ Procédure d'application de la contrainte whole-core

Assurez-vous que la contrainte whole-core est activée sur le domaine avant de définir la contrainte max-cores.

1. **Appliquez la contrainte whole-core sur le domaine.**

   ```
   primary# ldm set-core 1 domain-name
   ```

2. **Assurez-vous que la contrainte whole-core est activée sur le domaine.**

   ```
   primary# ldm ls -o resmgmt domain-name
   ```

   Notez que `max-cores` est défini sur `unlimited`. Le domaine ne peut pas être utilisé avec le partitionnement forcé tant que la contrainte max-cores n'est pas activée.

Exemple 60 Application de la contrainte whole-core

L'exemple suivant illustre l'application de la contrainte whole-core au domaine `ldg1`. La première commande applique la contrainte, tandis que la deuxième commande vérifie qu'elle est activée.

```
primary# ldm set-core 1 ldg1
primary# ldm ls -o resmgmt ldg1
NAME
ldg1

CONSTRAINT
    cpu=whole-core
    max-cores=unlimited
```

▼ Procédure d'application de la contrainte max-cores

Assurez-vous que la contrainte whole-core est activée sur le domaine avant de définir la contrainte max-cores.

Vous pouvez uniquement activer, modifier ou désactiver la contrainte max-cores sur un domaine inactif et non pas sur un domaine associé ou actif. Lorsque vous mettez à jour la contrainte max-cores sur le domaine de contrôle, la commande `ldm set-domain` déclenche automatiquement une reconfiguration retardée.

1. Activez la contrainte max-cores sur le domaine.

```
primary# ldm set-domain max-cores=max-number-of-CPU-cores  domain-name
```

Remarque - Les unités cryptographiques associées à ces coeurs ne sont pas affectées par les ajouts de coeurs. Par conséquent, le système n'ajoute pas automatiquement les unités cryptographiques associées à ce domaine. Cependant, une unité cryptographique est automatiquement supprimée *uniquement* lorsque la dernière CPU virtuelle du coeur est supprimée. Cette action empêche qu'une unité cryptographique ne devienne “orpheline”.

2. Assurez-vous que la contrainte whole-core est activée.

```
primary# ldm ls -o resmgmt domain-name
```

3. Associez et redémarrez le domaine.

```
primary# ldm bind domain-name
primary# ldm start domain-name
```

A présent, vous pouvez utiliser le domaine avec partitionnement forcé.

Exemple 61 Application de la contrainte max-cores

Cet exemple montre comment limiter la contrainte max-cores à trois coeurs en définissant la propriété `max-cores` et vérifier que la contrainte est activée :

```
primary# ldm set-domain max-cores=3 ldg1
primary# ldm ls -o resmgmt ldg1
NAME
ldg1

CONSTRAINT
    cpu=whole-core
    max-cores=3
```

A présent, vous pouvez utiliser le domaine avec partitionnement forcé.

L'exemple suivant supprime la contrainte max-cores du domaine `ldg1` non associé et inactif, mais conserve la contrainte whole-core en l'état.

```
primary# ldm stop ldg1
primary# ldm unbind ldg1
primary# ldm set-domain max-cores=unlimited ldg1
```

Pour supprimer à la fois la contrainte max-cores et la contrainte whole-core du domaine `ldg1`, vous pouvez également affecter des CPU virtuelles à la place des coeurs de la manière suivante :

```
primary# ldm set-vcpu 8 ldg1
```

Dans les deux cas, vous devez associer et redémarrer le domaine.

```
primary# ldm bind ldg1
primary# ldm start ldg1
```

Interactions entre la contrainte whole-core et les autres fonctions des domaines

Cette section décrit les interactions entre la contrainte whole-core et les fonctions suivantes :

- "Reconfiguration dynamique de la CPU" à la page 351
- "Gestion dynamique des ressources" à la page 351

Reconfiguration dynamique de la CPU

La contrainte whole-core est totalement compatible avec la reconfiguration dynamique (DR) de la CPU. Lorsqu'un domaine est défini avec la contrainte whole-core, vous pouvez utiliser la commande `ldm add-core`, `ldm set-core` ou `ldm remove-core` pour modifier le nombre de coeurs sur un domaine actif.

Cependant, si un domaine lié ou actif n'est pas en mode de reconfiguration retardée, son nombre de coeurs ne peut pas dépasser le nombre maximal de coeurs. Ce maximum est défini avec la contrainte de coeur maximum, qui est automatiquement activée lorsque la contrainte whole-core est activée. Une opération de reconfiguration dynamique de la CPU ne respectant pas la contrainte de coeur maximum échoue.

Gestion dynamique des ressources

La contrainte whole-core est totalement compatible avec la gestion dynamique des ressources (DRM).

Les interactions attendues entre la contrainte whole-core et la DRM sont les suivantes :

- S'il existe une stratégie DRM pour un domaine, vous ne pouvez pas faire basculer le domaine d'une utilisation de coeurs limités (contrainte whole-core) à une utilisation de coeurs non limités et inversement. Par exemple :
  - Si un domaine a une utilisation de coeurs limités, vous ne pouvez pas utiliser la commande `ldm set-vcpu` pour spécifier un nombre de CPU virtuelles et supprimer la contrainte whole-core.
  - Si un domaine a une utilisation de coeurs non limités, vous ne pouvez pas utiliser la commande `ldm set-core` pour spécifier un nombre de coeurs entiers et ajouter la contrainte whole-core.
- Si un domaine a une utilisation de coeurs limités et que vous spécifiez la valeur `attack`, `decay`, `vcpu-min` ou `vcpu-max`, cette valeur doit être un multiple de coeur complet (whole-core).

Configuration du système avec des partitions forcées

Cette section décrit le partitionnement forcé avec le logiciel Oracle VM Server for SPARC et l'utilisation du partitionnement forcé pour assurer sa conformité aux conditions d'octroi de licences CPU d'Oracle.

Pour plus d'informations sur les conditions requises pour le partitionnement forcé des licences Oracle, reportez-vous à la section Partitionnement : partitionnement du serveur/matériel (`http://www.oracle.com/us/corporate/pricing/partitioning-070609.pdf`).

- **Coeurs et threads de CPU.** Les processeurs utilisés dans ces systèmes sont équipés de plusieurs coeurs de CPU dont chacun contient plusieurs threads de CPU.
- **Partitionnement forcé et coeurs complets de CPU.** Le partitionnement forcé est appliqué à l'aide de configurations whole-core de CPU. Une configuration whole-core de CPU possède des domaines auxquels des coeurs complets de CPU sont assignés au lieu de threads de CPU individuels. Par défaut, un domaine est configuré pour utiliser des threads de CPU.

  Lors de l'inclusion d'un domaine dans une configuration whole-core, le système crée et configure le nombre de coeurs de CPU spécifié et tous ses threads de CPU dans le domaine. L'utilisation d'une configuration whole-core de CPU restreint le nombre de coeurs de CPU pouvant être automatiquement assignés à un domaine lié ou actif.
- **Octroi de licence de partitionnement forcé Oracle.** Pour être conforme à l'exigence d'octroi de licence de partitionnement forcé d'Oracle, voir Hard Partitioning With Oracle VM Server for SPARC (`http://www.oracle.com/technetwork/server-storage/vm/ovm-sparc-hard-partitioning-1403135.pdf`).

Vous devez également utiliser des coeurs complets de CPU de la manière suivante :

- Un domaine qui exécute des applications utilisant l'octroi de licence de partitionnement forcé d'Oracle doit être configuré avec des coeurs complets de CPU et un nombre maximal de coeurs.
- Il n'est pas nécessaire de configurer un domaine qui n'exécute aucune application utilisant l'octroi de licence de partitionnement forcé d'Oracle avec des coeurs complets de CPU. Par exemple, si vous n'exécutez aucune application Oracle dans le domaine de contrôle, ce domaine n'a pas besoin d'être configuré avec des coeurs complets de CPU.

Vérification de la configuration d'un domaine

La commande `ldm list-o` permet de déterminer si un domaine est configuré avec des coeurs complets de CPU et décrit comment dresser la liste des coeurs de CPU affectés à un domaine.

- Pour déterminer si le domaine est configuré avec des coeurs complets de CPU :

  ```
  primary# ldm list -o resmgmt domain-name
  ```

  Assurez-vous que la contrainte whole-core apparaît dans la sortie et que la propriété `max-cores` spécifie le nombre maximal de coeurs de CPU configurés pour le domaine. Reportez-vous à la page de manuel `ldm`(1M).

  La commande suivante indique que le domaine `ldg1` est configuré avec des coeurs de processeur complets et avec un maximum de cinq coeurs :

  ```
  primary# ldm list -o resmgmt ldg1
  NAME
  ldg1

  CONSTRAINT
     whole-core
     max-cores=5
  ```

- Lorsqu'un domaine est associé, des coeurs de CPU sont affectés au domaine. Pour dresser la liste des coeurs de CPU affectés à un domaine :

  ```
  primary# ldm list -o core domain-name
  ```

  La commande suivante dresse la liste des coeurs affectés au domaine `ldg1` :

  ```
  primary# ldm list -o core ldg1
  NAME
  ldg1

  CORE
  ```

```
CID PCPUSET
1 (8, 9, 10, 11, 12, 13, 14, 15)
2 (16, 17, 18, 19, 20, 21, 22, 23)
```

Configuration d'un domaine avec des coeurs complets de CPU

Les tâches décrites dans cette section expliquent comment créer un nouveau domaine avec des coeurs complets de CPU, comment configurer un domaine existant avec des coeurs complets de CPU et comment configurer un domaine `primary` avec des coeurs complets de CPU.

Utilisez la commande suivante pour configurer un domaine de manière à ce qu'il utilise des coeurs complets de CPU :

`ldm set-core` *number-of-CPU-cores domain*

Cette commande spécifie également le nombre maximal de coeurs de CPU du domaine. Reportez-vous à la page de manuel `ldm`(1M).

Le nombre maximal de coeurs et l'allocation de coeurs de CPU sont gérés par des commandes distinctes. Ces commandes vous permettent d'allouer des coeurs de CPU, de définir un plafond ou d'effectuer ces deux opérations de façon totalement indépendante. L'unité d'allocation peut être définie sur coeurs même si aucun nombre maximal de coeurs n'a été configuré. Toutefois, l'exécution du système dans ce mode n'est *pas* compatible avec la configuration du partitionnement forcé sur votre système Oracle VM Server for SPARC.

- Allouez le nombre spécifié de coeurs de CPU à un domaine à l'aide de la sous-commande `add-core`, `set-core` ou `rm-core`.
- Définissez le nombre maximal de coeurs à l'aide de la sous-commande `create-domain` ou de la sous-commande `set-domain` pour spécifier la valeur de la propriété `max-cores`.

  Vous devez définir le plafond si vous souhaitez configurer le partitionnement forcé sur votre système Oracle VM Server for SPARC.

▼ Procédure de création d'un nouveau domaine avec des coeurs complets de CPU

Remarque - Vous devez uniquement vous arrêter et dissocier le domaine si vous souhaitez définir la contrainte max-cores.

1. **Créez le domaine.**

```
primary# ldm create domain-name
```

2. **Définissez le nombre de coeurs complets de CPU pour le domaine.**

```
primary# ldm set-core number-of-CPU-cores domain
```

3. **Définissez la propriété `max-cores` pour le domaine (facultatif).**

```
primary# ldm set-domain max-cores=max-number-of-CPU-cores domain
```

4. **Configurez le domaine.**
Pendant la configuration, assurez-vous d'utiliser la commande `ldm add-core`, `ldm set-core` ou `ldm rm-core`.

5. **Associez et démarrez le domaine.**

```
primary# ldm bind domain-name
primary# ldm start domain-name
```

Exemple 62 Création d'un nouveau domaine avec deux coeurs complets de CPU

Dans cet exemple, un domaine `ldg1` comportant deux coeurs complets de CPU est créé. La première commande crée le domaine `ldg1`. La seconde commande configure le domaine `ldg1` avec deux coeurs complets de CPU.

A ce stade, vous pouvez configurer plus précisément le domaine, sous réserve des restrictions décrites à l'étape 3 de la section "Procédure de création d'un nouveau domaine avec des coeurs complets de CPU" à la page 354.

Les troisième et quatrième commandes indiquent comment associer et démarrer le domaine `ldg1`, suite à quoi vous pouvez utiliser le domaine `ldg1`.

```
primary# ldm create ldg1
primary# ldm set-core 2 ldg1
...
primary# ldm bind ldg1
primary# ldm start ldg1
```

▼ Procédure de configuration d'un domaine existant avec des coeurs complets de CPU

Si un domaine existe déjà et qu'il est configuré pour l'utilisation de threads de CPU, vous pouvez modifier sa configuration pour qu'il utilise des coeurs complets de CPU.

1. **Arrêtez et dissociez le domaine (facultatif).**
Cette étape n'est requise que si vous avez également défini la contrainte max-cores.

```
primary# ldm stop domain-name
```

```
primary# ldm unbind domain-name
```

2. **Définissez le nombre de coeurs complets de CPU pour le domaine.**

```
primary# ldm set-core number-of-CPU-cores domain
```

3. **Définissez la propriété `max-cores` pour le domaine (facultatif).**

```
primary# ldm set-domain max-cores=max-number-of-CPU-cores domain
```

4. **Associez à nouveau et redémarrez le domaine (facultatif).**

   Cette étape n'est requise que si vous avez également défini la contrainte max-cores.

```
primary# ldm bind domain-name
primary# ldm start domain-name
```

Exemple 63 Configuration d'un domaine existant avec quatre coeurs complets de CPU

Cet exemple met à jour la configuration d'un domaine existant `ldg1` en le configurant avec quatre coeurs de processeur complets.

```
primary# ldm set-core 4 ldg1
```

▼ Procédure de configuration du domaine primary avec des coeurs complets de CPU

Si le domaine `primary` est configuré pour utiliser des threads de CPU, vous pouvez modifier sa configuration pour utiliser des coeurs complets de CPU.

1. **Placez le domaine `primary` en mode de reconfiguration retardée (facultatif).**

   Vous ne devez démarrer une reconfiguration retardée que si vous souhaitez modifier la propriété `max-cores`.

```
primary# ldm start-reconf primary
```

2. **Définissez le nombre de coeurs complets de CPU pour le domaine `primary`.**

```
primary# ldm set-core number-of-CPU-cores primary
```

3. **Définissez la propriété `max-cores` pour le domaine `primary` (facultatif).**

```
primary# ldm set-domain max-cores=max-number-of-CPU-cores primary
```

4. **Réinitialisez le domaine `primary`.**

   Utilisez la procédure appropriée pour réinitialiser le domaine `primary` ; celle-ci dépend de la configuration système. Voir la section "Réinitialisation du domaine root avec des extrémités PCIe configurées" à la page 162.

Vous devez uniquement réinitialiser le domaine si vous souhaitez modifier la propriété `max-cores`.

Exemple 64 Configuration du domaine de contrôle avec deux coeurs complets de CPU

Cet exemple configure des coeurs complets de CPU sur le domaine `primary`. La première commande déclenche le mode de reconfiguration retardée sur le domaine `primary`. La seconde commande configure le domaine `primary` avec deux coeurs complets de CPU. La troisième commande définit la propriété `max-cores` sur `2` et la quatrième commande réinitialise le domaine `primary`.

```
primary# ldm start-reconf primary
primary# ldm set-core 2 primary
primary# ldm set-domain max-cores=2 primary
primary# shutdown -i 5
```

Les étapes facultatives 1 et 4 sont nécessaires uniquement si vous souhaitez modifier la propriété `max-cores`.

Interaction des systèmes de partitionnement forcé avec d'autres fonctions de Oracle VM Server for SPARC

Cette section décrit l'interaction des systèmes de partitionnement forcé avec d'autres fonctions Oracle VM Server for SPARC.

Reconfiguration dynamique de la CPU

Vous pouvez utiliser la reconfiguration dynamique de CPU avec des domaines configurés avec des coeurs complets de CPU. Toutefois, vous pouvez ajouter ou supprimer des coeurs de CPU entiers, pas des threads de CPU individuels. L'état de partitionnement forcé du système est entretenu par la fonction de reconfiguration dynamique de CPU. En outre, si des coeurs de CPU sont ajoutés à un domaine de manière dynamique, la valeur maximale est appliquée. Par conséquent, la commande de reconfiguration dynamique de CPU échouerait si elle tentait de dépasser le nombre maximal de CPU.

Remarque - La propriété `max-cores` ne peut pas être modifiée, sauf si le domaine est arrêté et dissocié. Par conséquent, pour augmenter le nombre maximal de coeurs spécifié lors de la définition de la contrainte whole-core, vous devez tout d'abord arrêter et dissocier le domaine.

Utilisez les commandes suivantes pour ajouter ou supprimer de façon dynamique des coeurs complets de CPU dans un domaine lié ou actif et pour définir de façon dynamique le nombre de coeurs complets de CPU pour un domaine lié ou actif :

`ldm add-core` *number-of-CPU-cores domain*

`ldm rm-core` *number-of-CPU-cores domain*

`ldm set-core` *number-of-CPU-cores domain*

Remarque - Si le domaine n'est pas actif, ces commandes ajustent également le nombre maximal de coeurs de CPU pour le domaine. Si le domaine est associé ou actif, ces commandes ne modifient pas le nombre maximal de coeurs de CPU pour le domaine.

EXEMPLE 65 Ajout dynamique de deux coeurs complets de CPU à un domaine

Cet exemple montre comment ajouter de façon dynamique deux coeurs complets de CPU au domaine `ldg1`. Le domaine `ldg1` est un domaine actif configuré avec deux coeurs complets de CPU. La première commande montre que le domaine `ldg1` est actif. La seconde commande indique que le domaine `ldg1` est configuré avec des coeurs complets de CPU et avec un maximum de quatre coeurs : Les troisième et cinquième commandes indiquent les coeurs de CPU assignés au domaine avant et après l'ajout de deux coeurs complets de CPU. La quatrième commande ajoute de façon dynamique deux coeurs complets de CPU au domaine `ldg1`.

```
primary# ldm list ldg1
NAME    STATE   FLAGS   CONS  VCPU  MEMORY UTIL  UPTIME
ldg1    active  -n----  5000  16    2G     0.4%  5d 17h 49m
primary# ldm list -o resmgmt ldg1
NAME
ldg1

CONSTRAINT
   whole-core
        max-cores=4
primary# ldm list -o core ldg1
NAME
ldg1

CORE
CID PCPUSET
1 (8, 9, 10, 11, 12, 13, 14, 15)
2 (16, 17, 18, 19, 20, 21, 22, 23)
primary# ldm add-core 2 ldg1
primary# ldm list -o core ldg1
NAME
ldg1

CORE
CID PCPUSET
1 (8, 9, 10, 11, 12, 13, 14, 15)
2 (16, 17, 18, 19, 20, 21, 22, 23)
3 (24, 25, 26, 27, 28, 29, 30, 31)
4 (32, 33, 34, 35, 36, 37, 38, 39)
```

Gestion dynamique des ressources de la CPU

La gestion dynamique des ressources (DRM) permet de gérer automatiquement les ressources CPU sur certains domaines.

Gestion de l'alimentation

Vous pouvez définir une stratégie de gestion de l'alimentation (PM) séparée pour chaque domaine de partition forcée.

Réinitialisation ou nouvelle association de domaine

Un domaine configuré avec des coeurs complets de CPU reste configuré avec des coeurs complets de CPU lorsque le domaine est redémarré ou lorsque l'ensemble du système est redémarré. Un domaine utilise les mêmes coeurs de CPU physiques pendant toute la durée de l'association. Par exemple, si un domaine est réinitialisé, il utilise les mêmes coeurs de CPU physiques avant et après la réinitialisation. De même, si le système entier est mis hors tension alors qu'un domaine est associé, ce domaine sera configuré avec les mêmes coeurs de CPU physiques lorsque le système est remis sous tension. Si vous dissociez puis associez à nouveau un domaine, ou si l'ensemble du système est redémarré avec une nouvelle configuration, il est possible que le domaine utilise des coeurs de CPU physiques différents.

Affectation de ressources physiques à des domaines

Logical Domains Manager sélectionne automatiquement les ressources physiques à affecter à un domaine. Le logiciel Oracle VM Server for SPARC 3.4 permet également aux administrateurs experts de sélectionner de façon explicite les ressources physiques à affecter ou dont l'affectation à un domaine doit être annulée.

Les ressources affectées de manière explicite sont appelées *ressources nommées*. Les ressources affectées de manière automatique sont appelées *ressources anonymes*.

Attention - N'affectez *pas* de ressources nommées à moins que vous soyez un administrateur expert.

Vous pouvez affecter de façon explicite des ressources physiques au domaine de contrôle et à des domaines invités. Etant donné que le domaine de contrôle reste actif, il peut éventuellement être en mode de reconfiguration retardée avant que vous ne procédiez à des affectations de ressources physiques. Ou une reconfiguration retardée est automatiquement déclenché lorsque vous effectuez des affectations physiques. Reportez-vous à la section "Gestion des ressources physiques sur le domaine de contrôle" à la page 363. Pour plus d'informations sur les restrictions applicables aux ressources physiques, reportez-vous à la section "Restrictions applicables à la gestion des ressources physiques sur les domaines" à la page 363.

Vous pouvez affecter de façon explicite les ressources physiques suivantes au domaine de contrôle et à des domaines invités :

- **CPU physiques.** Affectez les ID des coeurs physiques au domaine en définissant la propriété `cid`.

  La propriété `cid` doit *uniquement* être utilisée par un administrateur au fait de la topologie du système à configurer. Cette fonction de configuration avancée applique des règles d'allocation particulières et peut avoir un impact sur les performances globales du système.

  Vous pouvez définir cette propriété en exécutant l'une des commandes suivantes :

  ```
  ldm add-core cid=core-ID[,core-ID[,...]] domain-name

  ldm set-core cid=core-ID[,core-ID[,...]] domain-name

  ldm rm-core [-f] cid=core-ID[,core-ID[,...]] domain-name
  ```

  Si vous spécifiez un ID de coeur en tant que valeur de la propriété `cid`, *core-ID* est explicitement affecté au domaine ou supprimé du domaine.

  Remarque - Vous ne pouvez pas utiliser la commande `ldm add-core` pour ajouter des ressources de coeur nommées à un domaine qui utilise déjà d'autres ressources anonymes.

- **Mémoire physique.** Affectez un ensemble de régions contiguës de la mémoire physique à un domaine en définissant la propriété `mblock`. Chaque région de mémoire physique est spécifiée sous la forme d'une adresse de mémoire physique et d'une taille.

  La propriété `mblock` doit *uniquement* être utilisée par un administrateur au fait de la topologie du système à configurer. Cette fonction de configuration avancée applique des règles d'allocation particulières et peut avoir un impact sur les performances globales du système.

  Vous pouvez définir cette propriété en exécutant l'une des commandes suivantes :

  ```
  ldm add-mem mblock=PA-start:size[,PA-start:size[,...]] domain-name

  ldm set-mem mblock=PA-start:size[,PA-start:size[,...]] domain-name
  ```

```
ldm rm-mem mblock=PA-start:size[,PA-start:size[,...]] domain-name
```

Pour affecter un bloc de mémoire à un domaine ou l'en supprimer, définissez la propriété `mblock`. Une valeur correcte comprend une adresse de départ de mémoire physique (*PA-start*) et une taille de bloc de mémoire (*size*) séparées par le signe deux-points (`:`).

Remarque - Vous ne pouvez pas utiliser la reconfiguration dynamique (DR) pour déplacer des ressources de mémoire ou de coeur entre des domaines en cours d'exécution lorsque vous définissez la propriété `mblock` ou `cid`. Pour déplacer des ressources entre des domaines, assurez-vous que les domaines ont l'état associé ou inactif. Pour plus d'informations sur la gestion des ressources physiques dans le domaine de contrôle, reportez-vous à la section "Gestion des ressources physiques sur le domaine de contrôle" à la page 363.

Remarque - Si vous migrez un domaine, les ressources nommées que vous avez affectées à l'aide des propriétés `cid` et `mblock` sont supprimées. Le domaine utilise à la place les ressources anonymes sur le système cible.

Vous pouvez utiliser la commande `ldm list-constraints` pour visualiser les contraintes de ressources de domaines. La contrainte `physical-bindings` spécifie les types de ressources physiquement assignés à un domaine. Lorsqu'un domaine est créé, la contrainte `physical-bindings` n'est pas définie tant qu'aucune ressource physique n'est affectée au domaine.

La contrainte `physical-bindings` est définie sur des valeurs données dans les cas suivants :

- `memory` lorsque la propriété `mblock` est spécifiée
- `core` lorsque la propriété `cid` est spécifiée
- `core,memory` lorsque les propriétés `cid` et `mblock` sont spécifiées

▼ Procédure de suppression de la contrainte `physical-bindings`

Pour supprimer la contrainte `physical-bindings` pour un domaine invité, vous devez d'abord supprimer toutes les ressources liées physiquement.

1. **Dissociez le domaine.**

```
primary# ldm unbind domain-name
```

2. **Supprimez les ressources nommées.**

- Pour supprimer les coeurs nommés :

  ```
  primary# ldm set-core cid= domain-name
  ```

- Pour supprimer la mémoire nommée :

  ```
  primary# ldm set-mem mblock= domain-name
  ```

3. **Ajoutez des ressources de CPU ou de mémoire.**

   - Pour ajouter une ressource de CPU :

     ```
     primary# ldm add-vcpu number domain-name
     ```

   - Pour ajouter une ressource de mémoire :

     ```
     primary# ldm add-mem size[unit] domain-name
     ```

4. **Associez à nouveau le domaine.**

   ```
   primary# ldm bind domain-name
   ```

Procédure de suppression de toutes les ressources non associées physiquement

Pour contraindre les domaines invités qui n'ont pas la contrainte `physical-bindings` activée, vous devez d'abord supprimer toutes les ressources non liées physiquement.

1. **Dissociez le domaine.**

   ```
   primary# ldm unbind domain-name
   ```

2. **Définissez le nombre de ressource sur `0`.**

   - Pour définir la ressource de CPU :

     ```
     primary# ldm set-core 0 domain-name
     ```

   - Pour définir la ressource de mémoire :

     ```
     primary# ldm set-mem 0 domain-name
     ```

3. **Ajoutez des ressources de CPU ou de mémoire physiquement associées.**

   - Pour ajouter une ressource de CPU :

     ```
     primary# ldm add-core cid=core-ID domain-name
     ```

   - Pour ajouter une ressource de mémoire :

```
primary# ldm add-mem mblock=PA-start:size domain-name
```

4. **Associez à nouveau le domaine.**

```
primary# ldm bind domain-name
```

Gestion des ressources physiques sur le domaine de contrôle

Pour appliquer ou supprimer la contrainte `physical-bindings` du domaine de contrôle, suivez les étapes appropriées décrites dans la section précédente. Cependant, au lieu de dissocier le domaine, placez le domaine de contrôle dans une reconfiguration retardée.

Une modification de contrainte entre les ressources anonymes et les ressources nommées physiquement associées déclenche automatiquement une reconfiguration retardée. Vous pouvez tout de même paramétrer une reconfiguration retardée à l'aide de la commande `ldm start-reconf primary`.

Comme pour toute modification liée à une reconfiguration retardée, vous devez réinitialiser le domaine (le domaine de contrôle dans ce cas) pour terminer le processus.

Remarque - Lorsque le domaine de contrôle est en mode de reconfiguration retardée, vous pouvez effectuer un nombre illimité d'affectations de mémoire en exécutant les commandes `ldm add-mem` et `ldm rm-mem` sur le domaine de contrôle. Toutefois, vous ne pouvez procéder qu'à *une seule* affectation de coeur au domaine de contrôle à l'aide de la commande `ldm set-core`.

Restrictions applicables à la gestion des ressources physiques sur les domaines

Les restrictions suivantes s'appliquent à l'affectation de ressources physiques :

- Vous ne pouvez pas effectuer de liaisons de mémoire physique et non physique dans le même domaine, pas plus que vous ne pouvez effectuer de liaisons de coeur physique et non physique.
- Un domaine peut comporter des liaisons de mémoire non physique et des liaisons de coeur physique, ou encore des liaisons de coeur non physique et des liaisons de mémoire physique.

- Lorsque vous ajoutez une ressource physique à un domaine, le type de ressource correspondant est limité à l'état de liaison physique.
- Toute tentative d'ajouter ou de supprimer des CPU anonymes dans un domaine où `physical-bindings=core` est vouée à l'échec.
- Pour les ressources non liées, l'allocation et la vérification des ressources peut *uniquement* être effectuée lors de l'exécution de la commande `ldm bind`.
- Lorsque vous retirez de la mémoire physique d'un domaine, vous devez retirer *précisément* le bloc de mémoire physique précédemment ajouté.
- Les plages de mémoire physique *ne doivent pas* se chevaucher.
- Vous pouvez uniquement utiliser la commande `ldm add-core cid=` ou `ldm set-core cid=` pour assigner une ressource physique à un domaine.
- Si vous utilisez la commande `ldm add-mem mblock=` ou `ldm set-mem mblock=` pour assigner des blocs de mémoire physique multiples, les adresses et les tailles sont vérifiées immédiatement pour éviter des collisions avec d'autres liaisons.
- Un domaine auquel des coeurs partiels sont affectés peut utiliser la sémantique whole-core si les CPU restantes de ces coeurs sont libres et disponibles.

Utilisation de la reconfiguration dynamique de la mémoire

La reconfiguration dynamique (DR) de la mémoire repose sur la capacité et vous permet d'ajouter ou de supprimer une quantité quelconque de mémoire dans un domaine logique actif.

Voici les contraintes et les restrictions d'utilisation de la fonction de reconfiguration dynamique de mémoire :

- Vous pouvez effectuer des opérations de reconfiguration dynamique sur tous les domaines. Cependant, une seule opération de DR de mémoire peut être en cours sur un domaine à un moment donné.
- La fonction de DR de mémoire force un alignement de 256 Mo sur l'adresse et la taille de la mémoire impliquée dans une opération donnée. Voir la section "Alignement de la mémoire" à la page 367.
- La mémoire non alignée dans le pool de mémoire libre ne peut pas être assignée à un domaine à l'aide de la fonction de reconfiguration dynamique de mémoire. Reportez-vous à la section "Ajout de mémoire non alignée" à la page 368.

Si la mémoire d'un domaine ne peut pas être reconfigurée à l'aide d'une opération de DR de mémoire, le domaine doit être arrêté avant que la mémoire ne soit configurée. Si le domaine est le domaine de contrôle, vous devez d'abord lancer une reconfiguration retardée.

Dans certaines circonstances, Logical Domains Manager arrondit l'allocation de mémoire requise au multiple supérieur suivant de 8 kilo-octets ou 4 mégaoctets. L'exemple suivant

présente une sortie de la commande `ldm list-domain -l`, où la valeur de contrainte est inférieure à la taille réelle allouée :

```
Memory:
          Constraints: 1965 M
          raddr          paddr5          size
          0x1000000      0x291000000     1968M
```

Ajout de mémoire

Si un domaine est actif, vous pouvez utiliser la commande `ldm add-memory` pour ajouter de la mémoire de manière dynamique au domaine. La commande `ldm set-memory` ajoute également de manière dynamique de la mémoire si la taille de la mémoire indiquée est supérieure à la taille de la mémoire actuelle du domaine.

Suppression de mémoire

Si un domaine est actif, vous pouvez utiliser la commande `ldm remove-memory` pour supprimer de la mémoire de manière dynamique au domaine. La commande `ldm set-memory` supprime également de manière dynamique de la mémoire si la taille de la mémoire indiquée est inférieure à la taille de la mémoire actuelle du domaine.

La suppression de mémoire peut être une opération nécessitant beaucoup de temps. Vous pouvez suivre la progression d'une commande `ldm remove-memory` en exécutant la commande `ldm list -l` pour le domaine indiqué.

Vous pouvez annuler une demande de suppression en cours en interrompant la commande `ldm remove-memory` command (en appuyant sur Control-C) ou en émettant la commande `ldm cancel-operation memdr`. Si vous annulez une demande de suppression de mémoire, seule la partie restante de la demande de suppression est affectée, c'est-à-dire la quantité de mémoire devant encore être supprimée du domaine.

Demandes partielles de reconfiguration dynamique de mémoire

Il est possible qu'une demande d'ajout ou de suppression dynamique de mémoire dans un domaine ne soit pas effectuée dans sa totalité. Les résultats dépendent des capacités d'ajout et de suppression de la mémoire.

Remarque - La mémoire est effacée après sa suppression d'un domaine et avant d'être ajoutée à un autre domaine.

Reconfiguration de la mémoire du domaine de contrôle

Vous pouvez utiliser la fonction de reconfiguration dynamique de la mémoire pour reconfigurer la mémoire du domaine de contrôle. Si une demande de DR de mémoire ne peut pas être effectuée sur le domaine de contrôle, vous devez d'abord lancer une reconfiguration retardée.

L'utilisation de la reconfiguration dynamique de mémoire peut ne pas être adaptée à la suppression de grandes quantités de mémoire sur un domaine actif, car les opérations de reconfiguration dynamique de mémoire prennent beaucoup de temps. Surtout au cours de la configuration initiale du système, vous devez utiliser la reconfiguration retardée pour réduire la mémoire dans le domaine de contrôle.

Réduction de la mémoire du domaine de contrôle

Utilisez une reconfiguration retardée au lieu d'une reconfiguration dynamique de mémoire pour réduire la mémoire du domaine de contrôle dans une configuration usine par défaut initiale. Dans un tel cas, le domaine de contrôle possède toute la mémoire du système hôte. La fonction de reconfiguration dynamique de la mémoire n'est pas adaptée à cet objectif, car il n'est pas certain qu'un domaine actif ajoute, ou plus généralement qu'il supprime toute la mémoire demandée. Au lieu de cela, le SE s'exécutant sur ce domaine fait de son mieux pour remplir la demande. Par ailleurs, la suppression de mémoire peut être une opération de longue durée. Ces problèmes sont amplifiés lorsque des opérations importantes sur la mémoire sont impliquées, comme c'est le cas pour la réduction initiale de la mémoire du domaine de contrôle.

Pour ces raisons, utilisez une reconfiguration retardée en procédant comme suit :

1. Utilisez la commande `ldm start-reconf primary` pour mettre le domaine de contrôle en mode de reconfiguration retardée.
2. Partitionnez les ressources du système hôte qui détenues par le domaine de contrôle, si nécessaire.
3. Utilisez la commande `ldm cancel-reconf` pour annuler les opérations de l'étape 2, si nécessaire, et recommencez.
4. Réinitialisez le domaine de contrôle pour appliquer les modifications.

Reconfiguration dynamique et retardée

Si une reconfiguration retardée est en attente sur le domaine de contrôle, une demande de reconfiguration de la mémoire est rejetée pour tous les autres domaines. Si aucune reconfiguration retardée n'est en attente dans le domaine de contrôle, la demande de reconfiguration de la mémoire est rejetée pour tout domaine qui ne prend pas en charge la reconfiguration dynamique de la mémoire. Pour ces domaines, la demande est convertie en demande de reconfiguration retardée.

Alignement de la mémoire

Les demandes de reconfiguration de la mémoire ont différentes contraintes d'alignement qui dépendent de l'état du domaine sur lequel la demande est appliquée.

Alignement de la mémoire pour les domaines actifs

- **Ajout et suppression dynamiques.** L'adresse et la taille d'un bloc de mémoire sont alignées à 256 Mo pour un ajout et une suppression dynamiques. La taille de fonctionnement minimale est de 256 Mo.

  Une demande non alignée ou une demande de suppression supérieure à la taille associée est rejetée.

  Utilisez les commandes suivantes pour ajuster les allocations de mémoire :

  - `ldm add-memory`**.** Si vous spécifiez l'option `--auto-adj` avec cette commande, la quantité de mémoire à ajouter est alignée à 256 Mo, ce qui peut accroître la quantité de mémoire effectivement ajoutée au domaine.
  - `ldm remove-memory`**.** Si vous spécifiez l'option `--auto-adj` avec cette commande, la quantité de mémoire à supprimer est alignée à 256 Mo, ce qui peut diminuer la quantité de mémoire effectivement supprimée du domaine.
  - `ldm set-memory`**.** Cette commande est traitée comme une opération d'ajout ou de suppression. Si vous indiquez l'option `--auto-adj`, la quantité de mémoire à ajouter ou à supprimer est alignée sur 256 Mo comme décrit précédemment. Notez que cet alignement peut augmenter la taille de la mémoire obtenue pour le domaine.
- **Reconfiguration retardée.** L'adresse et la taille d'un bloc de mémoire sont alignées à 4 Mo. Si vous faites une demande non alignée, la demande est arrondie à un alignement à 4 Mo.

Alignement de la mémoire pour les domaines liés

L'adresse et la taille d'un bloc de mémoire sont alignées à 4 Mo pour les domaines liés. Si vous faites une demande non alignée, la demande est arrondie à un alignement à 4 Mo. Par conséquent, la taille de la mémoire obtenue pour le domaine peut être supérieure à celle indiquée.

Pour les commandes `ldm add-memory`, `ldm set-memory` et `ldm remove-memory`, l'option `--auto-adj` arrondit la taille de la mémoire obtenue sur 256 Mo. Par conséquent, la taille de la mémoire obtenue peut être supérieure à celle indiquée.

Alignement de la mémoire pour les domaines inactifs

Pour les commandes `ldm add-memory`, `ldm set-memory` et `ldm remove-memory`, l'option `--auto-adj` arrondit la taille de la mémoire obtenue sur 256 Mo. Il n'y a aucune contrainte d'alignement pour un domaine inactif. Les restrictions décrites à la section "Alignement de la mémoire pour les domaines liés" à la page 368 prennent effet dès qu'un tel domaine est associé.

Ajout de mémoire non alignée

La fonction de reconfiguration dynamique de la mémoire force un alignement de mémoire de 256 Mo sur l'adresse et la taille de la mémoire qui est ajoutée de manière dynamique à un domaine actif ou supprimée de ce dernier. Par conséquent, la mémoire non alignée dans un domaine actif ne peut pas être supprimée à l'aide de la reconfiguration dynamique de la mémoire.

De même, toute mémoire non alignée dans le pool de mémoire libre ne peut pas être ajoutée à un domaine actif à l'aide de la reconfiguration dynamique de la mémoire.

Après que toute la mémoire alignée a été allouée, vous pouvez utiliser la commande `ldm add-memory` pour ajouter la mémoire restante non alignée à un domaine lié ou inactif. Vous pouvez également utiliser cette commande pour ajouter la mémoire non alignée restante au domaine de contrôle au moyen d'une opération de reconfiguration retardée.

L'exemple suivant montre comment ajouter les deux blocs de mémoire de 128 Mo restants aux domaines `primary` et `ldg1`. Le domaine `ldom1` est à l'état associé.

La commande suivante lance une opération de reconfiguration retardée sur le domaine de contrôle.

```
primary# ldm start-reconf primary
Initiating a delayed reconfiguration operation on the primary domain.
All configuration changes for other domains are disabled until the
primary domain reboots, at which time the new configuration for the
primary domain also takes effect.
```

La commande suivante ajoute l'un des blocs de mémoire de 128 Mo au domaine de contrôle.

```
primary# ldm add-memory 128M primary
------------------------------------------------------------------------------
Notice: The primary domain is in the process of a delayed reconfiguration.
Any changes made to the primary domain will only take effect after it reboots.
------------------------------------------------------------------------------

primary# ldm list
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -ndcv-  SP      8     2688M    0.1%  23d 8h 8m

primary# ldm list
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  SP      8     2560M    0.5%  23d 8h 9m
ldom1            bound      ------  5000    1     524M
```

La commande suivante ajoute l'autre bloc de mémoire de 128 Mo au domaine `ldom1`.

```
primary# ldm add-mem 128M ldom1
primary# ldm list
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  SP      8     2560M    0.1%  23d 8h 9m
ldom1            bound      ------  5000    1     652M
```

Exemples de reconfiguration dynamique de mémoire

Les exemples suivants montrent comment effectuer des opérations de reconfiguration dynamique de la mémoire. Pour plus d'informations sur les commandes de la CLI correspondantes, reportez-vous à la page de manuel `ldm`(1M).

EXEMPLE 66 Opérations de reconfiguration dynamique de la mémoire sur des domaines actifs

Cet exemple montre comment ajouter de la mémoire de manière dynamique et comment la supprimer d'un domaine actif, `ldom1`.

La sortie `ldm list` montre la mémoire pour chaque domaine dans la zone Memory.

```
primary# ldm list
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  SP      4     27392M   0.4%  1d 22h 53m
ldom1            active     -n----  5000    2     2G       0.4%  1d 1h 23m
ldom2            bound      ------  5001    2     200M
```

La commande `ldm add-mem` se solde par une erreur, car vous devez indiquer la mémoire en multiples de 256 Mo. La commande suivante `ldm add-mem` utilise l'option `--auto-adj` pour que la quantité de mémoire soit arrondie à 256 Mo bien que vous indiquiez `200M` comme la quantité de mémoire à ajouter.

```
primary# ldm add-mem 200M ldom1
The size of memory must be a multiple of 256MB.

primary# ldm add-mem --auto-adj 200M ldom1
Adjusting request size to 256M.
The ldom1 domain has been allocated 56M more memory
than requested because of memory alignment constraints.

primary# ldm list
NAME             STATE   FLAGS   CONS    VCPU    MEMORY  UTIL    UPTIME
primary          active  -n-cv-  SP      4       27392M  5.0%    8m
ldom1            active  -n----  5000    2       2304M   0.5%    1m
ldom2            bound   ------  5001    2       200M
```

La commande `ldm rm-mem` existe avec une erreur, car vous devez indiquer la mémoire en multiples de 256 Mo. Lorsque vous ajoutez l'option `--auto-adj` à cette même commande, la suppression de mémoire aboutit, car la quantité de mémoire est arrondie à la limite suivante de 256 Mo.

```
primary# ldm rm-mem --auto-adj 300M ldom1
Adjusting requested size to 256M.
The ldom1 domain has been allocated 44M more memory
than requested because of memory alignment constraints.

primary# ldm list
NAME             STATE   FLAGS   CONS    VCPU    MEMORY  UTIL    UPTIME
primary          active  -n-cv-  SP      4       27392M  0.3%    8m
ldom1            active  -n----  5000    2       2G      0.2%    2m
ldom2            bound   ------  5001    2       200M
```

EXEMPLE 67 Opérations de reconfiguration dynamique de la mémoire sur des domaines liés

Cet exemple montre comment ajouter de la mémoire de manière dynamique et comment la supprimer d'un domaine lié, `ldom2`.

La sortie `ldm list` montre la mémoire pour chaque domaine dans la zone Memory. La première commande `ldm add-mem` ajoute 100 Mo de mémoire au domaine `ldom2`. La commande suivante `ldm add-mem` définit l'option `--auto-adj`, qui provoque l'ajout dynamique de 112 Mo supplémentaires à `ldom2`.

La commande `ldm rm-mem` supprime de manière dynamique 100 Mo au domaine `ldom2`. Si vous indiquez l'option `--auto-adj` à cette même commande pour supprimer 300 Mo de mémoire, la quantité de mémoire est arrondie à la limite inférieure suivante de 256 Mo.

```
primary# ldm list
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
```

```
primary          active     -n-cv-  SP      4     27392M   0.4%  1d 22h 53m
ldom1            active     -n----  5000    2     2G       0.4%  1d 1h 23m
ldom2            bound      ------  5001    2     200M

primary# ldm add-mem 100M ldom2

primary# ldm list
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  SP      4     27392M   0.5%  1d 22h 54m
ldom1            active     -n----  5000    2     2G       0.2%  1d 1h 25m
ldom2            bound      ------  5001    2     300M

primary# ldm add-mem --auto-adj 100M ldom2
Adjusting request size to 256M.
The ldom2 domain has been allocated 112M more memory
than requested because of memory alignment constraints.

primary# ldm list
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  SP      4     27392M   0.4%  1d 22h 55m
ldom1            active     -n----  5000    2     2G       0.5%  1d 1h 25m
ldom2            bound      ------  5001    2     512M

primary# ldm rm-mem 100M ldom2
primary# ldm list
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  SP      4     27392M   3.3%  1d 22h 55m
ldom1            active     -n----  5000    2     2G       0.2%  1d 1h 25m
ldom2            bound      ------  5001    2     412M

primary# ldm rm-mem --auto-adj 300M ldom2
Adjusting request size to 256M.
The ldom2 domain has been allocated 144M more memory
than requested because of memory alignment constraints.

primary# ldm list
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  SP      4     27392M   0.5%  1d 22h 55m
ldom1            active     -n----  5000    2     2G       0.2%  1d 1h 26m
ldom2            bound      ------  5001    2     256M
```

EXEMPLE 68 Définition des tailles de mémoire du domaine

Cet exemple montre comment utiliser la commande `ldm set-memory` pour ajouter de la mémoire à un domaine et en supprimer.

La sortie `ldm list` montre la mémoire pour chaque domaine dans la zone Memory.

```
primary# ldm list
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  SP      4     27392M   0.5%  1d 22h 55m
ldom1            active     -n----  5000    2     2G       0.2%  1d 1h 26m
ldom2            bound      ------  5001    2     256M
```

La commande `ldm set-mem` tente de définir la taille du domaine `primary` à 3 400 Mo. L'erreur résultant indique que la valeur indiquée ne se trouve pas dans la limite de 256 Mo. L'ajout de l'option `--auto-adj` à cette même commande vous permet de supprimer avec succès de

la mémoire et de rester dans la limite de 256 Mo. Cette commande émet également un avertissement pour indiquer que la totalité de la mémoire demandée n'a pas pu être supprimée, car le domaine utilise cette mémoire.

```
primary# ldm set-mem 3400M primary
An ldm set-mem 3400M command would remove 23992MB, which is not a multiple
of 256MB. Instead, run ldm rm-mem 23808MB to ensure a 256MB alignment.

primary# ldm set-mem --auto-adj 3400M primary
Adjusting request size to 3.4G.
The primary domain has been allocated 184M more memory
than requested because of memory alignment constraints.
Only 9472M of memory could be removed from the primary domain
because the rest of the memory is in use.
```

La commande `ldm set-mem` définit la taille de la mémoire du domaine `ldom2`, qui est à l'état lié, à 690 Mo. Si vous ajoutez l'option `--auto-adj` à cette même commande, 78 Mo supplémentaires de mémoire sont ajoutés de manière dynamique à `ldom2` pour rester dans une limite de 256 Mo.

```
primary# ldm set-mem 690M ldom2
primary# ldm list
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  SP      4     17920M   0.5%  1d 22h 56m
ldom1            active     -n----  5000    2     2G       0.6%  1d 1h 27m
ldom2            bound      ------  5001    2     690M

primary# ldm set-mem --auto-adj 690M ldom2
Adjusting request size to 256M.
The ldom2 domain has been allocated 78M more memory
than requested because of memory alignment constraints.

primary# ldm list
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  SP      4     17920M   2.1%  1d 22h 57m
ldom1            active     -n----  5000    2     2G       0.2%  1d 1h 27m
ldom2            bound      ------  5001    2     768M
```

Utilisation de groupes de ressources

Un *groupe de ressources*constitue une alternative pour afficher les ressources d'un système. Les ressources sont groupées en fonction des relations physiques entre noyaux de processeur, mémoire et bus d'E/S. Différentes plates-formes, et même différentes configurations de plates-formes dans une même famille de serveurs comme SPARC T5-2 and SPARC T5-8, peuvent avoir différents groupes de ressources qui reflètent les différences sur le matériel. Affichez les informations sur le groupe de ressources à l'aide de la commande `ldm list-resrc-group`.

L'appartenance des groupes de ressources est définie de façon statique par la configuration matérielle. Vous pouvez utiliser les commandes `ldm remove-core` et `ldm remove-memory` pour travailler sur les ressources à partir d'un groupe de ressources en particulier.

- La sous-commande `remove-core` indique le nombre de coeurs de CPU à supprimer du domaine. Lorsque vous définissez un groupe de ressources à l'aide de l'option `-g`, les coeurs sélectionnés en vue de leur suppression proviennent tous de ce groupe de ressources.
- La sous-commande `remove-memory` supprime la quantité de mémoire spécifiée à partir d'un domaine logique. Lorsque vous définissez un groupe de ressources à l'aide de l'option `-g`, la mémoire sélectionnée en vue de sa suppression provient toute de ce groupe de ressources.

Pour plus d'informations sur ces commandes, reportez-vous à la page de manuel`ldm`(1M).

Reportez-vous à la section "Liste des informations sur le groupe de ressources" à la page 382 pour consulter des exemples.

Exigences et restrictions des groupes de ressources

La fonctionnalité de groupe de ressources est disponible uniquement sur les serveurs SPARC T5, les serveurs de série SPARC T7, les serveurs SPARC M5 et SPARC M6, les serveurs des séries SPARC M7 et SPARC S7 et les Serveurs Fujitsu M10.

La fonctionnalité de groupe de ressources est soumise aux restrictions suivantes :

- Elle n'est pas disponible sur les plates-formes UltraSPARC T2, UltraSPARC T2 Plus, SPARC T3 et SPARC T4.
- La commande `ldm list-rsrc-group` n'affiche aucune information sur ces plates-formes non prises en charge. Les variantes`-g` des commandes `ldm remove-core` et `ldm remove-memory` ne sont pas fonctionnelles.
- Sur les plates-formes prises en charge, la spécification de `_sys_`à la place de *domain-name* déplace toute la mémoire système vers la mémoire libre dans un groupe de ressources différent. Cette commande n'est pas opérationnelle sur les plates-formes non prises en charge.

Utilisation de la gestion de l'alimentation

Pour activer la gestion de l'alimentation (PM), vous devez d'abord définir la stratégie PM au moins dans la version 3.0 du microprogramme ILOM. Cette section récapitule les informations nécessaires afin de pouvoir utiliser le mode PM avec le logiciel Oracle VM Server for SPARC.

Pour plus d'informations sur les fonctions de PM et ILOM, consultez les sections suivantes :

- Annexe A, Utilisation de la gestion de l'alimentation

- “Surveillance de la consommation d'énergie” du *Oracle Integrated Lights Out Manager (ILOM) 3.0 CLI Procedures Guide*
- *Mises à jour des fonctions et notes de version d'Oracle Integrated Lights Out Manger (ILOM) 3.0*

Utilisation de la gestion dynamique des ressources

Vous pouvez utiliser des stratégies pour déterminer la manière d'exécuter automatiquement des activités DR. A ce moment, vous pouvez *uniquement* créer des stratégies pour régir la gestion dynamique des ressources des CPU virtuelles.

Attention - Les restrictions suivantes affectent la gestion dynamique des ressources (DRM) de la CPU :

- Sur les plates-formes UltraSPARC T2 et UltraSPARC T2 Plus, la stratégie DRM ne peut pas être activée lorsque la stratégie élastique PM est définie.
- Sur les plates-formes UltraSPARC T2 et UltraSPARC T2 Plus, tout passage de la stratégie de performance à la stratégie élastique est retardé lorsque la DRM est activée.
- Vérifiez que vous avez désactivé la CPU DRM avant d'effectuer une opération de migration de domaine. Dans le cas contraire, un message d'erreur s'affiche.
- Lorsque la stratégie élastique PM est définie, vous pouvez uniquement utiliser la DRM si le microprogramme prend en charge l'utilisation normalisée (8.2.0).

Une *stratégie de gestion des ressources* définit dans quelles conditions des CPU virtuelles peuvent être ajoutées automatiquement à un domaine logique et supprimées de celui-ci. Une stratégie est gérée à l'aide des commandes `ldm add-policy`, `ldm set-policy` et `ldm remove-policy` :

```
ldm add-policy [enable=yes|no] [priority=value] [attack=value] [decay=value]
  [elastic-margin=value] [sample-rate=value] [tod-begin=hh:mm[:ss]]
  [tod-end=hh:mm[:ss]] [util-lower=percent] [util-upper=percent] [vcpu-min=value]
  [vcpu-max=value] name=policy-name  domain-name...
ldm set-policy [enable=[yes|no]] [priority=[value]] [attack=[value]] [decay=[value]]
  [elastic-margin=[value]] [sample-rate=[value]] [tod-begin=[hh:mm:ss]]
  [tod-end=[hh:mm:ss]] [util-lower=[percent]] [util-upper=[percent]] [vcpu-min=[value]]
  [vcpu-max=[value]] name=policy-name  domain-name...
ldm remove-policy [name=]policy-name...  domain-name
```

Pour plus d'informations sur ces commandes et sur la création de stratégies de gestion des ressources, reportez-vous à la page de manuel `ldm`(1M).

Une stratégie est en vigueur pendant la durée indiquée par les propriétés `tod-begin` et `tod-end`. L'heure spécifiée par `tod-begin` doit être antérieure à l'heure spécifiée par `tod-end` au cours d'une

période de 24 heures. Par défaut, les valeurs des propriétés `tod-begin` et `tod-end` sont 00:00:00 et 23:59:59, respectivement. Lorsque les valeurs par défaut sont utilisées, la stratégie est toujours active.

La stratégie utilise la valeur de la propriété `priority` pour spécifier une priorité pour la stratégie de gestion dynamique des ressources (DRM). Les valeurs de priorité sont utilisées pour déterminer la relation entre les stratégies DRM sur un domaine unique et entre les domaines DRM sur un système unique. Les valeurs numériques inférieures représentent les priorités supérieures (meilleures). Les valeurs valides sont comprises entre 1 et 9 999. La valeur par défaut est 99.

Le comportement de la propriété `priority` dépend de la disponibilité d'un pool de ressources de CPU libres, comme suit :

- **Des ressources de CPU libres sont disponibles dans le pool.** Dans ce cas, la propriété `priority` détermine quelle stratégie DRM sera appliquée lorsque plusieurs stratégies qui se chevauchent sont définies pour un même domaine.
- **Aucune ressource de CPU libre n'est disponible dans le pool.** Dans ce cas, la propriété `priority` spécifie si une ressource peut être déplacée dynamiquement d'un domaine de priorité inférieure vers un domaine de priorité supérieure sur le même système. La priorité d'un domaine est celle spécifiée par la stratégie DRM en effet pour ce domaine.

  Par exemple, un domaine de priorité supérieure peut acquérir des ressources de CPU à partir d'un autre domaine qui possède une stratégie DRM avec une priorité inférieure. Cette capacité d'acquisition de ressources s'applique uniquement aux domaines sur lesquels des stratégies DRM sont activées. Les domaines dont les valeurs `priority` sont identiques ne sont pas affectés par cette capacité. Ainsi, si la priorité par défaut est utilisée pour toutes les stratégies, les domaines ne peuvent pas obtenir de ressources des domaines de priorité inférieure. Pour tirer parti de cette fonction, réglez les valeurs des propriétés `priority` afin qu'elles soient différentes.

Par exemple, des stratégies DRM sont activées sur les domaines `ldg1` et `ldg2`. La propriété `priority` du domaine `ldg1` est `1`, qui est plus favorable que la valeur de propriété `priority` du domaine `ldg2` (`2`). Le domaine `ldg1` peut supprimer dynamiquement une ressource de CPU du domaine `ldg2` et se l'assigner dans les cas suivants :

- Le domaine `ldg1` requiert une autre ressource de CPU.
- Le pool de ressources de CPU disponibles est épuisé.

La stratégie utilise les valeurs de la propriété `util-high` et `util-low` pour définir les seuils haut et bas d'utilisation de la CPU. Si l'utilisation dépasse la valeur de `util-high`, des CPU virtuelles sont ajoutées au domaine jusqu'à ce que le nombre soit compris entre les valeurs `vcpu-min` et `vcpu-max`. Si l'utilisation est inférieure à la valeur `util-low`, les CPU virtuelles sont supprimées du domaine jusqu'à ce que le nombre soit compris entre les valeurs `vcpu-min` et `vcpu-max`. Si `vcpu-min` est atteint, plus aucune CPU virtuelle ne peut être supprimée de manière dynamique. Si `vcpu-max` est atteint, plus aucune CPU virtuelle ne peut être ajoutée de manière dynamique.

EXEMPLE 69 Ajout de stratégies de gestion des ressources

Par exemple, après avoir observé l'utilisation classique de vos systèmes pendant plusieurs semaines, vous pouvez définir des stratégies pour optimiser l'utilisation des ressources. L'utilisation la plus importante s'effectue tous les jours entre 9 heures et 18 heures, heure du Pacifique, et l'utilisation la plus faible a lieu tous les jours de 18 heures à 9 heures.

En fonction de cette observation de l'utilisation du système, vous décidez de créer les stratégies d'utilisation élevée et basse et fonction de l'utilisation globale de votre système :

- **Elevée :** Tous les jours de 9 heures à 18 heures, heure du Pacifique
- **Basse :** Tous les jours de 18 heures à 9 heures, heure du Pacifique

La commande `ldm add-policy` suivante créer la stratégie `high-usage` à utiliser au cours de la période d'utilisation élevée sur le domaine `ldom1`.

La stratégie `high-usage` suivant effectue les opérations suivantes :

- Indique que les heures de début et de fin sont 9 heures et 18 heures en définissant les propriétés `tod-begin` et `tod-end` respectivement.
- Indique que les limites inférieure et supérieure auxquelles effectuer une analyse de stratégie sont 25 % et 75 % en définissant les propriétés `util-lower` et `util-upper` respectivement.
- Indique que les nombres minimal et maximal de CPU virtuelles sont 2 et 16 en définissant les propriétés `vcpu-min` et `vcpu-max` respectivement.
- Indique que le nombre maximal de CPU virtuelles à ajouter au cours d'un cycle de contrôle des ressources est 1 en définissant la propriété `attack`.
- Indique que le nombre maximal de CPU virtuelles à supprimer au cours d'un cycle de contrôle des ressources est 1 en définissant la propriété `decay`.
- Indique que la priorité de cette stratégie est 1 en définissant la propriété `priority`. Une priorité de 1 signifie que cette stratégie sera appliquée même si une autre stratégie est en vigueur.
- Indique que le nom du fichier de stratégie est `high-usage` en définissant la propriété `name`.
- Utilise les valeurs par défaut pour les propriétés qui ne sont pas définies, notamment `enable` et `sample-rate`. Reportez-vous à la page de manuel `ldm`(1M).

```
primary# ldm add-policy tod-begin=09:00 tod-end=18:00 util-lower=25 util-upper=75 \
vcpu-min=2 vcpu-max=16 attack=1 decay=1 priority=1 name=high-usage ldom1
```

La commande `ldm add-policy` suivante crée la stratégie `med-usage` à utiliser au cours de la période d'utilisation faible sur le domaine `ldom1`.

La stratégie `med-usage` suivant effectue les opérations suivantes :

- Indique que les heures de début et de fin sont 18 heures et 9 heures en définissant les propriétés `tod-begin` et `tod-end` respectivement.

- Indique que les limites inférieure et supérieure auxquelles effectuer une analyse de stratégie sont 10 % et 50 % en définissant les propriétés `util-lower` et `util-upper` respectivement.
- Indique que les nombres minimal et maximal de CPU virtuelles sont 2 et 16 en définissant les propriétés `vcpu-min` et `vcpu-max` respectivement.
- Indique que le nombre maximal de CPU virtuelles à ajouter au cours d'un cycle de contrôle des ressources est 1 en définissant la propriété `attack`.
- Indique que le nombre maximal de CPU virtuelles à supprimer au cours d'un cycle de contrôle des ressources est 1 en définissant la propriété `decay`.
- Indique que la priorité de cette stratégie est 1 en définissant la propriété `priority`. Une priorité de 1 signifie que cette stratégie sera appliquée même si une autre stratégie est en vigueur.
- Indique que le nom du fichier de stratégie est `high-usage` en définissant la propriété `name`.
- Utilise les valeurs par défaut pour les propriétés qui ne sont pas définies, notamment `enable` et `sample-rate`. Reportez-vous à la page de manuel `ldm`(1M).

```
primary# ldm add-policy tod-begin=18:00 tod-end=09:00 util-lower=10 util-upper=50 \
 vcpu-min=2 vcpu-max=16 attack=1 decay=1 priority=1 name=med-usage ldom1
```

Liste des ressources du domaine

Cette section montre l'utilisation de la syntaxe des sous-commandes `ldm`, définit certains termes de sortie, tels que les balises et les statistiques d'utilisation et fournit des exemples qui sont semblables à ce qui s'affiche en sortie.

Sortie lisible par la machine

Si vous créez des scripts utilisant la sortie de la commande `ldm list`, utilisez toujours l'option `-p` pour produire un formulaire de la sortie lisible par la machine.

Pour visualiser l'utilisation de la syntaxe pour toutes les sous-commandes `ldm`, utilisez la commande suivante :

```
primary# ldm --help
```

Pour plus d'informations sur les sous-commandes `ldm`, reportez-vous à la page de manuel `ldm`(1M).

Définitions des balises

Les balises suivantes peuvent être affichées dans la sortie pour un domaine (`ldm list`). Si vous utilisez les options longues analysables (`-l -p`) pour la commande, les balises sont écrites en entier, par exemple, `flags=normal,control,vio-service`. Sinon, vous voyez des abréviations en lettres, par exemple `-n-cv-`. Les valeurs des balises de la liste dépendent de leur position. Les valeurs suivantes peuvent apparaître dans chacune des six colonnes de gauche à droite.

Colonne 1 – Démarrage ou arrêt des domaines

- `s` – Démarrage ou arrêt

Colonne 2 – Statut du domaine

- `n` – Normal
- `t` – Transition
- `d` – Domaine dégradé qui ne peut pas être démarré pour cause de ressources manquantes

Colonne 3 – Statut de la reconfiguration

- `d` – Reconfiguration retardée
- `r` – Reconfiguration dynamique de la mémoire

Colonne 4 – Domaine de contrôle

- `c` – Domaine de contrôle

Colonne 5 – Domaine de service

- `v` – Domaine de service d'E/S virtuelle

Colonne 6 – Statut de la migration

- `s` – Domaine source dans une migration
- `t` – Domaine cible dans une migration
- `e` – Erreur survenue pendant une migration

Définition des statistiques d'utilisation

Les statistiques d'utilisation par CPU virtuelle (`UTIL`) sont affichées par le biais de l'option longue (`-l`) de la commande `ldm list`. Les statiques sont le pourcentage de temps que la CPU virtuelle a passé sur l'exécution pour le compte d'un système d'exploitation invité. Une CPU virtuelle est considérée comme un exécutant pour le compte du système d'exploitation invité sauf lorsqu'il a été cédé à l'hyperviseur. Si le système d'exploitation invité ne cède pas de

CPU virtuelles à l'hyperviseur, l'utilisation des CPU dans le système d'exploitation invité sera toujours affichée comme étant à `100%`.

Les statistiques d'utilisation indiquées pour un domaine logique sont la moyenne des utilisations des CPU virtuelles dans le domaine. Les statistiques d'utilisation normalisée (`NORM`) correspondent au pourcentage de temps d'exécution des CPU virtuelles pour le compte des SE hôtes. Cette valeur prend en compte des opérations telles que les sauts de cycles. La virtualisation normalisée est uniquement disponible si votre système exécute la dernière version 8.2.0 du microprogramme système.

Si PM n'effectue pas d'opération de saut, l'utilisation 100 % correspond à l'utilisation normalisée 100 %. Lorsque PM ajuste le saut du cycle sur 4/8, l'utilisation 100 % correspond à 50 %, ce qui signifie que la CPU possède uniquement la moitié du nombre possible de cycles disponibles. En conséquence, les CPU entièrement utilisées ont un pourcentage d'utilisation normalisée de 50 %. Utilisez la commande `ldm list` ou `ldm list -l` pour afficher l'utilisation normalisée relative aux CPU virtuelles et aux SE hôtes.

Affichage des différentes listes

- Pour afficher les versions logicielles actuellement installées :

  ```
  primary# ldm -V
  ```

- Pour générer une liste abrégée de tous les domaines :

  ```
  primary# ldm list
  ```

- Pour générer une liste longue de tous les domaines :

  ```
  primary# ldm list -l
  ```

- Pour générer une liste étendue de tous les domaines :

  ```
  primary# ldm list -e
  ```

- Pour générer une liste analysable lisible par la machine de tous les domaines :

  ```
  primary# ldm list -p
  ```

- Vous pouvez générer une sortie en tant que sous-ensemble de ressources en entrant une ou plusieurs des options *format* suivantes. Si vous indiquez plusieurs formats, délimitez les éléments par des virgules sans espaces.

  ```
  primary# ldm list -o resource[,resource...] domain-name
  ```

  - `console` – La sortie contient la console virtuelle (`vcons`) et le service de concentrateur de console virtuelle (`vcc`)
  - `core` – La sortie contient des informations sur les domaines ayant des coeurs complets alloués

- `cpu` – La sortie contient des informations sur la CPU virtuelle (`vcpu`), la CPU physique (`pcpu`) et l'ID du coeur
- `crypto` – La sortie de l'unité cryptographique contient une unité arithmétique modulaire (`mau`) et toute autre unité cryptographique prise en charge, notamment la Control Word Queue (CWQ)
- `disk` – La sortie contient le disque virtuel (`vdisk`) et le serveur de disque virtuel (`vds`)
- La sortie `domain-name` – contient les variables (`var`), host ID (`hostid`), l'état du domaine, des balises, l'UUID et l'état du logiciel
- `memory` – La sortie contient `memory`
- `network` – La sortie contient l'adresse de contrôle d'accès au média (`mac`), le commutateur de réseau virtuel (`vsw`) et le périphérique réseau virtuel (`vnet`)
- `physio` – L'entrée/sortie physique contient l'interconnexion de composants périphériques (`pci`) et l'unité d'interface réseau (`niu`)
- `resmgmt` – La sortie contient des informations sur la stratégie DRM (gestion dynamique des ressources), indique quelle stratégie est actuellement utilisée et répertorie les contraintes associées à la configuration whole-core
- `serial` – La sortie contient le service de canal de domaine logique (`vldc`)et le client du canal de domaine logique virtuel (`vldcc`))
- `stats` – La sortie contient des statistiques relatives aux stratégies de gestion des ressources
- `status` – La sortie contient l'état sur la migration du domaine en cours

Les exemples suivants montrent différents sous-ensembles de sortie que vous pouvez définir :

- Pour dresser la liste des informations des CPU pour le domaine de contrôle :

```
primary# ldm list -o cpu primary
```

- Pour dresser la liste des informations de domaine pour un domaine invité :

```
primary# ldm list -o domain ldm2
```

- Pour dresser la liste des informations sur la mémoire et le réseau pour un domaine invité :

```
primary# ldm list -o network,memory ldm1
```

- Pour dresser la liste des informations sur la stratégie DRM pour un domaine invité :

```
primary# ldm list -o resmgmt,stats ldm1
```

- Pour afficher une variable et sa valeur pour un domaine :

```
primary# ldm list-variable variable-name domain-name
```

Par exemple, la commande suivante affiche la valeur de la variable `boot-device` sur le domaine `ldg1` :

```
primary# ldm list-variable boot-device ldg1
boot-device=/virtual-devices@100/channel-devices@200/disk@0:a
```

- Pour dresser la liste des ressources liées à un domaine :

```
primary# ldm list-bindings domain-name
```

- Pour dresser la liste des configurations de domaine logique stockées sur le SP :

  La commande `ldm list-config` répertorie les configurations de domaine logique qui sont stockées sur le processeur de service. Si l'option `-r` est utilisée, cette commande répertorie les configurations pour lesquelles des fichiers enregistrés automatiquement existent sur le domaine de contrôle.

  Pour plus d'informations sur les configurations, reportez-vous à la section "Gestion des configurations de domaine" à la page 389. Pour plus d'exemples, reportez-vous à la page de manuel `ldm`(1M).

```
primary# ldm list-config
factory-default
3guests
foo [next poweron]
primary
reconfig-primary
```

  Les libellés à la droite du nom de la configuration ont la signification suivante :

  - `[current]` – Dernière configuration démarrée, uniquement tant qu'elle correspond à la configuration en cours d'exécution, c'est-à-dire jusqu'à ce que vous démarreriez une reconfiguration. Après la reconfiguration, l'annotation devient `[next poweron]`.
  - `[next poweron]` – Configuration à utiliser lors du cycle d'alimentation suivant.
  - `[degraded]` – Version endommagée de la configuration précédemment initialisée.

- Pour répertorier toutes les ressources du serveur, liées ou non :

```
primary# ldm list-devices -a
```

- Pour répertorier la quantité de mémoire disponible à allouer :

```
primary# ldm list-devices mem
MEMORY
    PA                  SIZE
    0x14e000000         2848M
```

- Pour déterminer quelles portions de mémoire sont indisponibles pour des domaines logiques :

```
primary# ldm list-devices -a mem
MEMORY
    PA                  SIZE            BOUND
    0x0                 57M             _sys_
```

```
0x3900000           32M             _sys_
0x5900000           94M             _sys_
0xb700000           393M            _sys_
0x24000000          192M            _sys_
0x30000000          255G            primary
0x3ff0000000        64M             _sys_
0x3ff4000000        64M             _sys_
0x3ff8000000        128M            _sys_
0x800000000000      2G              ldg1
0x800800000000      2G              ldg2
0x801000000000      2G              ldg3
0x801800000000      2G              ldg4
0x802000000000      103G
0x81bc0000000       145G            primary
```

- Pour répertorier les services disponibles :

```
primary# ldm list-services
```

Liste des contraintes

Les contraintes de Logical Domains Manager sont une ou plusieurs ressources que vous voulez assigner à un domaine particulier. Soit vous recevez toutes les ressources dont vous avez demandé l'ajout au domaine, soit vous ne recevez aucune d'entre elles, en fonction des ressources disponibles. La sous-commande `list-constraints` répertorie les ressources que vous voulez assigner au domaine.

- Pour répertorier les contraintes pour un domaine :

```
# ldm list-constraints domain-name
```

- Pour répertorier les contraintes au format XML pour un domaine particulier :

```
# ldm list-constraints -x domain-name
```

- Pour répertorier les contraintes pour tous les domaines dans un format analysable :

```
# ldm list-constraints -p
```

Liste des informations sur le groupe de ressources

Affichez les informations sur le groupe de ressources à l'aide de la commande `ldm list-rsrc-group`.

La commande suivante affiche des informations sur tous les groupes de ressources :

```
primary# ldm list-rsrc-group
NAME                        CORE MEMORY IO
/SYS/CMU4                   12   256G   4
/SYS/CMU5                   12   256G   4
/SYS/CMU6                   12   128G   4
/SYS/CMU7                   12   128G   4
```

Comme les autres commandes `ldm list-*`, vous pouvez définir les options permettant d'afficher la sortie analysable, la sortie détaillée, ainsi que des informations sur des groupes de ressources et des domaines en particuliers. Pour plus d'informations, reportez-vous à la page de manuel `ldm`(1M).

L'exemple suivant utilise l'option `-l` pour afficher des informations détaillées sur le groupe de ressources `/SYS/CMU5`.

```
primary# ldm list-rsrc-group -l /SYS/CMU5
NAME                                    CORE    MEMORY    IO
/SYS/CMU5                               12      256G      4

CORE
    CID                                                 BOUND
    192, 194, 196, 198, 200, 202, 208, 210          primary
    212, 214, 216, 218                              primary

MEMORY

    PA              SIZE              BOUND
    0xc0000000000   228M              ldg1
    0xc0030000000   127G              primary
    0xc1ffc000000   64M               _sys_
    0xd0000000000   130816M           primary
    0xd1ffc000000   64M               _sys_

IO
    DEVICE          PSEUDONYM         BOUND
    pci@900         pci_24            primary
    pci@940         pci_25            primary
    pci@980         pci_26            primary
    pci@9c0         pci_27            primary
```

Utilisation des propriétés Perf-Counter

La fonctionnalité de contrôle d'accès au registre des performances permet d'obtenir, de définir et de rétablir les droits d'accès à un domaine de certains groupes de registres de performances.

Indiquez une valeur pour la propriété `perf-counters` à l'aide des commandes `ldm add-domain` et `ldm set-domain`. La nouvelle valeur de propriété `perf-counters` sera reconnue par le domaine invité à la prochaine réinitialisation. Si aucune valeur `perf-counters` est indiquée, la valeur est `htstrand`. Reportez-vous à la page de manuel `ldm`(1M).

Vous pouvez spécifier les valeurs suivantes pour la propriété `perf-counters` :

`global` Permet aux compteurs de performances globaux d'accéder au domaine accessible par ses ressources allouées. Seul un domaine à la fois peut accéder aux compteurs de performances globaux. Vous pouvez définir cette valeur seule ou avec les valeurs `strand` ou `htstrand`.

`strand` Permet aux compteurs de performances strand existant sur les CPU allouées au domaine d'accéder au domaine. Vous ne pouvez pas indiquer cette valeur en même temps que la valeur `htstrand`.

`htstrand` Se comporte de la même manière que la valeur `strand` et permet l'instrumentation des événements du mode hyperprivilège sur la CPU allouée au domaine. Vous ne pouvez pas indiquer cette valeur en même temps que la valeur `strand`.

Pour désactiver tous les accès des compteurs de performances, indiquez `perf-counters=`.

Si l'hyperviseur ne permet pas l'accès aux performances, il est impossible de définir la propriété sur `perf-counters`.

Les commandes `ldm list -o domain` et `ldm list -e` affichent la valeur de la propriété `perf-counters`. Si l'accès aux performances n'est pas pris en charge, la valeur `perf-counters` ne s'affiche pas dans la sortie.

EXEMPLE 70 Création d'un domaine et indication de l'accès à son registre des performances

Créez le nouveau domaine `ldg0` avec accès au jeu de registres `global` :

```
primary# ldm add-domain perf-counters=global ldg0
```

EXEMPLE 71 Indication de l'accès au registre des performances d'un domaine

Indiquez que le domaine `ldg0` a accès aux jeux de registres `global` et `strand` :

```
primary# ldm set-domain perf-counters=global,strand ldg0
```

EXEMPLE 72 Indication qu'un domaine n'a accès à aucun jeu de registres

Indiquez que le domaine `ldg0` n'a accès à aucun jeu de registres :

```
primary# ldm set-domain perf-counters= ldg0
```

EXEMPLE 73 Affichage des informations d'accès aux performances

Les exemples suivants illustrent comment afficher les informations d'accès aux performances à l'aide de la commande `ldm list -o domain`.

- La commande `ldm list -o domain` montre que les valeurs de performances `global` et `htstrand` sont définies dans le domaine `ldg0` :

```
primary# ldm list -o domain ldg0
NAME     STATE     FLAGS     UTIL
NORM
ldg0     active    -n----    0.0% 0.0%

SOFTSTATE
Solaris running

UUID
    062200af-2de2-e05f-b271-f6200fd3eee3

HOSTID
    0x84fb315d

CONTROL
    failure-policy=ignore
    extended-mapin-space=on
    cpu-arch=native
    rc-add-policy=
    shutdown-group=15
    perf-counters=global,htstrand

DEPENDENCY
    master=

PPRIORITY    4000

VARIABLES
    auto-boot?=false
    boot-device=/virtual-devices@100/channel-devices@200/disk@0:a
      /virtualdevices@100/channel@200/disk@0
    network-boot-arguments=dhcp,hostname=solaris,
      file=http://10.129.241.238:5555/cgibin/wanboot-cgi
    pm_boot_policy=disabled=0;ttfc=2000;ttmr=0;
```

- La commande `ldm list -p -o domain` présente les mêmes informations que dans l'exemple précédent, mais sous une forme analysable :

```
primary# ldm list -p -o domain ldg0
VERSION 1.12
DOMAIN|name=ldg0|state=active|flags=normal|util=|norm_util=
UUID|uuid=4e8749b9-281b-e2b1-d0e2-ef4dc2ce5ce6
HOSTID|hostid=0x84f97452
CONTROL|failure-policy=reset|extended-mapin-space=on|cpu-arch=native|rc-add-policy=|
shutdown-group=15|perf-counters=global,htstrand
DEPENDENCY|master=
VARIABLES
|auto-boot?=false
|boot-device=/virtual-devices@100/channel-devices@200/disk@0
|pm_boot_policy=disabled=0;ttfc=2500000;ttmr=0;
```

Problèmes liés à la gestion des ressources

La suppression d'un grand nombre de CPU d'un domaine invité peut échouer

Le message d'erreur suivant peut s'afficher lorsque vous tentez de supprimer un grand nombre de CPU d'un domaine invité :

```
Request to remove cpu(s) sent, but no valid response received
VCPU(s) will remain allocated to the domain, but might
not be available to the guest OS
Resource modification failed
```

Pour éviter ce problème, supprimez moins de 100 CPU à la fois du domaine.

Parfois, le bloc de mémoire ajouté de manière dynamique ne peut être supprimé dynamiquement que dans sa totalité.

En raison de la manière dont le SE Oracle Solaris traite les métadonnées pour la gestion de la mémoire ajoutée de manière dynamique, vous pourrez peut-être uniquement supprimer le bloc de mémoire entier qui a été précédemment ajouté de manière dynamique plutôt qu'un sous-ensemble correct de cette mémoire.

Cette situation se présente si un domaine dont la taille de mémoire est petite est dynamiquement augmenté, comme indiqué dans l'exemple suivant.

```
primary# ldm list ldom1
NAME  STATE FLAGS   CONS VCPU MEMORY UTIL UPTIME
ldom1 active -n--   5000 2    2G     0.4% 23h

primary# ldm add-mem 16G ldom1

primary# ldm rm-mem 8G ldom1
Memory removal failed because all of the memory is in use.

primary# ldm rm-mem 16G ldom1

primary# ldm list ldom1
NAME  STATE FLAGS   CONS VCPU MEMORY UTIL UPTIME
ldom1 active -n--   5000 2    2G     0.4% 23h
```

Pour contourner ce problème, utilisez la commande `ldm add-mem` pour ajouter de la mémoire de manière séquentielle par blocs de petite taille plutôt que par blocs de plus grande taille que vous pourriez souhaiter supprimer à l'avenir.

Si vous avez rencontré ce problème, effectuez l'une des actions suivantes :

- Arrêtez le domaine, supprimez la mémoire, puis redémarrez le domaine.
- Réinitialisez le domaine, ce qui provoque la réallocation des métadonnées de gestion de la mémoire du SE Oracle Solaris, de telle sorte que la mémoire ajoutée précédemment peut maintenant être supprimée de manière dynamique par blocs de plus petite taille.

♦ ♦ ♦ CHAPITRE 16

Gestion des configurations de domaine

Ce chapitre contient des informations sur la gestion des configurations de domaine.

Ce chapitre aborde les sujets suivants :

Gestion des configurations de domaine

Une *configuration* de domaine est une description complète de tous les domaines et de toutes leurs allocations de ressource au sein d'un seul système. Vous pouvez enregistrer et stocker les configurations sur le processeur de service (SP) pour une utilisation ultérieure.

Si vous enregistrez une configuration sur le processeur de service (SP), celle-ci sera conservée à chaque nouveau cycle d'alimentation. Vous pouvez enregistrer plusieurs configurations et spécifier la configuration à initialiser lors de la prochaine tentative de mise sous tension.

Lorsque vous mettez un système sous tension, le SP démarre la configuration sélectionnée. Le système exécute le même ensemble de domaines et utilise les mêmes allocations de ressource de virtualisation et de partitionnement que celles indiquées dans la configuration. La configuration par défaut est celle qui a été enregistrée le plus récemment. Vous pouvez également demander explicitement une autre configuration à l'aide de la commande `ldm set-spconfig` ou de la commande ILOM appropriée.

Attention - Enregistrez toujours votre configuration stable sur le SP et au format XML. L'enregistrement de la configuration de l'une de ces manières vous permet de récupérer la configuration de votre système après une coupure de courant et permet une utilisation ultérieure. Voir la section "Enregistrement des configurations de domaine" à la page 393.

Une copie locale de la configuration du SP et de la base de données des contraintes de Logical Domains est enregistrée sur le domaine de contrôle à chaque fois que vous enregistrez une

configuration sur le SP. Cette copie locale est nommée *bootset*. Le fichier bootset permet de charger la base de données des contraintes de Logical Domains correspondante lorsque le système est arrêté et redémarré.

Sur les serveurs SPARC T5, SPARC T7, SPARC M5, SPARC M6, SPARC M7, SPARC S7 et le Serveur Fujitsu M10, les bootsets du domaine de contrôle sont les copies principales des configurations. Au démarrage, Logical Domains Manager synchronise automatiquement toutes les configurations avec le SP. Cette opération garantit que les configurations figurant sur le SP sont toujours identiques aux bootsets stockés sur le domaine de contrôle.

Remarque - Les bootsets contenant des données système critiques, assurez-vous que le système de fichiers du domaine de contrôle utilise la mise en miroir de disques ou la technologie RAID afin de réduire l'impact des pannes de disque.

Un domaine physique est l'étendue des ressources gérées par une seule instance Oracle VM Server for SPARC. Un domaine physique peut être un système physique complet, comme c'est le cas avec les serveurs SPARC des séries T et S7 pris en charge. Ou il peut également s'agir de l'ensemble du système ou d'un sous-ensemble du système, comme c'est le cas des serveurs SPARC de série M pris en charge.

Méthodes de récupération de configuration disponibles

Oracle VM Server for SPARC prend en charge les méthodes de récupération de configuration suivantes :

- La méthode d'enregistrement automatique, utilisée lorsque la configuration n'est pas disponible sur le SP.

  Cette situation peut se produire dans l'un des cas suivants :

  - Le matériel qui abrite les configurations enregistrées a été remplacé
  - La configuration n'est pas à jour car vous avez oublié d'enregistrer les dernières modifications apportées à la configuration sur le SP ou parce qu'un cycle d'alimentation inattendu s'est produit
- La méthode `ldm add-domain`, utilisée si les configurations d'un sous-ensemble des domaines doivent être restaurées
- La méthode `ldm init-system`, qui ne doit être utilisée qu'en dernier recours. Utilisez cette méthode uniquement lorsque la configuration sur le SP et les informations d'enregistrement du domaine de contrôle sont perdues.

Restauration des configurations à l'aide de l'enregistrement automatique

Une copie de la configuration actuelle est automatiquement enregistrée sur le domaine de contrôle chaque fois que la configuration de domaine est modifiée. Cette opération d'enregistrement automatique ne sauvegarde pas explicitement la configuration sur le SP.

L'opération d'enregistrement automatique est effectuée immédiatement, même dans les situations suivantes :

- Lorsque la nouvelle configuration n'est pas explicitement enregistrée sur le SP
- Lorsque la modification de la configuration n'est pas effectuée tant que le domaine concerné n'a pas été réinitialisé

L'opération d'enregistrement automatique vous permet de récupérer une configuration lorsque les configurations enregistrées sur le SP sont perdues. Cette opération vous permet également de récupérer une configuration lorsque la configuration actuelle n'a pas été explicitement enregistrée sur le SP après un cycle d'alimentation du système. Dans ce cas, Logical Domains Manager peut restaurer cette configuration au redémarrage si elle est plus récente que la configuration sélectionnée pour l'initialisation suivante.

Remarque - Les événements de gestion de l'alimentation, FMA et ASR ne provoquent pas une mise à jour des fichiers enregistrés automatiquement.

Vous pouvez restaurer automatiquement ou manuellement les fichiers enregistrés automatiquement sur des configurations nouvelles ou existantes. Par défaut, lorsqu'une configuration enregistrée automatiquement est plus récente que la configuration correspondante en cours d'exécution, un message est écrit dans le journal de Logical Domains. Vous devez donc utiliser la commande `ldm add-spconfig -r` pour mettre à jour manuellement une configuration existante ou en créer une nouvelle en fonction des données enregistrées automatiquement. Notez que vous devez effectuer un cycle d'alimentation après avoir utilisé cette commande, afin d'effectuer une récupération manuelle.

Remarque - Lorsque qu'une reconfiguration retardée est en attente, les modifications de configuration sont immédiatement enregistrées automatiquement. Par conséquent, si vous exécutez la commande `ldm list-config -r`, la configuration enregistrée automatiquement apparaît comme plus récente que la configuration actuelle.

Pour plus d'informations sur l'utilisation des commandes `ldm *-spconfig` pour gérer les configurations et récupérer manuellement les fichiers enregistrés automatiquement, reportez-vous à la page de manuel `ldm`(1M).

Pour plus d'informations sur la procédure de sélection d'une configuration d'initialisation, reportez-vous à la section "Utilisation d'Oracle VM Server for SPARC avec le processeur de service" à la page 412. Vous pouvez également utiliser la commande `ldm set-spconfig`, décrite à page de manuel `ldm`(1M).

Stratégie de récupération automatique

La stratégie de récupération automatique définit comment traiter la récupération d'une configuration lorsqu'une configuration automatiquement enregistrée sur le domaine de contrôle est plus récente que la configuration en cours d'exécution correspondante. La stratégie de récupération automatique est spécifiée en définissant la propriété `autorecovery_policy` du service SMF `ldmd`. Cette propriété peut prendre les valeurs suivantes :

- `autorecovery_policy=1` – Consigne tous les messages d'avertissement lorsqu'une configuration enregistrée automatiquement est plus récente que la configuration en cours d'exécution correspondante. Ces messages sont consignés dans le fichier journal SMF `ldmd`. Vous devez effectuer manuellement toute récupération de configuration. Il s'agit de la stratégie par défaut.
- `autorecovery_policy=2` – Affiche un message de notification si une configuration enregistrée automatiquement est plus récente que la configuration en cours d'exécution correspondante. Ce message de notification est imprimé dans la sortie des commandes `ldm` la première fois qu'une commande `ldm` est émise après chaque redémarrage de Logical Domains Manager. Vous devez effectuer manuellement toute récupération de configuration.
- `autorecovery_policy=3` – Met automatiquement à jour la configuration si une configuration enregistrée automatiquement est plus récente que la configuration en cours d'exécution correspondante. Cette action remplace la configuration du SP qui sera utilisée lors du prochain cycle d'alimentation. Pour rendre cette configuration disponible, vous devez effectuer un autre cycle d'alimentation. Cette configuration est mise à jour avec la nouvelle configuration qui est enregistrée sur le domaine de contrôle. Cette action n'a aucune incidence sur la configuration actuellement en cours d'exécution. Elle a uniquement une incidence sur la configuration qui sera utilisée lors du cycle d'alimentation suivant. Un message est également consigné. Celui-ci indique qu'une configuration plus récente a été enregistrée sur le SP et qu'elle sera initialisée lors du cycle d'alimentation suivant du système. Ces messages sont consignés dans le fichier journal SMF `ldmd`.

▼ Procédure de modification de la stratégie de récupération automatique

1. **Connectez-vous au domaine de contrôle.**
2. **Connectez-vous en tant qu'administrateur.**

Pour Oracle Solaris 11.3, reportez-vous au Chapitre 1, "About Using Rights to Control Users and Processes" du manuel *Securing Users and Processes in Oracle Solaris 11.3*.

3. **Consultez la valeur de la propriété `autorecovery_policy`.**

```
# svccfg -s ldmd listprop ldmd/autorecovery_policy
```

4. **Arrêtez le service `ldmd`.**

```
# svcadm disable ldmd
```

5. **Modifiez la valeur de la propriété `autorecovery_policy`.**

```
# svccfg -s ldmd setprop ldmd/autorecovery_policy=value
```

Par exemple, pour faire que la stratégie effectue une récupération automatique, définissez la valeur de la propriété sur `3` :

```
# svccfg -s ldmd setprop ldmd/autorecovery_policy=3
```

6. **Actualisez et redémarrez le service `ldmd`.**

```
# svcadm refresh ldmd
# svcadm enable ldmd
```

Exemple 74 Modification de la stratégie de récupération automatique à partir du journal de récupération automatique

L'exemple suivant décrit comment consulter la valeur actuelle de la propriété `autorecovery_policy` et la remplacer par une nouvelle valeur. La valeur d'origine de cette propriété est `1`, ce qui signifie que les modifications enregistrées automatiquement sont consignées. La commande `svcadm` permet d'arrêter et de redémarrer le service `ldmd`, et la commande `svccfg` permet de consulter et de définir la valeur de la propriété.

```
# svccfg -s ldmd listprop ldmd/autorecovery_policy
ldmd/autorecovery_policy integer 1
# svcadm disable ldmd
# svccfg -s ldmd setprop ldmd/autorecovery_policy=3
# svcadm refresh ldmd
# svcadm enable ldmd
```

Enregistrement des configurations de domaine

Vous pouvez enregistrer une configuration de domaine pour un domaine unique ou pour tous les domaines sur un système.

A l'exception des ressources physiques nommées, la méthode suivante ne permet pas de conserver les liaisons. Toutefois, elle conserve les contraintes utilisées pour les créer. Après l'enregistrement et la restauration de la configuration, les domaines ont les mêmes ressources

virtuelles, mais ils ne sont pas nécessairement liés aux mêmes ressources physiques. Les ressources physiques nommées sont liées comme indiqué par l'administrateur.

- Pour enregistrer la configuration pour un seul domaine, créez un fichier XML contenant les contraintes du domaine.

  ```
  # ldm list-constraints -x domain-name >domain-name.xml
  ```

  L'exemple suivant indique comment créer un fichier XML, `ldg1.xml`, qui contient les contraintes du domaine `ldg1` :

  ```
  # ldm list-constraints -x ldg1 >ldg1.xml
  ```

- Pour enregistrer les configurations pour tous les domaines sur un système, créez un fichier XML contenant les contraintes de tous les domaines.

  ```
  # ldm list-constraints -x >file.xml
  ```

  L'exemple suivant indique comment créer un fichier XML, `config.xml`, contenant les contraintes de tous les domaines sur un système :

  ```
  # ldm list-constraints -x >config.xml
  ```

Restauration des configurations de domaine

Cette section décrit comment restaurer une configuration de domaine à partir d'un fichier XML pour des domaines invités et pour le domaine de contrôle (`primary`).

- Pour restaurer une configuration de domaine pour des domaines invités, utilisez la commande `ldm add-domain -i`, comme décrit à la section "Procédure de restauration d'une configuration de domaine à partir d'un fichier XML (ldm add-domain)" à la page 394. Bien qu'il soit possible d'enregistrer les contraintes du domaine `primary` dans un fichier XML, vous ne pouvez pas utiliser le fichier comme entrée pour cette commande.
- Pour restaurer une configuration de domaine pour le domaine `primary`, utilisez la commande `ldm init-system` et les contraintes de ressource du fichier XML pour reconfigurer votre domaine `primary`. Vous pouvez également utiliser la commande `ldm init-system` pour reconfigurer d'autres domaines décrits dans le fichier XML, mais ceux-ci risquent de rester inactifs une fois la configuration terminée. Voir la section "Procédure de restauration d'une configuration de domaine à partir d'un fichier XML (ldm init-system)" à la page 395.

▼ Procédure de restauration d'une configuration de domaine à partir d'un fichier XML (`ldm add-domain`)

Cette procédure est valable pour les domaines hôtes mais pas pour le domaine de contrôle (`primary`). Si vous souhaitez restaurer la configuration pour le domaine `primary` ou pour

d'autres domaines XML décrits dans le fichier XML, reportez-vous à la section "Procédure de restauration d'une configuration de domaine à partir d'un fichier XML (ldm init-system)" à la page 395.

1. **Créez le domaine à l'aide du fichier XML créé en entrée.**

```
# ldm add-domain -i domain-name.xml
```

2. **Associez le domaine.**

```
# ldm bind-domain [-fq] domain-name
```

L'option -f force la liaison du domaine même si des périphériques backend non valides sont détectés. L'option -q désactive la validation des périphériques backend afin d'accélérer l'exécution de la commande.

3. **Démarrez le domaine.**

```
# ldm start-domain domain-name
```

Exemple 75 Restauration d'un seul domaine à partir d'un fichier XML

L'exemple suivant décrit comment restaurer un domaine unique. En premier lieu, vous restaurez le domaine ldg1 à partir du fichier XML. Ensuite, vous associez et démarrez le domaine ldg1 que vous avez restauré.

```
# ldm add-domain -i ldg1.xml
# ldm bind ldg1
# ldm start ldg1
```

▼ Procédure de restauration d'une configuration de domaine à partir d'un fichier XML (ldm init-system)

Cette procédure explique comment utiliser la commande ldm init-system avec un fichier XML pour recréer une configuration enregistrée précédemment.

Attention - La commande ldm init-system peut ne pas restaurer correctement une configuration dans laquelle des commandes d'E/S physiques ont été utilisées. Parmi ces commandes, on trouve ldm add-io, ldm set-io, ldm remove-io, ldm create-vf et ldm destroy-vf. Pour plus d'informations, reportez-vous à la section "La commande ldm init-system peut ne pas correctement restaurer une configuration de domaine sur lesquels des modifications d'E/S physiques ont été apportées" du manuel *Notes de version d'Oracle VM Server for SPARC 3.4*.

Avant de commencer

Vous devez avoir créé un fichier de configuration XML en exécutant la commande ldm list-constraints -x. Le fichier doit décrire une ou plusieurs configurations de domaine.

1. **Connectez-vous au domaine `primary`.**

2. **Vérifiez que le système est en configuration `factory-default`.**

   ```
   primary# ldm list-config | grep "factory-default"
   factory-default [current]
   ```

   Si le système n'est pas en configuration `factory-default`, reportez-vous à la section "Procédure de restauration de la configuration usine par défaut" à la page 435.

3. **Connectez-vous en tant qu'administrateur.**

   Pour Oracle Solaris 11.3, reportez-vous au Chapitre 1, "About Using Rights to Control Users and Processes" du manuel *Securing Users and Processes in Oracle Solaris 11.3*.

4. **Restaurez la ou les configurations de domaine à partir d'un fichier XML.**

   ```
   # ldm init-system [-frs] -i filename.xml
   ```

   Le domaine `primary` doit être réinitialisé pour que la configuration prenne effet. L'option `-r` réinitialise le domaine `primary` après la configuration. Si vous ne spécifiez pas l'option `-r`, vous devez effectuer la réinitialisation manuellement.

   L'option `-s` restaure uniquement la configuration des services virtuels(`vds`, `vcc` et `vsw`) et peut être effectuée sans réinitialisation.

   L'option `-f` ignore la vérification de la configuration des valeurs d'usine et ne prend pas en compte les paramètres système déjà configurés. Utilisez l'option `-f` avec précaution. La commande `ldm init-system` suppose que le système est configuré avec les valeurs d'usine et applique par conséquent directement les modifications spécifiées par le fichier XML. L'utilisation de l'option `-f` lorsque la configuration du système n'est pas celle d'origine risque de générer un système dont la configuration ne respecte pas les spécifications énoncées dans le fichier XML. Il est probable qu'une ou plusieurs modifications ne soient pas appliquées au système, en fonction de la combinaison des modifications contenues dans le fichier XML et de la configuration initiale.

   Le domaine `primary` est reconfiguré comme indiqué dans le fichier. Tous les domaines autres que `primary` ayant des configurations dans le fichier XML sont reconfigurés mais restent inactifs.

Exemple 76 Restauration de domaines à partir de fichiers de configuration XML

Les exemples suivants décrivent comment utiliser la commande `ldm init-system` pour restaurer le domaine `primary` et tous les domaines sur un système à partir de la configuration `factory-default`.

- **Restauration du domaine `primary`.** L'option `-r` permet de réinitialiser le domaine `primary` à l'issue de la configuration. Le fichier `primary.xml` contient la configuration de domaine XML que vous avez enregistrée précédemment.

```
primary# ldm init-system -r -i primary.xml
```

- **Restauration de tous les domaines sur un système.** Restaurez les configurations contenues dans le fichier XML `config.xml` pour tous les domaines du système. Le fichier `config.xml` contient les configurations de domaine XML que vous avez enregistrées précédemment. Le domaine `primary` est redémarré automatiquement par la commande `ldm init-system`. Tous les autres domaines sont restaurés, mais ne sont pas liés, ni redémarrés.

  ```
  # ldm init-system -r -i config.xml
  ```

  Après la réinitialisation du système, les commandes suivantes associent et redémarrent les domaines `ldg1` et `ldg2` :

  ```
  # ldm bind ldg1
  # ldm start ldg1
  # ldm bind ldg2
  # ldm start ldg2
  ```

Problèmes de connexion du processeur de service d'adressage

Sur les serveurs SPARC des séries T7, M7 et S7, l'interconnexion ILOM est utilisée pour communiquer entre le service `ldmd` et le processeur de service (SP).

Si une tentative d'utiliser la commande `ldm` pour la gestion des configurations de domaine échoue en raison d'une erreur lors de la communication avec le SP, suivez les étapes de récupération ci-après s'appliquant à votre plate-forme :

- **Serveurs SPARC T7, SPARC M7 et SPARC S7 :** vérifiez l'état de l'interconnexion ILOM et réactivez le service `ilomconfig-interconnect`. Reportez-vous à "Procédure de vérification de la configuration d'interconnexion ILOM" à la page 45 et "Procédure de réactivation du service d'interconnexion ILOM" à la page 46.
- **Serveurs SPARC T3, SPARC T4, SPARC T5, SPARC M5 et SPARC M6 :** redémarrez le service `ldmd`.

  ```
  primary# svcadm enable ldmd
  ```

Si ces étapes ne parviennent pas à rétablir les communications, redémarrez le processeur de service (SP).

Problèmes liés à la gestion de la configuration

`init-system` ne restaure pas les contraintes de coeur nommées pour les domaines invités à partir des fichiers XML enregistrés

Si vous avez affecté des ressources de coeur nommées à un domaine, il est possible que la commande `ldm init-system` ne parvienne pas à réaffecter ces ressources nommées à ce domaine. Cela peut se produire parce que la commande `ldm init-system` lance une reconfiguration retardée du domaine `primary` et que vous ne pouvez effectuer qu'une seule opération de CPU virtuelle par reconfiguration retardée. En conséquence, cette commande ne parvient pas à restaurer les contraintes de coeur des CPU nommés pour les domaines invités à partir d'un fichier XML enregistré.

Solution de contournement : effectuez les opérations suivantes :

1. Créez un fichier XML pour le domaine primary.

   ```
   primary# ldm list-constraints -x primary > primary.xml
   ```

2. Créez un fichier XML pour le ou les domaines invités.

   ```
   primary# ldm list-constraints -x domain-name[,domain-name][,...] > guest.xml
   ```

3. Effectuez un cycle d'alimentation du système et initialisez une configuration usine par défaut.
4. Appliquez la configuration XML au domaine `primary`.

   ```
   primary# ldm init-system -r -i primary.xml
   ```

5. Appliquez la configuration XML aux domaines invités.

   ```
   primary# ldm init-system -f -i guest.xml
   ```

♦ ♦ ♦ CHAPITRE 17

Gestion des erreurs matérielles

Ce chapitre contient des informations sur la gestion des erreurs matérielles par Oracle VM Server for SPARC.

Ce chapitre aborde les sujets suivants :

Présentation de la gestion des erreurs matérielles

Le logiciel Oracle VM Server for SPARC ajoute les capacités RAS suivantes pour les plates-formes SPARC de niveau professionnel, en commençant par les séries de serveurs SPARC T5, SPARC M5 et SPARC S7 :

- **Liste noire de l'architecture de gestion des pannes (FMA, Fault Management Architecture).** Lorsque l'architecture FMA détecte des ressources CPU ou de mémoire défectueuses, Oracle VM Server for SPARC les place sur une liste noire. Une ressource défectueuse se trouvant sur la liste noire ne peut pas être réassignée à un domaine tant que l'architecture FMA ne l'a pas marquée comme réparée.
- **Mode de récupération.** Permet la récupération automatique des configurations de domaine qui ne peuvent pas être initialisées à cause de ressources défectueuses ou manquantes.

La Plate-forme Fujitsu M10 prend également en charge le mode de récupération. Bien que la liste noire de ressources défectueuses ne soit pas prise en charge, la fonction de remplacement automatique de la Plate-forme Fujitsu M10 propose une fonctionnalité semblable.

Utilisation de l'architecture FMA pour mettre sur liste noire les ressources défectueuses ou annuler leur configuration

L'architecture FMA contacte Logical Domains Manager lorsqu'elle détecte une ressource défectueuse. Ensuite, Logical Domains Manager tente de mettre fin à l'utilisation de cette ressource dans tous les domaines en cours d'exécution. Pour s'assurer qu'une ressource défectueuse ne pourra pas être assignée à un domaine, l'architecture FMA ajoute cette ressource à une liste noire.

Logical Domains Manager prend uniquement en charge la mise sur liste noire pour les ressources de CPU et de mémoire, et non pour les ressources d'E/S.

Si une ressource défectueuse n'est pas utilisée, Logical Domains Manager la supprime de la liste des ressources disponibles, que vous pouvez consulter dans la sortie de `ldm list-devices`. Cette ressource est alors marquée en interne comme étant “sur liste noire” de façon à ce qu'elle ne puisse pas être réassignée à un domaine ultérieurement.

Si la ressource défectueuse est en cours d'utilisation, Logical Domains Manager tente d'évacuer la ressource. Pour éviter une interruption de service sur les domaines en cours d'exécution, Logical Domains Manager tente d'abord d'utiliser la reconfiguration dynamique de la CPU ou de la mémoire pour évacuer la ressource défectueuse. Logical Domains Manager mappe à nouveau un coeur défectueux si celui-ci est disponible pour être utilisé en tant que cible. Si cette “évacuation en direct” réussit, la ressource défectueuse est marquée en interne comme placée sur liste noire et ne s'affiche pas dans la sortie `ldm list-devices`, de manière à ce qu'elle ne puisse pas être assignée à un domaine ultérieurement.

Si l'évacuation en direct échoue, Logical Domains Manager affecte en interne le statut « évacuation en attente » à la ressource défectueuse. La ressource s'affiche normalement dans la sortie `ldm list-devices` car elle est encore utilisée sur les domaines en cours d'exécution jusqu'à ce que les domaines invités affectés soient réinitialisés ou arrêtés.

Lorsque le domaine invité affecté est arrêté ou réinitialisé, Logical Domains Manager tente d'évacuer les ressources défectueuses et de les marquer en interne comme placées sur liste noire afin qu'elles ne puissent pas être assignées ultérieurement. Un tel périphérique n'est pas affiché dans la sortie `ldm`. Une fois l'exécution en attente effectuée, Logical Domains Manager tente de démarrer le domaine invité. Toutefois, si le domaine invité ne peut pas être démarré car les ressources disponibles sont insuffisantes, le domaine invité est marqué comme “dégradé” et le message d'avertissement suivant est émis, afin d'inviter l'utilisateur à effectuer une récupération manuelle.

```
primary# ldm ls
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  NORM  UPTIME
primary          active     -n-cv-  UART    368   2079488M 0.1%  0.0%  16h 57m
gd0              bound      -d----  5000    8
```

```
warning: Could not restart domain gd0 after completing pending evacuation.
The domain has been marked degraded and should be examined to see
if manual recovery is possible.
```

Lorsque le système est arrêté et redémarré, l'architecture FMA répète les demandes d'évacuation des ressources qui sont encore défectueuses et Logical Domains Manager gère ces demandes en évacuant les ressources défectueuses et en les marquant en interne comme placées sur liste noire.

Avant l'introduction de la mise sur liste noire par l'architecture FMA, une panique d'un domaine invité due à une ressource défectueuse pouvait entraîner une boucle de réinitialisation infinie. En utilisant l'évacuation de ressource et la mise sur liste noire lorsque le domaine invité est réinitialisé, vous pouvez éviter cette boucle et empêcher les tentatives ultérieures d'utilisation d'une ressource défectueuse.

Récupération de domaines après la détection de ressources défectueuses ou manquantes

Si un serveur SPARC T5, un serveur de série SPARC T7, un serveur SPARC M5, un serveur SPARC M6 , un serveur de série SPARC M7, un serveur de série SPARC S7 ou un Serveur Fujitsu M10 détecte une ressource défectueuse ou manquante lors de la mise sous tension, le Logical Domains Manager essaie de récupérer les domaines configurés en utilisant les ressources qui restent disponibles. Lors de la récupération, le système (ou le domaine physique sur les serveurs de série SPARC M5, SPARC M6 et SPARC M7) est considéré comme étant en *mode de récupération*. Le mode de récupération est activé par défaut. Reportez-vous à "Contrôle du mode de récupération" à la page 405.

A la mise sous tension, le microprogramme du système revient à la configuration usine par défaut si la dernière configuration de mise sous tension sélectionnée ne peut pas être initialisée dans l'une des situations suivantes :

- La topologie d'E/S au sein de chaque commutateur PCIe dans la configuration ne correspond pas à la topologie d'E/S de la dernière configuration de mise sous tension sélectionnée.
- Les ressources de CPU ou de mémoire de la dernière configuration de mise sous tension sélectionnée ne sont plus présentes dans le système.

Quand le mode de récupération est activé, Logical Domains Manager récupère tous les domaines actifs et liés à partir de la dernière configuration de mise sous tension sélectionnée. La configuration en cours d'exécution qui en résulte est appelée la *configuration dégradée*. La configuration dégradée est enregistrée sur le SP et reste la configuration active jusqu'à ce

qu'une nouvelle configuration soit enregistrée ou que le domaine physique soit soumis à un cycle d'alimentation.

Remarque - Le domaine physique n'a pas besoin d'un cycle d'alimentation pour activer la configuration dégradée après la récupération, puisque la configuration dégradée est déjà en cours d'exécution.

Si le domaine physique est soumis à un cycle d'alimentation, le microprogramme du système tente d'abord d'initialiser la dernière configuration de mise sous tension d'origine. De cette manière, si le ou les disques manquants ou défectueux ont été remplacés dans l'intervalle, le système peut initialiser la configuration normale d'origine. Si la dernière configuration de mise sous tension sélectionnée ne peut pas être initialisée, le microprogramme tente d'initialiser la configuration dégradée associée, si elle existe. Si la configuration dégradée ne peut pas être initialisée ou n'existe pas, la configuration d'usine par défaut est initialisée et le mode de récupération est appelé.

L'opération de récupération se déroule dans l'ordre suivant

- **Domaine de contrôle.** Logical Domains Manager récupère le domaine de contrôle en restaurant sa CPU, sa mémoire et la configuration d'E/S ainsi que ses services d'E/S virtuels.

  Si la quantité de CPU ou de mémoire requise pour tous les domaines récupérables est supérieure aux ressources restantes, le nombre de CPU, de coeurs ou de mémoires est réduit proportionnellement à la taille des autres domaines. Par exemple, dans un système à quatre domaines sur lequel 25 % des CPU et de la mémoire est assigné à chaque domaine, la configuration dégradée qui en découle continue d'assigner 25 % des CPU et de la mémoire à chaque domaine. Si le domaine `primary` possédait à l'origine jusqu'à deux coeurs (16 CPU virtuelles) et huit Go de mémoire, la taille du domaine de contrôle n'est pas réduite.

  Les complexes root et les périphériques PCIe assignés à d'autres domaines sont supprimés du domaine de contrôle. Les fonctions virtuelles sur des complexes root détenus par le domaine de contrôle sont recréées. Tout complexe root, périphérique PCIe, fonction physique ou fonction virtuelle manquant assigné au domaine de contrôle est marqué comme évacué. Logical Domains Manager réinitialise ensuite le domaine de contrôle pour appliquer les modifications.
- **Domaines root.** Après la réinitialisation du domaine de contrôle, Logical Domains Manager récupère les domaines root. La quantité de CPU et de mémoire est réduite proportionnellement aux autres domaines récupérables, le cas échéant. Si un complexe root n'est plus présent physiquement sur le système, il est marqué comme évacué. Ce complexe root n'est pas configuré dans le domaine pendant l'opération de récupération. Un domaine root est récupéré tant qu'au moins un des complexes root assignés au domaine root est disponible. Si aucun de ses complexes root n'est disponible, le domaine root n'est pas récupéré. Logical Domains Manager initialise le domaine root et recrée les fonctions virtuelles sur les fonctions physiques détenues par le domaine root. Il retire également les emplacements PCIe prêtés par le domaine root. Tous les emplacements PCIe, fonctions

physiques et fonctions virtuelles manquants sont marqués comme évacués. Dans la mesure du possible, tous les services d'E/S virtuels fournis par le domaine sont recréés.

- **Domaines d'E/S.** Logical Domains Manager récupère tout domaine d'E/S. Tous les emplacements PCIe et les fonctions virtuelles manquants sur le système sont marqués comme évacués. Si aucun des périphériques d'E/S requis n'est présent, le domaine n'est pas récupéré et ses ressources CPU et de mémoire sont disponibles pour une utilisation par d'autres domaines. Dans la mesure du possible, tous les services d'E/S virtuels fournis par le domaine sont recréés.
- **Domaines invités.** Un domaine invité est *uniquement* récupéré si au moins un des domaines de service qui le servent a été récupéré. Si le domaine invité ne peut pas être récupéré, ses ressources de CPU et de mémoire sont disponibles pour une utilisation par d'autres domaines invités.

Dans la mesure du possible, le système alloue à un domaine le même nombre de CPU et la même quantité de mémoire que dans la configuration d'origine. Si le nombre de CPU ou la quantité de mémoire disponibles sont insuffisants, ces ressources sont réduites proportionnellement pour consommer les ressources disponibles restantes. Si vous avez affecté des ressources nommées à un domaine et qu'il est plus tard récupéré en mode de récupération, aucune tentative de réaffectation de ces ressources nommées à ce domaine n'est effectuée.

Remarque - Lorsqu'un système est en mode récupération, vous pouvez uniquement exécuter les commandes `ldm list-*`. Toutes les autres commandes `ldm` sont désactivées jusqu'à la fin de l'opération de récupération.

Logical Domains Manager tente uniquement de récupérer les domaines liés et actifs. La configuration des ressources existante de tout domaine non lié est copiée en l'état dans la nouvelle configuration.

Au cours d'une opération de récupération, il est possible que la quantité de ressources disponibles soit inférieure à la précédente configuration initialisée. De ce fait, il est possible que Logical Domains Manager ne puisse récupérer que certains des domaines précédemment configurés. D'autre part, un domaine récupéré peut ne pas disposer de toutes les ressources dont il disposait dans la configuration d'origine. Par exemple, un domaine lié récupéré peut présenter moins de ressources d'E/S qu'il n'en avait dans la configuration précédente. Un domaine risque de ne pas être récupéré si ses périphériques d'E/S sont absents ou si son domaine de service parent n'a pas pu être récupéré.

Le mode de récupération consigne les étapes de sa réalisation dans le journal SMF Logical Domains Manager, `/var/svc/log/ldoms-ldmd:default.log`. Un message est écrit sur la console système lorsque Logical Domains Manager démarre une récupération, réinitialise le domaine de contrôle et à l'issue de la récupération.

Attention - Il n'est pas garanti qu'un domaine récupéré soit parfaitement opérationnel. Le domaine risque de ne pas inclure une ressource essentielle à l'exécution de l'instance du système d'exploitation ou d'une application. Par exemple, un domaine récupéré peut ne disposer que d'une ressource réseau et d'aucune ressource de disque. Ou un système de fichiers nécessaire à l'exécution d'une application peut manquer dans un domaine récupéré. L'utilisation d'E/S à chemins d'accès multiples pour un domaine limite les conséquences de l'absence de certaines ressources d'E/S.

Configuration matérielle et logicielle requise pour le mode de récupération

- **Configuration matérielle —-** La fonctionnalité de mode de récupération est prise en charge sur les serveurs SPARC T5, SPARC T7, SPARC M5, SPARC M6, SPARC M7 et SPARC S7, ainsi que sur le Serveur Fujitsu M10.
- **Configuration du microprogramme –** Au minimum la version 9.1.0.a du microprogramme système pour les serveurs SPARC T5, SPARC M5 et SPARC M6. Au minimum la version 9.4.3 du microprogramme système pour les serveurs de série SPARC T7 et SPARC M7. Toute version publiée du microprogramme système pour les serveurs de série SPARC S7. Au minimum la version XCP2230 du microprogramme système pour le Serveur Fujitsu M10.
- **Configuration logicielle –** Les domaines root non-`primary` qui prêtent des emplacements PCIe doivent exécuter au moins le SE Oracle Solaris 10 1/13 ou le SE Oracle Solaris 11.2.

Configuration dégradée

Chaque domaine physique ne peut disposer que d'une seule configuration dégradée enregistrée sur le SP. S'il existe déjà une configuration dégradée, elle est remplacée par la dernière configuration dégradée créée.

Vous ne pouvez pas interagir directement avec les configurations dégradées. Si nécessaire, le microprogramme du système initialise de manière transparente la version dégradée de la prochaine configuration de mise sous tension. Cette transparence permet au système d'initialiser la configuration d'origine après un cycle d'alimentation lorsque les ressources manquantes réapparaissent. Lorsque la configuration active est une configuration dégradée, elle est marquée indiquée comme `[degraded]` dans la sortie `ldm list-spconfig`.

La fonctionnalité d'enregistrement automatique est désactivée lorsque la configuration active est une configuration dégradée. Si vous enregistrez une nouvelle configuration sur le SP alors

qu'une configuration dégradée est active, la nouvelle configuration est une configuration non dégradée normale.

Remarque - Une ressource précédemment manquante qui réapparaît après un cycle d'alimentation ultérieur n'a pas d'effet sur le contenu d'une configuration normale. Toutefois, si vous sélectionnez par la suite la configuration qui a déclenché le mode de récupération, le SP initialise la configuration non dégradée d'origine, puisque l'ensemble de son matériel est à présent disponible.

Contrôle du mode de récupération

La propriété SMF `ldmd/recovery_mode` contrôle le comportement du mode de récupération. Le mode de récupération est activé par défaut.

Si la propriété `ldmd/recovery_mode` n'est pas présente ou est définie sur `auto`, le mode de récupération est activé.

Lorsque la propriété `ldmd/recovery_mode` est définie sur `never`, Logical Domains Manager sort du mode de récupération sans effectuer d'action et le domaine physique exécute la configuration d'usine par défaut.

Remarque - Si le microprogramme du système demande le mode de récupération lorsque celui-ci n'est pas activé, exécutez les commandes suivantes pour activer le mode de récupération à la demande :

```
primary# svccfg -s ldmd setprop ldmd/recovery_mode = astring: auto
primary# svcadm refresh ldmd
primary# svcadm restart ldmd
```

Le mode de récupération est uniquement initié immédiatement dans ce scénario si vous n'avez apporté aucune modification au système, c'est-à-dire s'il dispose toujours de la configuration d'usine par défaut.

En outre, pour activer le mode de récupération, vous pouvez spécifier une valeur de délai d'attente pour l'initialisation d'un domaine root lors de la récupération. Par défaut, la valeur de la propriété `ldmd/recovery_mode_boot_timeout` est de 30 minutes. Les valeurs valides commencent à 5 minutes.

Marquage de domaines comme dégradés

Un domaine est marqué comme dégradé si la mise sur liste noire d'une ressource par l'architecture FMA ne laisse pas assez de ressources au domaine pour démarrer. Le domaine reste dans l'état lié, ce qui empêche les ressources restantes qui lui sont assignées d'être réassignées à d'autres domaines.

Marquage de ressources d'E/S comme évacuées

Une ressource d'E/S détectée comme manquante par le mode de récupération est marquée comme évacuée à l'aide d'un astérisque (`*`) dans la sortie de liste `ldm`.

♦ ♦ ♦ CHAPITRE 18

Réalisation d'autres tâches d'administration

Ce chapitre contient des informations sur l'utilisation du logiciel Oracle VM Server for SPARC pour des tâches qui ne sont pas décrites dans les chapitres précédents.

Ce chapitre aborde les sujets suivants :

Entrée de noms dans la CLI

En général, les noms du Logical Domains Manager peuvent comporter jusqu'à 256 caractères.

Les sections suivantes décrivent les restrictions de dénomination dans la CLI du Logical Domains Manager.

- Noms de variable
  - Le premier caractère doit être une lettre, un nombre ou une barre oblique (/).
  - Les lettres suivantes doivent être des lettres, des nombres ou des signes de ponctuation.
- Noms de fichier qui sont utilisés dans des périphériques backend de disque virtuel, noms de commutateur virtuel et noms de fichier de chemin

  Les noms ne doivent contenir que des lettres, des nombres ou des signes de ponctuation.
- Noms de configuration

  La longueur des noms de configuration de domaine (ou noms de configuration de processeur de service) est limitée par le processeur de service. La limite se situe actuellement autour de 69 caractères mais peut varier selon la plate-forme.

  Si vous spécifiez un nom de configuration de processeur de service trop long, le message d'erreur suivant peut s'afficher :

  ```
  primary# ldm add-spconfig \
  test5678901234567892123456789312345678941234567895123456789612345678 97
  Error: Operation failed because an invalid configuration name was given
  ```
- Noms de service et de client de périphérique virtuel

  Les noms de périphérique virtuel sont utilisés pour créer la propriété `devalias` utilisée par OpenBoot PROM. Toutefois, le microprogramme OpenBoot PROM ne prend pas en charge les noms `devalias` dépassant 31 caractères.

  Si vous indiquez un nom de périphérique virtuel qui dépasse 31 caractères, la commande aboutit mais la propriété `devalias` n'est pas créée. La commande émet également l'avertissement suivant :

  ```
  primary# ldm add-vds primary-vds012345678901234567890 primary
  Warning: Device name primary-vds012345678901234567890 is too long to create devalias
  ```
- Noms de chemin de matériel

  Ces noms représentent les chemins d'accès aux ressources physiques. Ces noms servent à indiquer le port dans la commande `ldm add-vsan` et le groupe de ressources dans la commande `ldm list-rsrc-group`.
- Tous les autres noms
  - Le premier caractère doit être une lettre ou un nombre.
  - Les caractères suivants doivent être des lettres, des chiffres ou l'un des caractères suivants : `-_+#.:;~()`.

Mise à jour des valeurs de propriété dans le fichier /etc/system

Ne procédez à aucune modification manuelle dans le fichier /etc/system. Ce fichier est automatiquement généré lors de la réinitialisation au moment où les fichiers du répertoire/etc/system.d indiquent les valeurs de propriété de réglage.

▼ Procédure d'ajout et de modification d'une valeur de propriété de réglage

1. **Recherchez le nom de la propriété de réglage dans le fichier /etc/system et dans les fichiers /etc/system.d.**

   Par exemple, si vous souhaitez indiquer une valeur pour la propriété vds:vd_volume_force_slice, recherchez d'abord si la propriété est déjà définie.

   ```
   # grep 'vds:vd_volume_force_slice' /etc/system /etc/system.d/*
   ```

2. **Mettez à jour la valeur de la propriété :**

   - **Si vous trouvez la propriété dans l'un des fichiers/etc/system.d, mettez à jour sa valeur dans le fichier.**
   - **Si vous trouvez la propriété dans le fichier /etc/system ou si elle n'y figure pas, créez un fichier dans le répertoire /etc/system.d et attribuez-lui un nom tel que dans l'exemple suivant :**

     /etc/system.d/com.*company-name*:ldoms-config

Délai d'arrêt d'un domaine fortement chargé pouvant être dépassé

Le délai d'une commande ldm stop-domain peut être dépassé avant que le domaine ne termine l'arrêt. Lorsque cela se produit, une erreur similaire à la suivante est renvoyée par Logical Domains Manager.

```
LDom ldg8 stop notification failed
```

Cependant, le domaine peut toujours traiter la demande d'arrêt. Utilisez la commande `ldm list-domain` pour vérifier l'état du domaine. Par exemple :

```
# ldm list-domain ldg8
NAME          STATE   FLAGS  CONS   VCPU MEMORY  UTIL UPTIME
ldg8          active  s----  5000   22   3328M   0.3% 1d 14h 31m
```

La liste précédente montre le domaine comme actif, mais l'indicateur `s` indique que le domaine est en cours d'arrêt. Il doit s'agir d'un état transitoire.

L'exemple suivant montre que le domaine s'est maintenant arrêté.

```
# ldm list-domain ldg8
NAME          STATE   FLAGS  CONS   VCPU MEMORY  UTIL UPTIME
ldg8          bound   -----  5000   22   3328M
```

La commande `ldm stop` utilise la commande `shutdown` pour arrêter un domaine. L'exécution de la séquence d'arrêt prend en général plus de temps qu'un arrêt rapide. Vous pouvez effectuer un arrêt rapide en exécutant la commande `ldm stop -q`. Reportez-vous à la page de manuel `ldm`(1M).

L'exécution d'une longue séquence d'arrêt peut entraîner l'affichage du message suivant :

domain-name `stop timed out. The domain might still be in the process of shutting down. Either let it continue, or specify -f to force it to stop.`

Lors de l'exécution de cette séquence d'arrêt, l'indicateur `s` s'affiche également pour le domaine.

Utilisation du SE Oracle Solaris avec Oracle VM Server for SPARC

Cette section décrit les modifications de comportement du SE Oracle Solaris survenant une fois qu'une configuration créée par Logical Domains Manager a été instanciée.

Microprogramme OpenBoot indisponible une fois que le SE Oracle Solaris a démarré

Le microprogramme OpenBoot n'est pas disponible après le démarrage du SE Oracle Solaris, car il a été déchargé de la mémoire.

Pour accéder à l'invite `ok` à partir du SE Oracle Solaris, vous devez arrêter le domaine à l'aide de la commande `halt` du SE Oracle Solaris.

Arrêt et redémarrage d'un serveur

Lorsque vous effectuez une opération de maintenance nécessitant d'effectuer un cycle d'alimentation du serveur sur un système exécutant le logiciel Oracle VM Server for SPARC, vous devez d'abord enregistrer les configurations actuelles des domaines logiques sur le SP.

Pour enregistrer vos configurations de domaine actuelles sur le SP, utilisez la commande suivante :

```
# ldm add-config config-name
```

Résultat des interruptions du SE Oracle Solaris

Vous pouvez amorcer des interruptions du SE Oracle Solaris de la manière suivante :

1. Appuyer sur la séquence de touches L1-A lorsque le périphérique d'entrée est défini sur `keyboard`.
2. Entrer la commande `send break` lorsque la console est à l'invite `telnet`.

Lorsque vous lancez ce type d'interruption, le SE Oracle Solaris renvoie l'invite suivante :

```
c)ontinue, s)ync, r)eset, h)alt?
```

Tapez la lettre représentant ce que vous voulez que le système fasse après ces types d'interruptions.

Résultats de la réinitialisation du domaine de contrôle

Vous pouvez utiliser les commandes `reboot` et `shutdown -i 5` pour réinitialiser le domaine de contrôle (`primary`).

- `reboot` :
  - **Aucun autre domaine configuré.** Réinitialise le domaine de contrôle sans arrêt progressif et sans mise hors tension.
  - **Autres domaines configurés.** Réinitialise le domaine de contrôle sans arrêt progressif et sans mise hors tension.

- `shutdown -i 5` :
  - **Aucun autre domaine configuré.** L'hôte est hors tension après l'arrêt progressif et le reste tant qu'il n'est pas remis sous tension au niveau du SP.
  - **Autres domaines configurés.** Se réinitialise avec un arrêt progressif et sans mise hors tension.

Il arrive qu'un système Oracle VM Server for SPARC ne revienne pas à l'invite `ok` après une réinitialisation consécutive à l'exécution de la commande `uadmin 1 0` à partir de la ligne de commande. Ce comportement anormal se présente uniquement si la variable `auto-reboot?` est définie sur `true`. Si `auto-reboot?` est définie sur `false`, la commande génère le résultat attendu. Pour contourner ce problème, utilisez la commande `uadmin 2 0` ou exécutez toujours la variable `auto-reboot?` définie sur `false`.

Pour plus d'informations sur les conséquences de la réinitialisation d'un domaine ayant le rôle de domaine root, reportez-vous à la section "Réinitialisation du domaine root avec des extrémités PCIe configurées" à la page 162.

Utilisation d'Oracle VM Server for SPARC avec le processeur de service

Cette section décrit les informations sur l'utilisation du processeur de service (SP) Integrated Lights Out Manager (ILOM) avec Logical Domains Manager. Pour plus d'informations sur l'utilisation du logiciel ILOM, reportez-vous aux documents relatifs à votre plate-forme disponibles à l'adresse `http://www.oracle.com/technetwork/documentation/sparc-tseries-servers-252697.html`.

Une option `config` supplémentaire est disponible dans la commande ILOM existante ::

```
-> set /HOST/bootmode config=config-name
```

Cette option vous permet de régler la configuration sur une autre configuration à la remise sous tension suivante, y compris la configuration d'expédition `factory-default`.

Vous pouvez appeler la commande que l'hôte soit sous tension ou non. Elle est appliquée au prochain redémarrage ou à la prochaine réinitialisation de l'hôte.

Pour réinitialiser la configuration du domaine logique, définissez l'option sur `factory-default`.

```
-> set /HOST/bootmode config=factory-default
```

Vous pouvez également sélectionner d'autres configurations créées avec Logical Domains Manager à l'aide de la commande `ldm add-config` et stockées sur le processeur de service (SP). Le nom que vous indiquez dans la commande `ldm add-config` de Logical Domains Manager peut être utilisé pour sélectionner cette configuration avec la commande `bootmode` d'ILOM. Par exemple, supposons que vous stockiez la configuration portant le nom `ldm-config1`.

```
-> set /HOST/bootmode config=ldm-config1
```

Ensuite, vous devez effectuer un cycle d'alimentation du système pour charger la nouvelle configuration.

Reportez-vous à la page de manuel `ldm`(1M) pour plus d'informations sur la commande `ldm add-config`.

Configuration des dépendances de domaine

Vous pouvez utiliser Logical Domains Manager pour établir des relations de dépendance entre les domaines. Un domaine ayant un ou plusieurs domaines dépendant de lui-même est appelé un *domaine maître*. Un domaine dépendant d'un autre domaine est appelé un *domaine esclave*.

Chaque domaine esclave peut définit jusqu'à quatre domaines maîtres en définissant la propriété `master`. Par exemple, le domaine esclave `pine` définit quatre domaines maîtres dans la liste séparée par des virgules suivante :

```
# ldm add-domain master=alpha,beta,gamma,delta pine
```

Les domaines maîtres `alpha`, `beta`, `gamma` et `delta` indiquent tous une stratégie de panne `stop`.

Chaque domaine maître peut définir ce qui se produit sur ses domaines esclaves en cas de panne du domaine maître. Par exemple, si un domaine maître est en panne, il peut demander à ses domaines esclaves de paniquer. Si un domaine esclave a plusieurs domaines maîtres, chaque domaine maître doit avoir la même stratégie de panne. Ainsi, le premier domaine maître à être en panne déclenche la stratégie de panne qu'il a définie sur tous ses domaines esclaves.

La stratégie de panne du domaine maître est contrôlée en définissant l'une des valeurs suivantes dans la propriété `failure-policy` :

- `ignore` ignore tous les domaines esclaves
- `panic` panique tous les domaines esclaves (à l'instar de l'exécution de la commande `ldm panic`)
- `reset` arrête immédiatement puis redémarre tous les domaines esclaves (à l'instar de l'exécution de la commande `ldm stop -f`, puis de la commande `ldm start`)
- `stop` arrête tous les domaines esclaves (à l'instar de l'exécution de la commande `ldm stop -f`)

Dans cet exemple, les domaines maîtres définissent leur stratégie de panne comme suit :

```
primary# ldm set-domain failure-policy=ignore apple
primary# ldm set-domain failure-policy=panic lemon
primary# ldm set-domain failure-policy=reset orange
primary# ldm set-domain failure-policy=stop peach
primary# ldm set-domain failure-policy=stop alpha
primary# ldm set-domain failure-policy=stop beta
primary# ldm set-domain failure-policy=stop gamma
primary# ldm set-domain failure-policy=stop delta
```

Vous pouvez utiliser ce mécanisme pour créer des dépendances explicites entre les domaines. Par exemple, un domaine invité dépend implicitement du domaine de service devant lui fournir ses périphériques virtuels. L'E/S d'un domaine invité est bloquée lorsque le domaine de service duquel il dépend n'est pas en état de fonctionnement. En définissant un domaine invité en tant qu'esclave de son domaine de service, vous pouvez définir le comportement du domaine invité lorsque son domaine de service est en panne. Lorsque cette dépendance n'est pas établie, un domaine invité attend simplement la reprise du service de son domaine de service.

Remarque - Logical Domains Manager ne vous autorise pas à créer des relations de domaine qui créent un cycle de dépendance. Pour plus d'informations, voir "Cycles de dépendance" à la page 416.

Pour des exemples XML de dépendances de domaines, reportez-vous à la section "Ressource d’informations sur le domaine (ldom_info)" du manuel *Guide du développeur d’Oracle VM Server for SPARC 3.4*.

Exemples de dépendances de domaine

Les exemples suivants montrent comment configurer les dépendances de domaine.

EXEMPLE 77 Configuration d'une stratégie de panne à l'aide de dépendances de domaine

La première commande crée un domaine maître appelé `twizzle`. Cette commande utilise `failure-policy=reset` pour indiquer que les domaines esclaves se réinitialisent si le domaine `twizzle` est en panne. La seconde commande modifie un domaine maître appelé `primary`. Cette commande utilise `failure-policy=reset` pour indiquer que les domaines esclaves se réinitialisent si le domaine `primary` est en panne. La troisième commande crée un domaine esclave appelé `chocktaw` qui dépend de deux domaines maîtres, `twizzle` et `primary`. Le domaine esclave utilise `master=twizzle,primary` pour définir ses domaines maîtres. Si le domaine `twizzle` ou `primary` est en panne, le domaine`chocktaw` se réinitialise.

```
primary# ldm add-domain failure-policy=reset twizzle
primary# ldm set-domain failure-policy=reset primary
primary# ldm add-domain master=twizzle,primary chocktaw
```

EXEMPLE 78 Modification d'un domaine pour assigner un domaine maître

Cet exemple montre comment utiliser la commande `ldm set-domain` pour modifier le domaine `orange` afin d'assigner le domaine `primary` en tant que domaine maître. La seconde commande utilise la commande `ldm set-domain` pour assigner `orange` et `primary` en tant que domaines maîtres pour le domaine `tangerine`. La troisième commande répertorie les informations sur tous ces domaines.

```
primary# ldm set-domain master=primary orange
primary# ldm set-domain master=orange,primary tangerine
primary# ldm list -o domain
NAME             STATE      FLAGS   UTIL
primary          active     -n-cv-  0.2%

SOFTSTATE
Solaris running

HOSTID
    0x83d8b31c

CONTROL
    failure-policy=ignore

DEPENDENCY
    master=

------------------------------------------------------------------------------
NAME             STATE      FLAGS   UTIL
orange           bound      ------

HOSTID
    0x84fb28ef

CONTROL
    failure-policy=ignore

DEPENDENCY
    master=primary

------------------------------------------------------------------------------
NAME             STATE      FLAGS   UTIL
tangerine        bound      ------

HOSTID
    0x84f948e9

CONTROL
    failure-policy=ignore

DEPENDENCY
    master=orange,primary
```

EXEMPLE 79 Affichage d'une liste de domaines analysable

L'exemple suivant montre la liste d'exemples avec une sortie analysable :

```
primary# ldm list -o domain -p
```

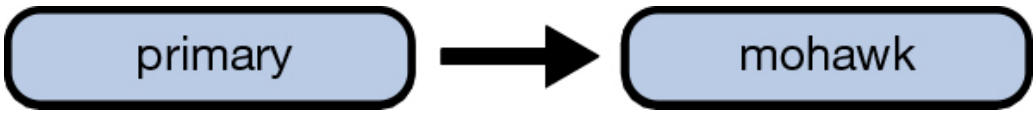
Cycles de dépendance

Logical Domains Manager ne vous autorise pas à créer des relations de domaine qui créent un cycle de dépendance. Un *cycle de dépendance* est une relation entre deux domaines ou plus qui aboutit à une situation dans laquelle un domaine esclave dépend de lui-même ou un domaine maître dépend d'un de ses domaines esclaves.

Logical Domains Manager détermine si un cycle de dépendance existe avant d'ajouter une dépendance. Logical Domains Manager démarre sur le domaine esclave et effectue une recherche sur tous les chemins indiqués par la matrice maître jusqu'à ce que le chemin soit atteint. Tous les cycles de dépendance trouvés sur le chemin sont signalés comme des erreurs.

L'exemple suivant montre comment un cycle de dépendance peut être créé. La première commande crée un domaine esclave nommé `mohawk` qui définit le domaine maître en tant que `primary`. Par conséquent, `mohawk` dépend de `primary` dans la chaîne de dépendance illustrée dans la figure suivante.

FIGURE 26 Dépendance d'un domaine unique

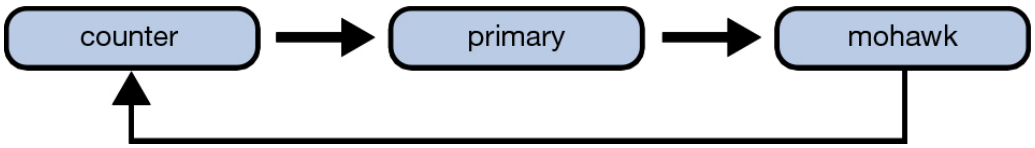


La seconde commande crée un domaine esclave nommé `primary` qui définit son domaine maître en tant que `counter`. Par conséquent, `mohawk` dépend de `primary`, qui dépend de `counter` dans la chaîne de dépendance illustrée dans le schéma suivant :

FIGURE 27 Dépendance de plusieurs domaines

La troisième commande tente de créer une dépendance entre les domaines `counter` et `mohawk`, ce qui produirait le cycle de dépendance illustré dans le schéma suivant :

FIGURE 28 Cycle de dépendance de domaines



La commande `ldm set-domain` échouera avec le message d'erreur suivant :

```
# ldm add-domain master=primary mohawk
# ldm set-domain master=counter primary
# ldm set-domain master=mohawk counter
Dependency cycle detected: LDom "counter" indicates "primary" as its master
```

Détermination de l'endroit où les erreurs sont survenues lors du mappage du CPU et des adresses de mémoire

Cette section décrit comment vous pouvez corréler les informations transmises par l'architecture de gestion des pannes (FMA) d'Oracle Solaris avec les ressources des domaines logiques identifiées comme étant en panne.

Le FMA signale les erreurs CPU en termes de nombres de CPU physiques et les erreurs de mémoire en termes d'adresses de mémoire physique.

Si vous voulez déterminer dans quel domaine logique une erreur est survenue et le numéro de CPU virtuelle correspondant ou l'adresse de mémoire réelle dans le domaine, vous devez effectuez un mappage.

Mappage de CPU

Il permet de trouver le domaine ainsi que le numéro de CPU virtuelle dans le domaine correspondant à un numéro de CPU physique donné.

Générez tout d'abord une longue liste analysable de tous les domaines à l'aide de la commande suivante :

```
primary# ldm list -l -p
```

Recherchez l'entrée dans les sections VCPU de la liste ayant une zone `pid` équivalente au numéro de CPU physique.

- Si vous trouvez une entrée de ce type, la CPU est dans le domaine sous lequel l'entrée est répertoriée, et le numéro de CPU virtuelle dans le domaine est indiqué par la zone `vid` de l'entrée.
- Si vous ne trouvez pas d'entrée de ce type, la CPU ne se trouve dans aucun domaine.

Mappage de la mémoire

Il permet de trouver le domaine ainsi que l'adresse de mémoire réelle dans le domaine correspondant à une adresse de mémoire physique (PA) donnée.

Générez tout d'abord une longue liste analysable de tous les domaines.

```
primary# ldm list -l -p
```

Recherchez la ligne dans les sections MEMORY de la liste où la PA se trouve dans la plage de *pa* à (*pa* + *size* - 1) ; c'est-à-dire *pa* ≤ PA ≤ (*pa* + *size* - 1). *pa* et *size* représentent les valeurs dans les champs correspondants de la ligne.

- Si vous trouvez une entrée de ce type, le PA est dans le domaine sous lequel l'entrée est répertoriée, et l'adresse réelle correspondante dans le domaine est indiquée par la zone *ra* + (PA - *pa*).
- Si vous ne trouvez pas d'entrée de ce type, le PA ne se trouve dans aucun domaine.

Exemple de mappage de CPU et de mémoire

EXEMPLE 80 Détermination de la configuration des domaines

La commande suivante produit une longue liste analysable de configurations de domaines logiques.

```
primary# ldm list -l -p
VERSION 1.6
DOMAIN|name=primary|state=active|flags=normal,control,vio-service|
cons=SP|ncpu=4|mem=1073741824|util=0.6|uptime=64801|
softstate=Solaris running
VCPU
```

```
|vid=0|pid=0|util=0.9|strand=100
|vid=1|pid=1|util=0.5|strand=100
|vid=2|pid=2|util=0.6|strand=100
|vid=3|pid=3|util=0.6|strand=100
MEMORY
|ra=0x8000000|pa=0x8000000|size=1073741824
IO
|dev=pci@780|alias=bus_a
|dev=pci@7c0|alias=bus_b
...
DOMAIN|name=ldg1|state=active|flags=normal|cons=5000|
ncpu=2|mem=805306368|util=29|uptime=903|
softstate=Solaris running
VCPU
|vid=0|pid=4|util=29|strand=100
|vid=1|pid=5|util=29|strand=100
MEMORY
|ra=0x8000000|pa=0x48000000|size=805306368
...
DOMAIN|name=ldg2|state=active|flags=normal|cons=5001|
ncpu=3|mem=1073741824|util=35|uptime=775|
softstate=Solaris running
VCPU
|vid=0|pid=6|util=35|strand=100
|vid=1|pid=7|util=34|strand=100
|vid=2|pid=8|util=35|strand=100
MEMORY
|ra=0x8000000|pa=0x78000000|size=1073741824
...
```

EXEMPLE 81 Détermination de la CPU virtuelle correspondant à un numéro de CPU physique

La configuration des domaines logiques est présentée dans l'Exemple 80, "Détermination de la configuration des domaines". L'exemple suivant explique comment déterminer le domaine et la CPU virtuelle correspondant au numéro de CPU physique `5`, ainsi que le domaine et l'adresse réelle correspondant à l'adresse physique `0x7e816000`.

En recherchant dans les entrées `VCPU` de la liste l'une des zones `pid` égale à `5`, vous pouvez trouver l'entrée suivante sous le domaine logique `ldg1`.

```
|vid=1|pid=5|util=29|strand=100
```

Par conséquent, le numéro de CPU physique `5` se trouve dans le domaine `ldg1` et dans le domaine, il a le numéro de CPU virtuelle `1`.

En recherchant dans les entrées `MEMORY` de la liste, vous pouvez trouver l'entrée suivante sous le domaine `ldg2`.

```
ra=0x8000000|pa=0x78000000|size=1073741824
```

Où 0x78000000 <= 0x7e816000 <= (0x78000000 + 1073741824 - 1), c'est-à-dire *pa* <= PA <= (*pa* + *taille* - 1). Par conséquent, le PA se trouve dans le domaine `ldg2` et l'adresse réelle correspondante est 0x8000000 + (0x7e816000 - 0x78000000) = 0xe816000.

Utilisation des identificateurs uniques universellement

Un identificateur universel unique (UUID) est assigné à chaque domaine. L'UUID est assigné lorsqu'un domaine est créé. Pour les domaines propriétaire, l'UUID est assigné lorsque le démon `ldmd` s'initialise.

Remarque - L'UUID est perdu si vous utilisez la commande `ldm migrate-domain -f` pour migrer un domaine vers la machine cible qui exécute une version antérieure de Logical Domains Manager. Lorsque vous migrez un domaine d'une machine source qui exécute une version antérieure de Logical Domains Manager, le domaine est associé à un nouvel UUID dans le cadre de la migration. Sinon, l'UUID est migré.

Vous pouvez obtenir l'UUID pour un domaine en exécutant la commande `ldm list -l`, `ldm list-bindings` ou `ldm list -o domain`. Les exemples suivants montrent l'UUID pour le domaine `ldg1` :

```
primary# ldm create ldg1
primary# ldm ls -l ldg1
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
ldg1             inactive   ------

UUID
    6c908858-12ef-e520-9eb3-f1cd3dbc3a59

primary# ldm ls -l -p ldg1
VERSION 1.6
DOMAIN|name=ldg1|state=inactive|flags=|cons=|ncpu=|mem=|util=|uptime=
UUID|uuid=6c908858-12ef-e520-9eb3-f1cd3dbc3a59
```

Commande et API d'information sur le domaine virtuel

La commande `virtinfo` vous permet de rassembler des informations sur un domaine virtuel en cours d'exécution. Vous pouvez également utiliser l'API d'informations sur le domaine virtuel pour créer des programmes destinés à rassembler des informations concernant les domaines virtuels.

La liste suivante montre certaines informations que vous pouvez rassembler sur un domaine virtuel à l'aide de la commande ou de l'API :

- Type de domaine (implémentation, contrôle, invité, E/S, service, root)
- Nom du domaine déterminé par le gestionnaire de domaines logiques
- Identificateur unique universellement (UUID) du domaine
- Nom du noeud de réseau du domaine de contrôle du domaine

- Numéro de série du châssis sur lequel le domaine est en cours d'exécution

Pour plus d'informations sur la commande `virtinfo`, reportez-vous à la page de manuel `virtinfo`(1M). Pour plus d'informations sur l'API, reportez-vous aux pages de manuel `libv12n`(3LIB) et `v12n`(3EXT).

Utilisation des canaux de domaines logiques

Oracle VM Server for SPARC utilise des canaux de domaine logique (LDC) pour implémenter toutes les communications telles que la console, les périphériques d'E/S virtuels et le trafic de contrôle. Un LDC est la méthode utilisée pour établir les communications entre deux adresses. Même si généralement, chaque adresse se trouve dans un domaine différent, les adresses peuvent se trouver dans le même domaine pour permettre les communications loopback.

Ce logiciel et le microprogramme système fournissent un grand pool d'adresses de LDC que vous pouvez utiliser pour le domaine de contrôle et les domaines invités. Ce pool d'adresses de LDC est disponible uniquement pour les serveurs SPARC T4 et SPARC T5, les serveurs des séries SPARC T7 et SPARC M5, les serveurs SPARC M6, les serveurs de série SPARC M7 et SPARC S7 et les Serveurs Fujitsu M10.

Le microprogramme système requis pour prendre en charge le pool d'adresses de LDC est la version 8.5.2 pour les serveurs SPARC T4, la version 9.2.1 pour les serveurs SPARC T5, SPARC M5 et SPARC M6, la version 9.4.3 pour les serveurs des séries SPARC T7 et SPARC M7, toute version publiée pour les serveurs de série SPARC S7 et XCP2240 pour les Serveurs Fujitsu M10.

Les limites d'adresses de LDC suivantes s'appliquent toujours si vous exécutez une version plus ancienne du microprogramme système sur une plate-forme prise en charge ou sur une plate-forme UltraSPARC T2, UltraSPARC T2 Plus ou SPARC T3 :

- **Serveur UltraSPARC T2 –** 512 adresses de LDC
- **Serveur UltraSPARC T2 Plus –** 768 adresses de LDC
- **Serveur SPARC T3 –** 768 adresses de LDC
- **Serveur SPARC T4 –** 768 adresses de LDC
- **Serveur SPARC T5 –** 768 adresses de LDC
- **Serveur de série SPARC T7 –** 768 adresses de LDC
- **Serveur SPARC M5 –** 768 adresses de LDC
- **Serveur SPARC M6 –** 768 adresses de LDC
- **Serveurs de série SPARC M7 –** 768 adresses de LDC
- **Serveurs Fujitsu M10 –** 768 adresses de LDC

Cette restriction peut poser des problèmes sur le domaine de contrôle en raison du nombre potentiellement important d'adresses de LDC utilisées pour les communications de données d'E/S virtuelles et le contrôle Logical Domains Manager des autres domaines.

Si vous essayez d'ajouter un service ou de lier un domaine, de sorte que le nombre d'adresses de LDC dépasse la limite sur un domaine logique donné, l'opération échoue avec un message d'erreur similaire au suivant :

```
13 additional LDCs are required on guest primary to meet this request,
but only 9 LDCs are available
```

Les directives suivantes vous permettent de planifier correctement l'utilisation d'adresses de LDC et vous expliquent pourquoi vous pouvez dépasser les capacités LDC du domaine de contrôle :

- Le domaine de contrôle utilise environ 15 adresses de LDC pour divers objectifs de communication avec l'hyperviseur, l'architecture de gestion des pannes FMA (Fault Management Architecture) et le processeur système (SP), indépendamment du nombre de domaines par ailleurs configurés. Le nombre d'adresses LDC utilisées par le domaine de contrôle dépend de la plate-forme et de la version du logiciel utilisé.
- Le Logical Domains Manager alloue une adresse de LDC au domaine de contrôle de chaque domaine, y compris à lui-même, pour le trafic de contrôle.
- Chaque service d'E/S virtuel sur le domaine de contrôle utilise une adresse de LDC pour chaque client de ce service connecté. Chaque domaine nécessite au moins un réseau virtuel, un disque virtuel et une console virtuelle.

L'équation suivante intègre ces directives pour définir le nombre d'adresses de LDC requis par le domaine de contrôle :

15 + *number-of-domains* + (*number-of-domains* x *number-of-virtual-services*)
= *total-LDC-endpoints*

number-of-domains représente le nombre total de domaines (y compris le domaine de contrôle) et *number-of-virtual-services* représente le nombre total de périphériques d'E/S servis par ce domaine.

L'exemple suivant présente comment utiliser l'équation pour déterminer le nombre d'adresses de LDC lorsqu'il existe un domaine de contrôle et huit autres domaines :

15 + 9 + (8 x 3) = 48 adresses de LDC

L'exemple suivant implique 45 domaines invités et chaque domaine comprend cinq disques virtuels, deux réseaux virtuels et une console virtuelle. Le calcul aboutit au résultat suivant :

15 + 46 + 45 x 8 = 421 adresses de LDC

En fonction du nombre d'adresses de LDC prises en charge par votre plate-forme, le Logical Domains Manager peut accepter ou refuser la configuration.

Si vous manquez d'adresses de LDC sur le domaine de contrôle, envisagez de créer des domaines de service ou des domaines d'E/S pour fournir des services virtuels d'E/S aux domaines invités. Cette opération permet de créer des adresses de LDC sur les domaines d'E/S et les domaines de service au lieu de les créer sur le domaine de contrôle.

Un domaine invité peut également manquer d'adresses de LDC. Cette situation peut être due à la propriété `inter-vnet-link` définie sur `on`, qui associe d'autres adresses de LDC aux domaines invités pour les connecter directement les uns aux autres.

L'équation suivante détermine le nombre d'adresses de LDC requis par un domaine invité lorsque `inter-vnet-link=off` :

2 + *number-of-vnets* + *number-of-vdisks* = *total-LDC-endpoints*

2 représente la console virtuelle et le trafic du contrôle ; *number-of-vnets* correspond au nombre total de périphériques réseau virtuels associés au domaine invité et *number-of-vdisks* représente le nombre total de disques virtuels associés au domaine invité.

L'exemple suivant présente comment utiliser l'équation pour déterminer le nombre d'adresses de LDC par domaine invité si `inter-vnet-link=off` et si vous disposez de deux disques virtuels et de deux réseaux virtuels :

2 + 2 + 2 = 6 adresses de LDC

L'équation suivante détermine le nombre d'adresses de LDC requises par un domaine invité lorsque `inter-vnet-link=on` :

2 + [[(*number-of-vnets-from-vswX* x *number-of-vnets-in-vswX*)] ...] + *number-of-vdisks* = *total-LDC-endpoints*

2 représente la console virtuelle et le trafic de contrôle, *number-of-vnets-from-vswX* représente le nombre total de périphériques de réseau virtuel associés au domaine invité du commutateur virtuel *vswX*, *number-of-vnets-in-vswX* correspond au nombre total de périphériques de réseau virtuel sur le commutateur virtuel *vswX* et *number-of-virtual-disks* représente le nombre total de disques virtuels associés au domaine invité.

L'exemple suivant présente comment utiliser l'équation pour déterminer le nombre d'adresses de LDC par domaine invité si `inter-vnet-link=on` et si vous disposez de deux disques virtuels et de deux commutateurs virtuels. Le premier commutateur virtuel dispose de huit réseaux virtuels et en associe quatre au domaine. Le second commutateur virtuel associe ses huit réseaux virtuels au domaine :

2 + (4 x 8) + (8 x 8) + 2 = 100 adresses de LDC

Par défaut, `inter-vnet-link=auto` est défini pour les réseaux virtuels que vous créez à l'aide du logiciel Oracle VM Server for SPARC 3.4. Cette fonctionnalité désactive automatiquement

les liaisons inter-vnet (inter-vnet-links) si le nombre dépasse le seuil fixé. Toutefois, tous les périphériques de réseau virtuel que vous avez créés avec `inter-vnet-link=on` doivent être modifiés, de façon explicite, en `inter-vnet-link=off` pour réduire le nombre de canaux LDC. Pour plus d'informations, reportez-vous à la section "Canaux LDC inter-Vnet" à la page 257.

Vous pouvez toujours définir `inter-vnet-link=off` pour réduire le nombre d'adresses de LDC dans le ou les domaines disposant de périphériques de réseau virtuel. Cependant, la valeur de propriété `off` n'a aucune incidence sur le domaine de service comportant le commutateur virtuel, qui nécessite toujours une connexion LDC à chaque périphérique réseau virtuel. Lorsque cette propriété est définie sur `off`, les canaux LDC ne sont pas utilisés pour les communications inter-vnet. A la place, un canal LDC est affecté uniquement aux communications entre périphériques réseau virtuels et périphériques commutateurs virtuels. Reportez-vous à la page de manuel `ldm`(1M).

Remarque - Bien que la désactivation de l'affectation des liens inter-vnet réduise le nombre d'adresses de LDC, elle risque d'avoir une incidence négative sur les performances réseau entre invités. Cette dégradation se produit car le trafic de communication entre invités passe par le commutateur virtuel plutôt que de passer directement d'un domaine invité à un autre.

Initialisation d'un grand nombre de domaines

Vous pouvez initialiser le nombre de domaines suivant en fonction de votre serveur :

- Jusqu'à 256 sur les Serveurs Fujitsu M10 par partition physique
- Jusqu'à 128 sur les serveurs de série SPARC M7 par domaine physique
- Jusqu'à 128 sur les serveurs SPARC M6 par domaine physique
- Jusqu'à 128 sur les serveurs SPARC M5 par domaine physique
- Jusqu'à 128 sur les serveurs de série SPARC T7
- Jusqu'à 128 sur les serveurs SPARC T5
- Jusqu'à 128 sur les serveurs SPARC T4
- Jusqu'à 128 sur les serveurs SPARC T3
- Jusqu'à 128 sur les serveurs UltraSPARC T2 Plus
- Jusqu'à 64 sur les serveurs UltraSPARC T2

Si des CPU virtuelles non allouées sont disponibles, assignez-les au domaine de service afin de contribuer au traitement des demandes d'E/S virtuelles. Attribuez 4 à 8 CPU virtuelles au domaine de service si vous créez plus de 32 domaines. Dans les cas où les configurations de domaines maximales ne disposent que d'une CPU dans le domaine de service, ne placez pas une contrainte inutile sur cette CPU lors de la configuration et de l'utilisation du domaine. Les services de commutateur virtuel (`vsw`) doivent être répartis sur l'ensemble des adaptateurs réseau

disponibles dans la machine. Par exemple, si vous initialisez 128 domaines sur un serveur Sun SPARC Enterprise T5240, créez quatre services `vsw`, chacun servant 32 instances réseau virtuelles (`vnet`). L'attribution de plus de 32 instances `vnet` par service `vsw` peut entraîner un blocage forcé du domaine du service.

Pour exécuter les configurations maximales, une machine nécessite une quantité de mémoire suffisante pour prendre en charge les domaines invités. La quantité de mémoire dépend de votre plate-forme et de votre SE. Reportez-vous à la documentation de votre plate-forme, *Oracle Solaris 10 8/11 Installation Guide: Planning for Installation and Upgrade* et *Installing Oracle Solaris 11.3 Systems*.

La mémoire libre et l'espace de swap utilisés dans un domaine invité augmentent lorsque les services `vsw` utilisés par le domaine fournissent des services à de nombreux réseaux virtuels (dans plusieurs domaines). Ceci est dû aux liaisons de poste à poste entre toutes les instances `vnet` connectées à `vsw`. Le domaine de service bénéficie d'une quantité de mémoire supplémentaire. La valeur minimale recommandée est 4 Go lorsque vous exécutez plus de 64 domaines. Démarrez les domaines par groupe de 10 ou moins et attendez qu'ils s'initialisent avant de démarrer le lot suivant. Ce conseil est également valable pour les systèmes d'exploitation des domaines. Vous pouvez réduire le nombre de liaisons en désactivant les liens inter-vnet. Reportez-vous à la section "Canaux LDC inter-Vnet" du manuel *Guide d'administration d'Oracle VM Server for SPARC 3.4*.

Arrêt correct et cycle d'alimentation d'un système Oracle VM Server for SPARC

Si vous avez effectué des changements de configuration depuis le dernier enregistrement d'une configuration sur le processeur de service, avant d'éteindre ou d'effectuer un cycle d'alimentation d'un système Oracle VM Server for SPARC, veillez à enregistrer la dernière configuration que vous souhaitez conserver.

▼ Procédure de mise hors tension d'un système comportant plusieurs domaines actifs

1. **Arrêtez et dissociez tous les domaines non E/S.**
2. **Arrêtez et dissociez tous les domaines d'E/S actifs.**
3. **Passez le domaine à l'état d'initialisation 5.**

```
primary# shutdown -i 5
```

Au lieu de la commande `shutdown`, vous pouvez également utiliser la commande `init 5`.

▼ Procédure d'arrêt et de redémarrage du système

1. **Mettez le système hors tension.**

   Voir "Procédure de mise hors tension d'un système comportant plusieurs domaines actifs" à la page 425.

2. **Mettez le système sous tension à l'aide du SP.**

Persistance des variables Logical Domains

Les mises à jour de variables persistent après la réinitialisation du domaine, mais pas après un cycle d'alimentation du système, à moins qu'elles ne soient lancées à partir du microprogramme OpenBoot sur le domaine de contrôle ou suivies de l'enregistrement de la configuration sur le processeur de service.

Prenez note des conditions suivantes :

- Lorsque le domaine de contrôle se réinitialise, si aucun domaine invité n'est lié, et en l'absence de reconfiguration différée, le processeur de service effectue un cycle d'alimentation du système.
- Lorsque le domaine de contrôle se réinitialise, si des domaines invités sont liés ou actifs (ou si le domaine de contrôle est en pleine reconfiguration retardée), le processeur de service n'effectue pas de cycle d'alimentation du système.

Les variables Logical Domains d'un domaine peuvent être spécifiées en suivant l'une des méthodes suivantes :

- A l'invite OpenBoot.
- A l'aide de la commande SE Oracle Solaris `eeprom`(1M).
- A l'aide de la CLI Logical Domains Manager (`ldm`).
- De manière limitée, à partir du processeur de service (SP) à l'aide de la commande `bootmode`. Cette méthode ne peut être utilisée que pour certaines variables et uniquement dans la configuration `factory-default`.

Les mises à jour de variables effectuées à l'aide de l'une de ces méthodes doivent toujours persister après les réinitialisations du domaine. Les mises à jour de variables s'appliquent toujours dans toutes les configurations de domaine ultérieures enregistrées sur le processeur de service.

Dans le logiciel Oracle VM Server for SPARC 3.4, il y a quelques cas où les mises à jour de variables ne persistent pas comme prévu :

- Toutes les méthodes de mise à jour d'une variable persistent pour toutes les réinitialisations de ce domaine. Toutefois, elles ne persistent pas après un cycle d'alimentation du système, sauf si une configuration de domaine logique ultérieure est enregistrée sur le processeur de service.

  Toutefois, sur le domaine de contrôle, les mises à jour effectuées en utilisant les commandes du microprogramme OpenBoot ou la commande `eeprom` *persistent* après un cycle d'alimentation du système et ce, même sans enregistrer par la suite une nouvelle configuration de domaine logique sur le SP. La commande `eeprom` prend en charge ce comportement sur les serveurs à partir de la série SPARC T4.

Si vous craignez que les variables Logical Domains changent, procédez de l'une des manières suivantes :

- Accédez à l'invite `ok` du système et mettez à jour les variables.
- Mettez à jour les variables pendant que Logical Domains Manager est désactivé :

  ```
  # svcadm disable ldmd
  update variables
  # svcadm enable ldmd
  ```

- Si vous exécutez Live Upgrade, effectuez les opérations suivantes :

  ```
  # svcadm disable -t ldmd
  # luactivate be3
  # init 6
  ```

Si vous modifiez l'heure ou la date sur un domaine logique, par exemple en utilisant la commande `ntpdate`, le changement persiste après toutes les réinitialisations du domaine, mais pas après un cycle d'alimentation de l'hôte. Pour que les changements de date et d'heure persistent, enregistrez la configuration avec le changement en question sur le contrôleur système et initialisez à partir de cette configuration.

Les ID de bogue suivants ont été créés pour résoudre ces problèmes : 15375997, 15387338, 15387606 et 15415199.

Ajustement de la limite d'interruption

Le matériel fournissant un nombre limité d'interruptions, Oracle Solaris limite le nombre d'interruptions que chaque périphérique peut utiliser. La limite par défaut doit correspondre aux

besoins d'une configuration système type, sachant que la valeur devra peut-être être ajustée pour certaines configurations système.

Remarque - Ces limitations ne s'appliquent pas aux serveurs de série SPARC M7 et SPARC T7.

Lorsque vous activez la virtualisation d'E/S sur un bus PCIe, des ressources matérielles d'interruptions sont affectées à chaque domaine d'E/S. Chaque domaine se voit attribuer un nombre limité de ces ressources, ce qui peut conduire à des problèmes d'allocations d'interruptions. Cette situation n'a d'incidence que sur les plates-formes UltraSPARC T2, UltraSPARC T2 Plus, SPARC T3, SPARC T4, SPARC T5, SPARC M5 et SPARC M6.

Cet avertissement sur la console Oracle Solaris signifie que l'approvisionnement d'interruptions a été épuisé lorsque les pilotes de périphériques d'E/S ont été joints :

```
WARNING: ddi_intr_alloc: cannot fit into interrupt pool
```

Plus précisément, la limite peut nécessiter des ajustements si le système est divisé en plusieurs domaines logiques et si un nombre trop important de périphériques d'E/S est assigné à un domaine invité. Oracle VM Server for SPARC divise le nombre total des interruptions en ensembles plus petits et les assigne à des domaines invités. Si un nombre trop important de périphériques d'E/S est assigné à un domaine invité, ses approvisionnements d'interruptions risquent d'être trop faibles pour fournir à chaque périphérique la limite par défauts d'interruptions. Le domaine invité épuise donc son approvisionnement en interruptions avant d'être entièrement associé à tous les pilotes.

Certains pilotes fournissent une routine de rappels facultatifs qui permet au SE Oracle Solaris d'ajuster automatiquement leurs interruptions. La limite par défaut ne s'applique pas à ces pilotes.

Déterminez comment les interruptions sont utilisées à l'aide des macros MDB `::irmpools` et `::irmreqs`. La macro `::irmpools` affiche l'approvisionnement global des interruptions divisées en pools. La macro `::irmreqs` affiche les périphériques qui sont mappés vers chaque pool. Pour chaque périphérique, `::irmreqs` affiche si la limite par défaut est appliquée par une routine de rappels facultatifs, le nombre d'interruptions demandées par chaque pilote et le nombre d'interruptions de chaque pilote.

Bien que ces macros n'affichent pas d'informations sur les pilotes qui n'ont pas pu être joints, vous pouvez utiliser des informations pour calculer l'étendue de l'ajustement de la limite par défaut. Vous pouvez forcer un périphérique qui utilise plus d'une interruption sans fournir de routine de rappels pour utiliser moins d'interruptions en ajustant la limite par défaut. Pour de tels dispositifs, réduire la limite par défaut permet de libérer des interruptions pouvant être utilisées par d'autres périphériques.

Pour ajuster la limite par défaut, définissez la propriété `ddi_msix_alloc_limit` sur une valeur comprise entre `1` to `8` dans le fichier `/etc/system`. Réinitialisez ensuite le système pour que la modification prenne effet.

Pour plus d'informations sur la création ou la mise à jour des valeurs de propriété `/etc/system`, reportez-vous à la section "Mise à jour des valeurs de propriété dans le fichier `/etc/system`" à la page 409.

Pour optimiser les performances, commencez par affecter de grandes valeurs et réduisez les valeurs par petits incréments jusqu'à la réussite de l'initialisation du système, sans avertissements. Utilisez les macros `::irmpools` et `::irmreqs` pour mesurer l'impact de l'ajustement sur tous les pilotes joints.

Par exemple, supposez que les avertissements suivants sont émis lors de l'initialisation du SE Oracle Solaris dans un domaine invité :

```
WARNING: emlxs3: interrupt pool too full.
WARNING: ddi_intr_alloc: cannot fit into interrupt pool
```

Les macros `::irmpools` et `::irmreqs` affichent les informations suivantes :

```
# echo "::irmpools" | mdb -k
ADDR             OWNER   TYPE   SIZE  REQUESTED  RESERVED
00000400016be970 px#0    MSI/X  36    36         36

# echo "00000400016be970::irmreqs" | mdb -k
ADDR             OWNER   TYPE   CALLBACK NINTRS NREQ NAVAIL
00001000143acaa8 emlxs#0 MSI-X  No       32     8    8
00001000170199f8 emlxs#1 MSI-X  No       32     8    8
000010001400ca28 emlxs#2 MSI-X  No       32     8    8
0000100016151328 igb#3   MSI-X  No       10     3    3
00001000019549d30 igb#2   MSI-X  No       10     3    3
0000040000e0f878 igb#1   MSI-X  No       10     3    3
000010001955a5c8 igb#0   MSI-X  No       10     3    3
```

La limite par défaut dans cet exemple comporte 8 interruptions par périphérique, ce qui n'est pas suffisant pour lier le périphérique `emlxs3` final au système. En supposant que toutes les instances `emlxs` se comportent de la même manière, `emlxs3` a probablement demandé 8 interruptions.

En soustrayant les 12 interruptions utilisées par tous les périphériques `igb` de la taille totale du pool des 36 interruptions, 24 interruptions sont disponibles pour les périphériques `emlxs`. La division des 24 interruptions par 4 suggère que 6 interruptions par périphérique permettent à tous les périphériques `emlxs` de se joindre avec des performances égales. L'ajustement suivant est ainsi ajouté au fichier `/etc/system` :

```
set ddi_msix_alloc_limit = 6
```

Pour plus d'informations sur la création ou la mise à jour des valeurs de propriété `/etc/system`, reportez-vous à la section "Mise à jour des valeurs de propriété dans le fichier `/etc/system`" à la page 409.

Lorsque le système réussit à s'initialiser sans avertissement, les macros `::irmpools` et `::irmreqs` affichent les informations mises à jour suivantes :

```
primary# echo "::irmpools" | mdb -k
ADDR             OWNER   TYPE   SIZE  REQUESTED  RESERVED
00000400018ca868 px#0    MSI/X  36    36         36
```

```
# echo "00000400018ca868::irmreqs" | mdb -k
ADDR             OWNER   TYPE   CALLBACK NINTRS NREQ NAVAIL
0000100016143218 emlxs#0 MSI-X  No       32     8    6
0000100014269920 emlxs#1 MSI-X  No       32     8    6
000010001540be30 emlxs#2 MSI-X  No       32     8    6
00001000140cbe10 emlxs#3 MSI-X  No       32     8    6
00001000141210c0 igb#3   MSI-X  No       10     3    3
0000100017549d38 igb#2   MSI-X  No       10     3    3
0000040001ceac40 igb#1   MSI-X  No       10     3    3
000010001acc3480 igb#0   MSI-X  No       10     3    3
```

Liste des dépendances d'E/S de domaine

Les opérations d'E/S d'un domaine sont souvent fournies par un autre domaine, comme un domaine de service ou un domaine d'E/S. Par exemple, un domaine de service peut exporter un périphérique virtuel ou un domaine root peut fournir un accès direct à un périphérique physique.

Faites attention à ces dépendances d'E/S implicites, car une interruption dans un domaine de service ou un domaine root se solde également par une interruption de service du domaine dépendant.

Vous pouvez utiliser la commande `ldm list-dependencies` pour visualiser les dépendances d'E/S entre les domaines. Outre l'établissement d'une liste de dépendances d'un domaine, vous pouvez inverser la sortie pour afficher les domaines dépendants d'un domaine en particulier.

La liste ci-dessous répertorie les types de dépendances d'E/S que vous pouvez visualiser à l'aide de la commande `ldm list-dependencies` :

`VDISK` — Dépendances créée lorsqu'un disque virtuel est connecté à un moteur de traitement de disque virtuel qui a été exporté par un serveur de disque virtuel

`VNET` — Dépendance créée lorsqu'un périphérique réseau virtuel est connecté à un commutateur virtuel

`IOV` — Dépendance créée lorsqu'une fonction virtuelle SR-IOV est associée à une fonction physique SR-IOV

Les commandes `ldm list-dependencies` suivantes indiquent quelques façons permettant de visualiser les informations sur la dépendance du domaine :

- Pour afficher des informations détaillées sur la dépendance du domaine, utilisez l'option `-l`.

```
primary# ldm list-dependencies -l
DOMAIN         DEPENDENCY      TYPE      DEVICE
primary
```

```
svcdom
ldg0            primary         VDISK     primary-vds0/vdisk0
                                VNET      primary-vsw0/vnet0
                svcdom          VDISK     svcdom-vds0/vdisk1
                                VNET      svcdom-vsw0/vnet1
ldg1            primary         VDISK     primary-vds0/vdisk0
                                VNET      primary-vsw0/vnet0
                                IOV       /SYS/MB/NET0/IOVNET.PF0.VF0
                svcdom          VDISK     svcdom-vds0/vdisk1
                                VNET      svcdom-vsw0/vnet1
                                IOV       /SYS/MB/NET2/IOVNET.PF0.VF0
```

- Pour afficher des informations détaillées sur les domaines dépendants groupés en fonction de leur dépendances, utilisez les commandes `-l` et `-r`.

```
primary# ldm list-dependencies -r -l
DOMAIN          DEPENDENT       TYPE      DEVICE
primary         ldg0            VDISK     primary-vds0/vdisk0
                                VNET      primary-vsw0/vnet0
                ldg1            VDISK     primary-vds0/vdisk0
                                VNET      primary-vsw0/vnet0
                                IOV       /SYS/MB/NET0/IOVNET.PF0.VF0
svcdom          ldg0            VDISK     svcdom-vds0/vdisk1
                                VNET      svcdom-vsw0/vnet1
                ldg1            VDISK     svcdom-vds0/vdisk1
                                VNET      svcdom-vsw0/vnet1
                                IOV       /SYS/MB/NET2/IOVNET.PF0.VF0
```

Activation du démon de Logical Domains Manager

Le démon de Logical Domains Manager, `ldmd`, est automatiquement activé lorsque le package logiciel Oracle VM Server for SPARC est installé. Une fois que le démon est activé, vous pouvez créer, modifier et contrôler les domaines logiques.

Sur les serveurs des séries SPARC T7, SPARC M7 et SPARC S7, le service d'interconnexion ILOM permet la communication entre le démon `ldmd` et le processeur de service (SP). Le service `ilomconfig-interconnect` est activé par défaut. Pour vérifier que le service d'interconnexion ILOM est activé, reportez-vous à la section "Procédure de vérification de la configuration d'interconnexion ILOM" à la page 45.

Attention - Ne désactivez pas le service `ilomconfig-interconnect`. La désactivation de ce service pourrait empêcher le fonctionnement correct des domaines logiques et du SE.

▼ Procédure d'activation du démon de Logical Domains Manager

Utilisez cette procédure pour activer le démon `ldmd` s'il a été désactivé.

1. **Utilisez la commande `svcadm` pour activer le démon de Logical Domains Manager, `ldmd`.**

   ```
   # svcadm enable ldmd
   ```

   Pour plus d'informations sur la commande `svcadm`, reportez-vous à la page de manuel `svcadm`(1M).

2. **Assurez-vous que Logical Domains Manager est en cours d'exécution.**

   La commande `ldm list`doit répertorier tous les domaines actuellement définis sur le système. Le domaine `primary` en particulier doit être répertorié et être à l'état `active`. L'exemple de sortie suivant indique que seul le domaine `primary` est défini sur le système.

   ```
   # ldm list
   NAME             STATE    FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
   primary          active   ---c-   SP      64    3264M    0.3%  19d 9m
   ```

Sauvegarde et restauration des données de configuration enregistrées automatiquement

Les sections suivantes décrivent comment enregistrer et restaurer les répertoires de configuration sauvegardés automatiquement et le fichier de la base de données de contraintes.

Sauvegarde et restauration des répertoires de configuration enregistrés automatiquement

Vous pouvez enregistrer et restaurer les répertoires de configuration d'enregistrement automatique avant d'installer le système d'exploitation sur le domaine de contrôle. Chaque fois que vous réinstallez le système d'exploitation sur le domaine de contrôle, vous devez enregistrer et restaurer les données de configuration du domaine sauvegardées automatiquement, qui se trouvent dans les répertoires `/var/share/ldomsmanager/autosave-`*autosave-name*.

Vous pouvez utiliser la commande `tar` ou `cpio` pour sauvegarder et restaurer tout le contenu des répertoires.

Remarque - Chaque répertoire enregistré automatiquement comprend un horodatage pour la dernière mise à jour de la configuration du SP pour la configuration concernée. Si vous restaurez les fichiers enregistrés automatiquement, l'horodatage risque de ne plus être synchronisé. Dans ce cas, les configurations enregistrées automatiquement restaurées sont affichées dans leur état précédent, `[newer]` ou à jour.

Pour plus d'informations sur les configurations enregistrées automatiquement, reportez-vous à la section "Gestion des configurations de domaine" du manuel *Guide d'administration d'Oracle VM Server for SPARC 3.4*.

▼ Procédure d'enregistrement et de restauration des répertoires de sauvegarde automatique

1. **Sauvegardez les répertoires enregistrés automatiquement.**

```
# cd /
# tar -cvpf autosave.tar var/share/ldomsmanager/autosave-*
```

2. **(Facultatif) Supprimez les répertoires enregistrés automatiquement pour garantir une opération de restauration propre.**

   Parfois, un répertoire enregistré automatiquement peut comprendre des fichiers inutiles, peut-être laissés par une configuration précédente, risquant d'endommager la configuration qui a été téléchargée sur le SP. Dans ce cas, nettoyez le répertoire enregistré automatiquement avant l'opération de restauration comme indiqué dans cet exemple :

```
# cd /
# rm -rf var/share/ldomsmanager/autosave-*
```

3. **Restaurez les répertoires enregistrés automatiquement.**

   Ces commandes restaurent les fichiers et les répertoires du répertoire `/var/share/ldomsmanager`.

```
# cd /
# tar -xvpf autosave.tar
```

Sauvegarde et restauration du fichier de la base de données de contraintes de Logical Domains

Chaque fois que vous mettez à niveau le système d'exploitation sur le domaine de contrôle, vous devez enregistrer et restaurer le fichier de base de données de contraintes Logical Domains `/var/share/ldomsmanager/ldom-db.xml`.

Enregistrez et restaurez également le fichier `/var/share/ldomsmanager/ldom-db.xml` lorsque vous effectuez d'autres opérations qui détruisent des données du fichier du domaine de contrôle, telles qu'un swap de disque.

Remarque - Le répertoire `/var/share/ldomsmanager` est partagé entre tous les environnements d'initialisation. `/var/opt/SUNWldm` est un lien symbolique vers `/var/share/ldomsmanager` pour garantir la compatibilité ascendante.

Configuration usine par défaut et désactivation de domaines

La configuration initiale sur laquelle la plate-forme apparaît en tant que système unique hébergeant un seul système d'exploitation est appelée configuration usine par défaut. Si vous souhaitez désactiver les domaines logiques, vous voudrez probablement également restaurer cette configuration afin que le système récupère l'accès à toutes les ressources (CPU, mémoire, E/S), pouvant avoir été assignées à d'autres domaines.

Cette section décrit la procédure pour supprimer tous les domaines invités, supprimer toutes les configurations de domaines et remettre la configuration sur les valeurs par défaut d'usine.

▼ Procédure de suppression de tous les domaines invités

1. **Arrêtez tous les domaines.**

```
primary# ldm stop-domain -a
```

2. **Dissociez tous les domaines sauf le domaine `primary`.**

```
primary# ldm unbind-domain ldom
```

Remarque - Vous risquez de ne pas pouvoir dissocier un domaine d'E/S s'il fournit des services requis par le domaine de contrôle. Dans ce cas, passez cette étape.

3. **Détruisez tous les domaines sauf le domaine `primary`.**

```
primary# ldm remove-domain -a
```

Procédure de suppression des configurations de domaine

1. **Répertoriez toutes les configurations de domaines stockées sur le processeur de service (SP).**

   ```
   primary# ldm list-config
   ```

2. **Supprimez toutes les configurations (*config-name*) enregistrées précédemment sur le SP sauf la configuration `factory-default`.**
   Utilisez la commande suivante pour chacune de ces configurations :

   ```
   primary# ldm rm-config config-name
   ```

   Après avoir supprimé toutes les configurations précédemment enregistrées sur le SP, le domaine `factory-default` est le domaine suivant à utiliser lorsque le domaine de contrôle (`primary`) est réinitialisé.

Procédure de restauration de la configuration usine par défaut

1. **Sélectionnez la configuration usine par défaut.**

   ```
   primary# ldm set-config factory-default
   ```

2. **Arrêtez le domaine de contrôle.**

   ```
   primary# shutdown -i5 -g0 -y
   ```

3. **Effectuez un cycle d'alimentation du système pour charger la configuration usine par défaut.**

   ```
   -> stop /SYS
   -> start /SYS
   ```

Procédure de désactivation de Logical Domains Manager

La désactivation de Logical Domains Manager n'arrête pas les domaines en cours d'exécution, mais désactive la capacité à créer de nouveaux domaines, à modifier la configuration des domaines existants ou à surveiller l'état des domaines.

Attention - Si vous désactivez Logical Domains Manager, vous désactivez certains services, notamment le compte-rendu des erreurs et la gestion de l'alimentation. Dans le cas du compte-rendu d'erreurs, si vous êtes dans la configuration `factory-default`, vous pouvez réinitialiser le domaine de contrôle pour restaurer le compte-rendu d'erreurs. Toutefois, vous ne pouvez pas réactiver la gestion de l'alimentation. Par ailleurs, certains outils de gestion ou de surveillance du système reposent sur Logical Domains Manager.

- **Désactivez Logical Domains Manager à partir du domaine de contrôle.**

```
primary# svcadm disable ldmd
```

Procédure de restauration de la configuration usine par défaut à partir du processeur de service

Vous pouvez restaurer la configuration usine par défaut à partir du processeur de service.

1. **Restaurez la configuration usine par défaut à partir du processeur de service.**

```
-> set /HOST/bootmode config=factory-default
```

2. **Effectuez un cycle d'alimentation du système pour charger la configuration usine par défaut.**

```
-> reset /SYS
```

♦ ♦ ♦ ANNEXE A

Utilisation de la gestion de l'alimentation

Cette annexe contient des informations relatives à l'utilisation de la gestion de l'alimentation sur les produits Oracle VM Server for SPARC.

Utilisation de la gestion de l'alimentation

Pour activer la gestion de l'alimentation (PM), vous devez d'abord définir la stratégie PM dans le microprogramme Oracle Integrated Lights Out Manager (ILOM) 3.0. Cette section récapitule les informations nécessaires afin de pouvoir utiliser le mode PM avec le logiciel Oracle VM Server for SPARC.

Pour plus d'informations sur ILOM, reportez-vous aux documents suivants :

- “Surveillance de la consommation d'énergie” du *Oracle Integrated Lights Out Manager (ILOM) 3.0 CLI Procedures Guide*
- *Mises à jour des fonctions et notes de version d'Oracle Integrated Lights Out Manger (ILOM) 3.0*

La stratégie d'alimentation régit l'utilisation de l'énergie du système à tout moment. Les stratégies d'alimentation suivantes sont prises en charge, dans la mesure où la plate-forme sous-jacente a implémenté les fonctions PM :

- **Désactivé.** Autorise le système à utiliser la totalité de l'énergie disponible.
- **Performances.** Permet d'activer une ou plusieurs des fonctions PM ayant une incidence négligeable sur les performances :
  - Désactivation automatique du coeur de CPU
  - Saut du cycle d'horloge de la CPU
  - Tension CPU dynamique et mise à l'échelle de la fréquence (DVFS)
  - Mise à l'échelle du lien de cohérence

  - Power Aware Dispatcher (PAD) Oracle Solaris
- **Elastique.** Adapte l'utilisation de l'alimentation du système au niveau actuel d'utilisation à l'aide des fonctions PM décrites dans la section des performances. Par exemple, l'état d'alimentation des ressources est réduit lorsque l'utilisation diminue.

Fonctions de gestion de l'alimentation

Les fonctions PM sont les suivantes :

- **Désactivation automatique du coeur de CPU.** Lorsque la stratégie des performances ou la stratégie élastique est appliquée, le Logical Domains Manager désactive automatiquement un coeur de CPU lorsque tous les threads matériels (strands) de ce coeur ne sont pas liés à un domaine. Cette fonctionnalité est uniquement disponible pour les plates-formes UltraSPARC T2, UltraSPARC T2 Plus, SPARC T3 et SPARC T4.
- **Saut du cycle d'horloge de la CPU.** Lorsque la stratégie élastique est activée, Logical Domains Manager ajuste automatiquement le nombre de cycles d'horloge exécutant les instructions dans les ressources de CPU suivantes qui sont liées aux domaines :
  - Processeurs (SPARC T3 ou SPARC T4 pour les domaines qui exécutent le SE Oracle Solaris 10 ou Oracle Solaris 11)
  - Noyaux (SPARC M5 uniquement pour les domaines qui exécutent le SE Oracle Solaris 10)
  - Paires de noyaux (SPARC T5 ou SPARC M6 uniquement pour les domaines qui exécutent le SE Oracle Solaris 10)
  - SPARC Cache Cluster (SCC) (serveurs des séries SPARC T7, SPARC M7 et SPARC S7 uniquement pour les domaines qui exécutent le SE Oracle Solaris 10)

  Logical Domains Manager ignore également le cycle si le processeur, le noyau ou la paire de noyaux ou SCC ne sont liés à aucun strand.
- **Tension CPU dynamique et mise à l'échelle de la fréquence (DVFS).** Lorsque la stratégie élastique est activée, Logical Domains Manager ajuste automatiquement la fréquence d'horloge des processeurs ou des SCC qui sont liés aux domaines exécutés sous le SE Oracle Solaris 10. Logical Domains Manager permet également de limiter la fréquence d'horloge sur les processeurs SPARC T5, SPARC M5 et SPARC M6 qui ne sont liés à aucun strand. Sur les serveurs de série SPARC T7, la fréquence d'horloge est réduite sur les SCC. Cette fonctionnalité est disponible uniquement sur les serveurs SPARC T5, les serveurs de série SPARC T7, les serveurs SPARC M5, les serveurs SPARC M6 et les serveurs des séries SPARC M7 et SPARC S7.
- **Mise à l'échelle du lien de cohérence.** Lorsque la stratégie élastique est activée, Logical Domains Manager fait en sorte que l'hyperviseur ajuste automatiquement le nombre de liens de cohérence utilisés. Cette fonction n'est pas disponible sur les systèmes SPARC T5-2.

- **Limite de puissance.** Vous pouvez définir une *limite de puissance* sur les serveurs SPARC T3, SPARC T4, SPARC T5, les serveurs de série SPARC T7, les serveurs SPARC M5 et SPARC M6 et les serveurs des séries SPARC M7 et SPARC S7 pour limiter la consommation d'énergie d'un système. Si l'énergie consommée est supérieure à la limite de puissance, la gestion de l'alimentation utilise des techniques pour réduire la consommation. Vous pouvez utiliser le processeur de service (SP) d'ILOM pour définir la limite de puissance.

  Consultez les documents suivants :

  - *Oracle Integrated Lights Out Manager (ILOM) 3.0 CLI Procedures Guide*
  - *Mises à jour des fonctions et notes de version d'Oracle Integrated Lights Out Manger (ILOM) 3.0*

  Vous pouvez utiliser l'interface d'ILOM pour définir une limite de puissance, une période de grâce et une action de violation. Si la limite de puissance est dépassée pendant plus longtemps que la période de grâce, l'action de violation est effectuée.

  Si la consommation d'énergie actuelle dépasse la limite de puissance, une tentative est faite pour réduire l'état d'alimentation des CPU. Si la consommation d'énergie chute en dessous de la limite de puissance, l'état d'alimentation de ces ressources peut augmenter. Si la stratégie élastique est en vigueur sur le système, une augmentation de l'état d'alimentation des ressources est générée par le niveau d'utilisation.
- **Solaris Power Aware Dispatcher (PAD).** Un domaine invité qui exécute le système d'exploitation Oracle Solaris 11.1 utilise le Power Aware Dispatcher (PAD) sur les serveurs SPARC T5, les serveurs de série SPARC T7, les serveurs SPARC M5, les serveurs SPARC M6 et les serveurs des séries SPARC M7 et SPARC S7 pour réduire la consommation d'énergie des ressources inactives ou sous-utilisées. La fonction PAD, contrairement à Logical Domains Manager, ajuste le niveau du saut du cycle d'horloge de la CPU ou du SCC et le niveau DVFS.

Pour obtenir des instructions sur la configuration de la stratégie d'alimentation à l'aide de la CLI du microprogramme ILOM 3.0, reportez-vous à la section relative à la “surveillance de la consommation d'énergie” du *Oracle Integrated Lights Out Manager (ILOM) 3.0 CLI Procedures Guide*.

Affichage des données de consommation d'énergie

Le module d'observabilité de la gestion de l'alimentation Power Management (PM) Observability et la commande `ldmpower` vous permettent de visualiser les données de consommation d'énergie des threads de CPU pour vos domaines.

Le module PM Observability est activé par défaut car la propriété `ldmd/pm_observability_enabled` de l'utilitaire de gestion des services (SMF, Service Management Facility) est définie sur `true`. Reportez-vous à la page de manuel `ldmd`(1M).

La commande `ldmpower` comporte les options et les opérandes suivantes permettant de personnaliser les données de consommation d'énergie consignées :

```
ldmpower [-ehiprstvx | -o hours | -m minutes] | -c resource [-l domain-name[,domain-name[,...]]]
  [interval [count]]
```

Pour plus d'informations sur les options, reportez-vous à la page de manuel `ldmpower`(1M).

Pour exécuter cette commande en tant qu'utilisateur non privilégié, vous devez bénéficier du profil de droits `LDoms Power Mgmt Observability`. Si le profil de droits `LDoms Management` ou `LDoms Review` vous a déjà été attribué, vous êtes automatiquement autorisé à exécuter la commande `ldmpower`.

Pour plus d'informations sur la manière dont Oracle VM Server for SPARC utilise les droits, reportez-vous à la section "Profils contenus dans Logical Domains Manager" à la page 35.

Ces profils de droits peuvent être affectés directement à des utilisateurs ou ils peuvent être affectés à un rôle qui sera à son tour affecté à des utilisateurs. Lorsque l'un de ces profils est directement affecté à un utilisateur, vous devez utiliser la commande `pfexec` ou un shell de profil tel que `pfbash` ou `pfksh` pour pouvoir utiliser la commande `ldmpower` afin d'afficher les données de consommation d'énergie des threads de la CPU. Reportez-vous à la section "Délégation de la gestion de Logical Domains à l'aide de droits" à la page 31.

Les exemples suivants indiquent comment activer le module PM Observability et décrivent des modes de collecte des données de consommation d'énergie pour les CPU assignées à vos domaines.

EXEMPLE 82 Activation du module PM Observability

La commande suivante active le module PM Observability en définissant la propriété `ldmd/pm_observability_enabled` sur `true` lorsque la propriété est définie sur `false`.

```
# svccfg -s ldmd setprop ldmd/pm_observability_enabled=true
# svcadm refresh ldmd
# svcadm restart ldmd
```

EXEMPLE 83 Utilisation d'un shell de profil pour obtenir les données de consommation d'énergie des threads de la CPU à l'aide des rôles et des profils de droits

- L'exemple suivant montre comment créer le rôle `ldmpower` avec le profil de droits `LDoms Power Mgmt Observability` qui vous permet d'exécuter la commande `ldmpower`.

```
primary# roleadd -P "LDoms Power Mgmt Observability" ldmpower
primary# passwd ldmpower
New Password:
Re-enter new Password:
passwd: password successfully changed for ldmpower
```

Cette commande assigne le rôle `ldmpower` à l'utilisateur `sam`.

```
primary# usermod -R ldmpower sam
```

L'utilisateur `sam` prend le rôle `ldmpower` et peut utiliser la commande `ldmpower`. Par exemple :

```
$ id
uid=700299(sam) gid=1(other)
$ su ldmpower
Password:
$ pfexec ldmpower
Processor Power Consumption in Watts
DOMAIN  15_SEC_AVG  30_SEC_AVG  60_SEC_AVG
primary 75          84          86
gdom1   47          24          19
gdom2   10          24          26
```

- L'exemple suivant illustre l'utilisation des profils de droits pour exécuter la commande `ldmpower` :

  Affectez le profil de droits à un utilisateur.

  ```
  primary# usermod -P +"LDoms Power Mgmt Observability" sam
  ```

  Les commandes suivantes indiquent comment vérifier que l'utilisateur est bien `sam` et que les profils de droits `All`, `Basic Solaris User` et `LDoms Power Mgmt Observability` sont actifs.

  ```
  $ id
  uid=702048(sam) gid=1(other)
  $ profiles
  All
  Basic Solaris User
  LDoms Power Mgmt Observability
  $ pfexec ldmpower
  Processor Power Consumption in Watts
  DOMAIN  15_SEC_AVG  30_SEC_AVG  60_SEC_AVG
  primary 75          84          86
  gdom1   47          24          19
  gdom2   10          24          26
  ```

EXEMPLE 84 Affichage des données de consommation d'énergie du processeur

Les exemples suivants indiquent comment utiliser la commande `ldmpower` pour afficher les données consignées de consommation d'énergie du processeur pour vos domaines.

- La commande suivante affiche les données de consommation d'énergie moyenne du processeur dans des délais de 15, 30 et 60 secondes et pour tous les domaines :

```
primary# ldmpower
Processor Power Consumption in Watts
DOMAIN  15_SEC_AVG  30_SEC_AVG  60_SEC_AVG
primary 75          84          86
gdom1   47          24          19
gdom2   10          24          26
```

- La commande suivante affiche les données de consommation d'énergie extrapolée pour tous les domaines : `primary`, `gdom1` et `gdom2`.

```
primary# ldmpower -x
System Power Consumption in Watts
DOMAIN  15_SEC_AVG  30_SEC_AVG  60_SEC_AVG
primary 585/57.47%  701/68.96%  712/70.22%
gdom1   132/12.97%  94/9.31%    94/9.30%
gdom2   298/29.27%  218/21.47%  205/20.22%
```

- La commande suivante affiche les données de consommation d'énergie instantanée du processeur pour les domaines `gdom2` et `gdom5`. Elle rapporte les données toutes les dix secondes à cinq reprises.

```
primary# ldmpower -itl gdom2,gdom5 10 5
Processor Power Consumption in Watts
DOMAIN          TIMESTAMP               INSTANT
gdom2           2013.05.17 11:14:45     13
gdom5           2013.05.17 11:14:45     24

gdom2           2013.05.17 11:14:55     18
gdom5           2013.05.17 11:14:55     26

gdom2           2013.05.17 11:15:05     9
gdom5           2013.05.17 11:15:05     16

gdom2           2013.05.17 11:15:15     15
gdom5           2013.05.17 11:15:15     19

gdom2           2013.05.17 11:15:25     12
gdom5           2013.05.17 11:15:25     18
```

- La commande suivante affiche les données de consommation d'énergie moyenne au cours des 12 dernières heures pour tous les domaines. Les données sont affichées à des intervalles d'une heure à partir du dernier calcul horaire demandé.

```
primary# ldmpower -eto 12
Per domain MINIMUM and MAXIMUM power consumption ever recorded:
primary         2013.05.17 08:53:06     3               Min Processors
primary         2013.05.17 08:40:44     273             Max Processors
gdom1           2013.05.17 09:56:35     2               Min Processors
gdom1           2013.05.17 08:53:06     134             Max Processors
gdom2           2013.05.17 10:31:55     2               Min Processors
gdom2           2013.05.17 08:56:35     139             Max Processors

primary         2013.05.17 08:53:06     99              Min Memory
primary         2013.05.17 08:40:44     182             Max Memory
gdom1           2013.05.17 09:56:35     13              Min Memory
gdom1           2013.05.17 08:53:06     20              Max Memory
gdom2           2013.05.17 10:31:55     65              Min Memory
gdom2           2013.05.17 08:56:35     66              Max Memory

Processor Power Consumption in Watts
12 hour's worth of data starting from 2013.05.16 23:17:02
DOMAIN          TIMESTAMP               1 HOUR AVG
primary         2013.05.17 09:37:35     112
gdom1           2013.05.17 09:37:35     15
gdom2           2013.05.17 09:37:35     26

primary         2013.05.17 10:37:35     96
gdom1           2013.05.17 10:37:35     12
gdom2           2013.05.17 10:37:35     21

primary         2013.05.17 11:37:35     85
gdom1           2013.05.17 11:37:35     11
gdom2           2013.05.17 11:37:35     23
...
```

Glossaire

Ce glossaire définit la terminologie, les abréviations et les acronymes de la documentation d'Oracle VM Server for SPARC.

A

API Interface de programmation d'application.

ASN Abstract Syntax Notation.

audit Suivi des modifications apportées au système et identification de l'utilisateur qui a apporté les modifications.

`auditreduce` Commande permettant de fusionner et de sélectionner des enregistrements d'audit à partir de fichiers de piste d'audit (voir la page de manuel `auditreduce`(1M)).

autorisation Méthode s'appuyant sur les droits du SE Oracle Solaris pour désigner les personnes autorisées à effectuer des tâches et à accéder aux données.

B

`bge` Pilote Broadcast Gigabit Ethernet sur périphériques BCM57xx.

BSM Module de sécurité de base.

`bsmconv` Commande permettant d'activer le BSM (voir la page de manuel `bsmconv`(1M)).

`bsmunconv` Commande permettant de désactiver le BSM (voir la page de manuel `bsmunconv`(1M)).

C

CMT Chip multithreading.

configuration Nom de la configuration de domaine logique enregistrée sur le processeur de service.

conformité Détermine si la configuration d'un système est en conformité avec un profil de sécurité prédéfini.

contraintes Les contraintes de Logical Domains Manager sont une ou plusieurs ressources que vous voulez assigner à un domaine particulier. Soit vous recevez toutes les ressources dont vous avez demandé l'ajout au domaine, soit vous ne recevez aucune d'entre elles, en fonction des ressources disponibles.

CWQ Control Word Queue; unité cryptographique.

domaine de contrôle Domaine privilégié qui crée et gère d'autres services et domaines logiques à l'aide de Logical Domains Manager.

D

DHCP Protocole DHCP (Dynamic Host Configuration Protocol).

DIO E/S directes.

DMA Le DMA (Direct Memory Access, Accès direct à la mémoire) est la capacité à transférer directement des données entre la mémoire et un périphérique (par exemple, une carte réseau) sans impliquer la CPU.

DMP Dynamic Multipathing (Veritas).

domaine Voir la section domaine logique.

DPS Logiciel Date Plane Software.

DR Reconfiguration dynamique.

`drd` Démon de reconfiguration dynamique du SE SE Oracle Solaris pour Logical Domains Manager (voir la page de manuel `drd`(1M)).

DRM Gestion dynamique des ressources.

DS Module de services de domaine.

DVD Disque versatile numérique.

E

EFI Interface de microprogramme extensible.

ETM Module de gestion des tables de codage.

F

FC_AL Boucle arbitrée de canal de fibre.

FMA Architecture de gestion des pannes (Fault Management Architecture).

`fmd` Démon du gestionnaire de pannes du SE SE Oracle Solaris (voir la page de manuel `fmd`(1M)).

`fmthard` Commande permettant de remplir les étiquettes sur les disques durs (voir la page de manuel `fmthard`(1M)).

`format` Utilitaire de partitionnement et de maintenance de disque (voir la page de manuel `format`(1M)).

G

domaine invité Utilise les services des domaines d'E/S et de service et est géré par le domaine de contrôle.

GLDv3 Pilote LAN générique version 3.

Go Giga-octet.

H

hyperviseur Couche de microprogramme interposé entre le système d'exploitation et la couche matérielle.

sécurisation Modification de la configuration du SE Oracle Solaris pour améliorer la sécurité.

I

Domaine d'E/S Domaine ayant la propriété directe et l'accès direct aux périphériques d'E/S physiques et qui partage ces périphériques avec d'autres domaines logiques sous la forme de périphériques virtuels.

E/S Périphériques d'E/S, tels que des disques internes et des contrôleurs PCIe, ainsi que les périphériques et adaptateurs qui leur sont associés.

IB Infiniband.

IDE Integrated Drive Electronics (Electronique d'unité de disque intégrée).

IDR Interim Diagnostic Release (Version de diagnostic intermédiaire).

ILOM Integrated Lights Out Manager, système dédié composé de matériel et de logiciels le prenant en charge permettant de gérer un serveur indépendamment du système d'exploitation.

`ioctl` Appel de contrôle d'entrée/sortie.

IPMP Internet Protocol Network Multipathing.

K

`kaio` Entrée/sortie de noyau asynchrone.

Ko Kilo-octet.

KU Mise à jour du noyau.

L

domaine logique Machine virtuelle composée d'un regroupement logique et discret de ressources, qui dispose d'un système d'exploitation et d'une identité qui lui sont propres au sein d'un système informatique unique. Egalement appelé un *domaine*.

LAN Réseau local.

LDAP Protocole LDAP (Lightweight Directory Access Protocol).

LDC Canal de domaine logique.

`ldm` Utilitaire de Logical Domains Manager (voir la page de manuel `ldm`(1M)).

`ldmd` Démon de Logical Domains Manager

`lofi` Fichier loopback.

Logical Domains Manager CLI permettant de créer et de gérer des domaines logiques et d'allouer les ressources aux domaines.

M

MAC Adresse de contrôle d'accès au média, que Logical Domains Manager peut assigner automatiquement ou que vous pouvez assigner manuellement.

MAU Unité arithmétique modulaire.

MD Description de la machine dans la base de données du serveur.

`mem`, `memory` Unité de mémoire - Taille par défaut en octets ou indiquée en giga-octets (`G`), kilo-octets (`K`) ou méga-octets (`M`). Mémoire virtualisée du serveur qui peut être allouée aux domaines invités.

`metadb` Commande permettant de créer et de supprimer les répliques de la base de données d'état des métapériphériques Solaris Volume Manager (voir la page de manuel `metadb`(1M)).

`metaset` Commande permettant de configurer les ensembles de disques (voir la page de manuel `metaset`(1M)).

`mhd` Commande permettant d'effectuer des opérations de contrôle de disque multihôte (voir la page de manuel `mhd`(7i)).

MIB Base MIB (Management Information Base).

MMF Fibre multimode.

MMU Unité de gestion de la mémoire.

Mo Méga-octet.

`mpgroup` Nom du groupe multipathing pour le basculement du disque virtuel.

`mtu` Unité de transmission maximale.

réduction Installation du nombre minimum de packages nécessaires du SE Oracle Solaris principal.

N

NIS Network Information Services (Services d'information réseau).

NIU Unité d'interface réseau (serveurs Sun SPARC Enterprise T5120 et T5220 d'Oracle).

NTS Serveur de terminal réseau.

NVRAM Mémoire à accès aléatoire non volatile.

nxge Pilote pour un adaptateur Ethernet 10 Gb NIU.

O

OID Identificateur d'objet sous forme de séquence numérique identifiant chacun des objets présents dans une base MIB (Management Information Base, base d'informations de gestion) de manière unique.

OVF Format de virtualisation ouvert.

P

domaine physique Ensemble des ressources gérées par un seule instance Oracle VM Server for SPARC. Un domaine physique peut être un système physique complet, comme c'est le cas avec les serveurs SPARC des séries T- et S- pris en charge. Ou il peut également s'agir de l'ensemble du système ou d'un sous-ensemble du système, comme c'est le cas des serveurs SPARC de série M pris en charge.

fonction physique Fonction PCI prenant en charge les fonctionnalités SR-IOV telles que définies dans la norme SR-IOV. Une fonction physique contient la structure des fonctions SR-IOV et est utilisée pour gérer la fonctionnalité SR-IOV. Les fonctions physiques sont des fonctions PCIe à part entière qui peuvent être découvertes, gérées et manipulées comme n'importe quel autre périphérique PCIe. Les fonctions physiques disposent de ressources de configuration complètes et peuvent être utilisées pour configurer ou contrôle le périphérique PCIe.

P2V Outil de conversion physique-à-virtuel Oracle VM Server for SPARC Voir la section Chapitre 19, "Oracle VM Server for SPARC Physical-to-Virtual Conversion Tool" du manuel *Oracle VM Server for SPARC 3.3 Administration Guide*.

PA Adresse physique.

PCI Bus d'interconnexion de composants périphériques.

PCI-X Bus étendu PCI.

PCIe Bus PCI EXPRESS.

pcpu CPU physique.

physio Entrée/sortie physique.

PICL Informations de plate-forme et bibliothèque de contrôle.

picld Démon PICL (voir la page de manuel `picld`(1M)).

PM Gestion de l'alimentation de la mémoire et des CPU virtuelles.

praudit Commande permettant d'imprimer le contenu d'un fichier de piste d'audit (voir la page de manuel `praudit`(1M)).

PRI Priorité.

R

RA Adresse réelle.

RAID Redundant Array of Inexpensive Disks : technologie permettant de regrouper des disques indépendants en une unité logique.

RPC Appel de procédure à distance.

S

contrôleur système (SC) Voir aussi processeur de services.

domaine de service Domaine logique qui fournit des périphériques, notamment des commutateurs virtuels, des connecteurs de consoles virtuelles et des serveurs de disques virtuels aux autres domaines logiques.

HBA SCSI Adaptateur de bus hôte SCSI.

processeur de service (SP) SP, également appelé contrôleur de système (SC), surveille et exécute la machine physique.

SAN Réseau de stockage.

SASL Authentification simple et couche de sécurité.

SAX API simple pour l'analyseur syntaxique XML, qui traverse un document XML. L'analyseur syntaxique SAX est basé sur les événements et utilisé principalement pour la transmission de données en continu.

SCSA Sun Common SCSI Architecture (architecture de Sun Common SCSI).

SMA Agent de gestion du système.

SMF Utilitaire de gestion des services.

SMI Structure of Management Information : structure permettant de définir et de regrouper des objets gérés destinés à être utilisés par la base MIB.

SNMP Protocole SNMP.

SR-IOV Virtualisation d'E/S à root unique.

SSH Secure Shell.

`ssh` Commande Shell sécurisé (voir la page de manuel `ssh`(1)).

`sshd` Démon de Shell sécurisé (voir la page de manuel `sshd`(1M)).

SunVTS Sun Validation Test Suite.

`svcadm` Manipule les instances de service (voir la page de manuel `svcadm`(1M)).

T

TLS Sécurité de la couche de transport.

U

UDP Protocole UDP (User Diagram Protocol).

unicast Communication réseau ayant lieu entre un émetteur unique et un récepteur unique.

`uscsi` Interface de commande SCSI utilisateur (voir la page de manuel `uscsi`(7I)).

UTP Paire torsadée non blindée.

V

fonction virtuelle Fonction PCI associée à une fonction physique. Une fonction virtuelle est une fonction PCIe allégée qui partage une ou plusieurs ressources physiques avec la fonction physique et avec

d'autres fonctions virtuelles associées à la même fonction physique. Les fonctions virtuelles ne peuvent avoir de ressources de configuration que pour leur propre comportement.

var Variable.

VBSC Contrôleur système de lame virtuelle.

vcc, vconscon Service du concentrateur de console virtuelle avec une plage de ports spécifique pour assigner des domaines invités.

vcons, vconsole Console virtuelle pour accéder aux messages au niveau du système. Une connexion est établie en connectant le service `vconscon` dans le domaine de contrôle à un port spécifique.

vcpu Unité de calcul centrale virtuelle Chaque coeur dans un serveur est représenté par une CPU virtuelle.

vdc Client de disque virtuel.

vdisk Un disque virtuel est un périphérique en mode bloc générique associé à différents types de périphériques physiques, de volumes ou de fichiers.

vds, vdiskserver Le serveur de disque virtuel vous permet d'importer des disques virtuels dans un domaine logique.

vdsdev, vdiskserverdevice Le périphérique de serveur de disque virtuel est exporté par le serveur de disque virtuel. Le périphérique peut être un disque entier, une tranche sur le disque, un fichier ou un volume de disque.

vldc Service de canal de domaine logique virtuel.

vldcc Client de canal de domaine logique virtuel.

vnet Le périphérique réseau virtuel implémente un périphérique Ethernet virtuel et communique avec les autres périphériques `vnet` du système à l'aide du commutateur réseau virtuel (`vswitch`)

VNIC Carte d'interface réseau virtuelle, c'est-à-dire instanciation virtuelle d'un périphérique réseau physique pouvant être créée à partir du périphérique réseau physique et assignée à une zone.

vNTS Service de terminal réseau virtuel.

vntsd Démon du serveur de terminal réseau virtuel du SE SE Oracle Solaris pour les consoles de domaine (voir la page de manuel `vntsd`(1M)).

volfs Système de fichiers de gestion du volume (voir la page de manuel `volfs`(7FS)).

vsw, vswitch Commutateur de réseau virtuel qui connecte les périphériques du réseau virtuel au réseau externe et commute également les paquets entre eux.

VTOC Table des matières du volume.

VxDMP Veritas Dynamic Mutipathing.

VxVM Veritas Volume Manager.

X

XFP eXtreme Fast Path.

XML Extensible Markup Language.

XMPP Extensible Messaging and Presence Protocol.

Z

ZFS Système de fichiers Zettabyte.

`zpool` Pool de stockage ZFS (voir la page de manuel `zpool`(1M)).

ZVOL Pilote d'émulation de volume ZFS.

Index

B

C

D

E

F

J

L

M

N

S

T

U

V

Z