Guide d'administration d'Oracle® VM Server for SPARC 2.2

Référence : E29607
Mai 2012

ORACLE

120709@25097

Table des matières

Préface

Le *Guide d'administration d'Oracle VM Server for SPARC .2.2* fournit des informations et des procédures détaillées qui décrivent la présentation, les considérations relatives à la sécurité, l'installation, la configuration, la modification et l'exécution des tâches communes pour le logiciel Oracle VM Server for SPARC .2.2 sur les serveurs, lames et modules serveur pris en charge. Reportez-vous à la section "Plates-formes prises en charge" du manuel *Notes de version d'Oracle VM Server for SPARC 2.2*.

Ce guide est destiné aux administrateurs système de ces serveurs qui ont des connaissances des systèmes UNIX et du système d'exploitation Oracle Solaris (SE Oracle Solaris).

Documentation connexe

Le tableau suivant présente la documentation disponible pour la version Oracle VM Server for SPARC .2.2. Ces documents sont disponibles aux formats HTML et PDF, sauf mention contraire.

TABLEAU P–1 Documentation connexe

Application	Titre
Logiciel Oracle VM Server for SPARC .2.2	*Guide d'administration d'Oracle VM Server for SPARC 2.2* *Guide de sécurité d'Oracle VM Server for SPARC 2.2* *Oracle VM Server for SPARC 2.2 Reference Manual* *Notes de version d'Oracle VM Server for SPARC 2.2*
Pages de manuel `drd`(1M) et `vntsd`(1M) d'Oracle VM Server for SPARC .2.2	Manuels de référence du SE Oracle Solaris : ■ Documentation Oracle Solaris 10 ■ Documentation Oracle Solaris 11
SE Oracle Solaris : installation et configuration	Guides relatifs à l'installation et la configuration du SE Oracle Solaris : ■ Documentation Oracle Solaris 10 ■ Documentation Oracle Solaris 11

TABLEAU P–1 Documentation connexe *(Suite)*

Application	Titre
Sécurité d'Oracle VM Server for SPARC et du SE Oracle Solaris	Livre blanc d'Oracle VM Server for SPARC et guides de sécurité du SE Oracle Solaris : ■ *Secure Deployment of Oracle VM Server for SPARC* (`http://www.oracle.com/technetwork/articles/systems-hardware-architecture/secure-ovm-sparc-deployment-294062.pdf`) ■ *Instrucciones de seguridad de Oracle Solaris 10* ■ *Directrices de seguridad de Oracle Solaris 11*

Vous trouverez des documents relatifs à votre serveur, votre logiciel ou au SE Oracle Solaris à l'adresse `http://www.oracle.com/technetwork/indexes/documentation/index.html`. Utilisez la zone de recherche pour rechercher les documents et les informations dont vous avez besoin.

Accéder au support Oracle

Les clients d'Oracle peuvent accéder à un 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 `http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs` si vous êtes malentendant.

Conventions typographiques

Le tableau suivant décrit les conventions typographiques utilisées dans ce manuel.

TABLEAU P–2 Conventions typographiques

Police de caractère	Signification	Exemple
`AaBbCc123`	Noms de commandes, de fichiers, de répertoires et informations affichées à l'écran.	Modifiez le fichier `.login`. Utilisez `ls -a` pour dresser la liste des fichiers. `machine_name% you have mail.`
`AaBbCc123`	Ce que vous tapez, mis en évidence par rapport aux informations affichées à l'écran.	`machine_name%` **`su`** `Mot de passe :`
aabbcc123	Paramètre substituable : à remplacer par une valeur ou un nom réel	La commande permettant de supprimer un fichier est `rm` *nomfichier*.

TABLEAU P–2 Conventions typographiques *(Suite)*

Police de caractère	Signification	Exemple
AaBbCc123	Titres de livre, nouveaux termes et termes à souligner	Lisez le chapitre 6 du *Guide de l'utilisateur*. Un *cache* est une copie stockée localement. N'enregistrez *pas* le fichier . **Remarque :** certains éléments mis en évidence apparaissent en caractères gras.

Invites de shell dans les exemples de commande

Le tableau suivant présente l'invite du système UNIX par défaut et l'invite du superutilisateur pour les shells inclus dans le SE Oracle Solaris. Notez que l'invite du système par défaut qui est affiché dans les exemples de commande varie en fonction de la version d'Oracle Solaris.

TABLEAU P–3 Invites de shell

Shell	Invite
Shell Bash, shell Korn et shell Bourne	`$`
Shell Bash, shell Korn et shell Bourne pour un superutilisateur	`#`
Shell C	`machine_name%`
Shell C pour un superutilisateur	`machine_name#`

PARTIE I

Logiciel Oracle VM Server for SPARC .2.2

Cette section présente le logiciel Oracle VM Server for SPARC .2.2 qui offre des capacités de visualisation de classe entreprise très efficaces pour les serveurs SPARC de série T d'Oracle.

◆ ◆ ◆ CHAPITRE 1

1 Présentation du logiciel Oracle VM Server for SPARC

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

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 "SE Oracle Solaris requis et recommandé" du manuel *Notes de version d'Oracle VM Server for SPARC 2.2*.

Oracle VM Server for SPARC offre des capacités de visualisation de classe entreprise très efficaces pour les serveurs SPARC de série T d'Oracle. 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 de série T et le SE Oracle Solaris.

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 10 dans le domaine `primary`, vous pouvez tout de même exécuter le SE Oracle Solaris 11 dans les domaines invités. De même, si vous exécutez le SE Oracle Solaris 11 dans le domaine `primary`, vous pouvez tout de même exécuter le SE Oracle Solaris 10 dans les domaines invités.

La seule différence entre l'exécution du SE Oracle Solaris 10 et celle du SE Oracle Solaris 11 dans le domaine `primary` tient aux différences de fonctionnalités entre les deux systèmes d'exploitation.

Ce chapitre aborde les sujets suivants :

Hyperviseur et Logical Domains

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

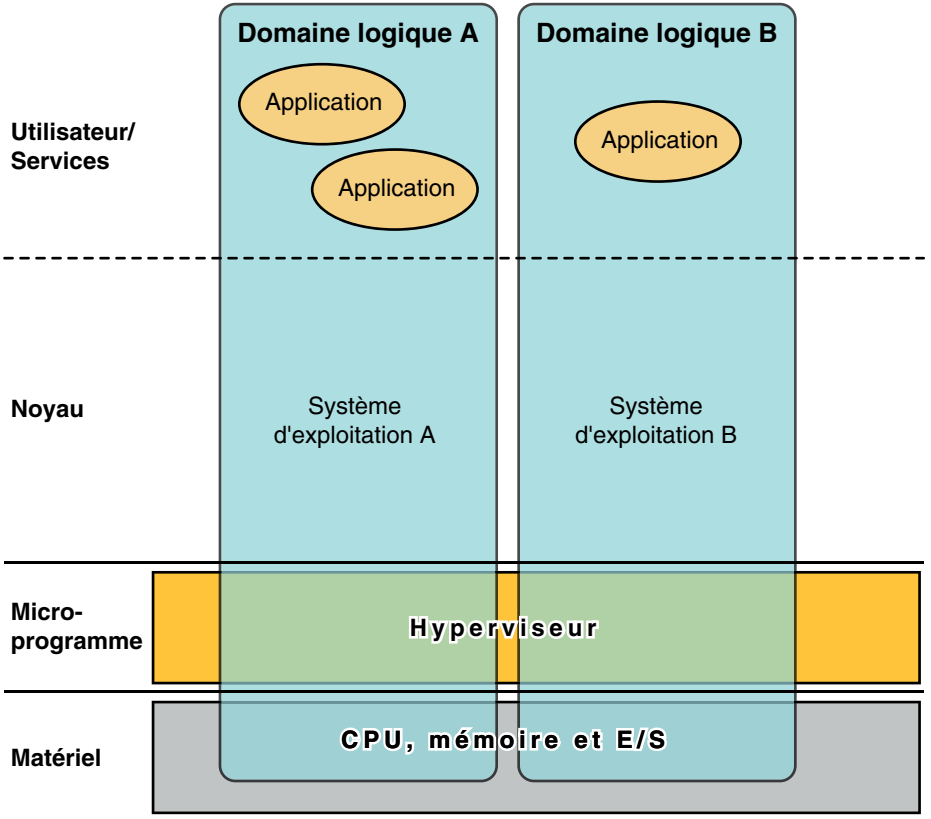
Le SPARC *hyperviseur* est une petite couche de microprogramme qui fournit une architecture de machine virtualisée stable sur laquelle on peut installer un système d'exploitation. Les serveurs Sun d'Oracle utilisant cet hyperviseur fournissent des fonctions matérielles qui prennent en charge le contrôle 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 avoir à exécuter un 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 de Logical Domains :

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

FIGURE 1–1 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. Cela 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 redémarré indépendamment des autres sans nécessiter une remise sous tension 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.

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. Vous trouverez ci-dessous les rôles que les domaines logiques peuvent avoir :

- **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. Vous ne pouvez 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. N'importe quel domaine peut être configuré en tant que 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 un complexe racine PCIe ou il peut posséder un emplacement PCIe ou un périphérique PCIe intégré à l'aide de la fonction d'E/S directes (DIO). Reportez-vous à la section “Assignation des périphériques d'extrémité PCIe” à la page 86.

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 racine.** Un domaine racine a un complexe racine 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 racine est également un domaine d'E/S, car il possède et a un accès direct aux périphériques d'E/S.

  Le nombre de domaines racine que vous pouvez avoir dépend de l'architecture de votre plate-forme. Par exemple, si vous utilisez un serveur Sun SPARC Enterprise T5440, vous pouvez avoir jusqu'à quatre domaines racine.
- **Domaine invité.** Un domaine invité est un domaine non E/S qui consomme des services de périphérique virtuel fournis par un ou plusieurs domaines. Un domaine invité n'a pas de périphérique d'E/S physiques, mais a uniquement des périphériques d'E/S virtuels, notamment 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 Logical Domains. 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. Vous effectuez cette opération 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 encours d'exécution pour utiliser la CLI de Logical Domains Manager.

Entrée/Sortie virtuelle

Dans un environnement Logical Domains, vous pouvez mettre à disposition jusqu'à 128 domaines sur un système SPARC de série T. Certains serveurs SPARC de série T, 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. Reportez-vous

au Chapitre 6, “Configuration des domaines d'E/S”. Cette limitation sur le nombre de périphériques d'E/S physiques à accès direct est traitée en implémentant un modèle d'E/S virtualisé.

Les domaines logiques n'ayant pas d'accès E/S physique sont configurés avec des périphériques d'E/S virtuelsCommuniquant 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

Logical Domains 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 :

- Client de disque virtuel (`vdc`) qui exporte l'interface de périphérique de bloc
- 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 Logical Domains, 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 du concentrateur de console virtuelle fonctionne comme un concentrateur pour tout le trafic de la console du domaine et communique avec le démon du serveur de terminal réseau virtuel (`vntsd`) pour fournir l'accès à chaque console via le 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 le redémarrage du domaine de contrôle. Pour plus d'informations, reportez-vous à la section "Reconfiguration des ressources" à la page 207.

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. Reportez-vous à la page de manuel `ldm(1M)`. Vous pouvez également indiquer une configuration pour démarrer à partir du SP. Reportez-vous à la section "Utilisation de Logical Domains avec le processeur de service" à la page 264.

Pour plus d'informations sur la gestion des configurations, reportez-vous à la section "Gestion des configurations de Logical Domains" à la page 255.

Outil de conversion physique-à-virtuel Oracle VM Server for SPARC

L'outil de conversion physique-à-virtuel (P2V) Oracle VM Server for SPARC convertit automatiquement un système physique existant en système virtuel exécutant le SE Oracle Solaris 10 dans un domaine logique sur un système chip multithreading (CMT). Vous pouvez exécuter la commande `ldmp2v` à partir d'un domaine de contrôle qui exécute le SE Oracle Solaris 10 ou le SE Oracle Solaris 11 pour convertir l'un des systèmes source suivants en un domaine logique :

- Un système `sun4u` SPARC qui exécute au moins le SE Solaris 8, Solaris 9 ou Oracle Solaris 10.
- Un système `sun4v` qui exécute le SE Oracle Solaris 10, mais n'exécute pas le logiciel Oracle VM Server for SPARC

Remarque – Vous *ne pouvez pas* utiliser l'outil P2V pour convertir un système physique Oracle Solaris 11 en un système virtuel.

Pour plus d'informations sur l'outil et sur son installation, reportez-vous au Chapitre 13, “Outil de conversion physique-à-virtuel Oracle VM Server for SPARC”. Pour plus d'informations sur la commande `ldmp2v`, reportez-vous à la page de manuel `ldmp2v(1M)`.

Assistant de configuration d'Oracle VM Server for SPARC

L'assistant de configuration Oracle VM Server for SPARC vous guide tout au long de la configuration d'un domaine logique en paramétrant les propriétés de base. Il peut être utilisé pour configurer un système sur lequel le logiciel Oracle VM Server for SPARC est installé, mais pas encore configuré.

Après avoir rassemblé les données de configuration, l'assistant de configuration crée une configuration adaptée à l'initialisation en tant que domaine logique. Vous pouvez également utiliser les valeurs par défaut sélectionnées par l'assistant de configuration afin de créer une configuration système utilisable.

Remarque – La commande `ldmconfig` est uniquement prise en charge sur les systèmes Oracle Solaris 10.

L'assistant de configuration est un outil terminal.

Pour plus d'informations, reportez-vous au Chapitre 14, “Assistant de configuration d'Oracle VM Server for SPARC (Oracle Solaris 10)” et à la page de manuel `ldmconfig(1M)` man page.

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

Oracle VM Server for SPARC Management Information Base (MIB) permet aux applications tierces de gestion de système de contrôler 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 Chapitre 15, “Utilisation du logiciel MIB (Management Information Base) Oracle VM Server for SPARC”.

Remarque – Le logiciel Oracle VM Server for SPARC MIB peut uniquement être utilisé sur un système Oracle Solaris 10.

◆ ◆ ◆ CHAPITRE 2

Installation et activation du logiciel

Ce chapitre décrit la procédure d'installation ou de mise à niveau des différents composants logiciels requis pour activer le logiciel Oracle VM Server for SPARC .2.2. L'utilisation du logiciel Oracle VM Server for SPARC nécessite les composants suivants :

- Une plate-forme prise en charge ; reportez-vous à la section "Plates-formes prises en charge" du manuel *Notes de version d'Oracle VM Server for SPARC 2.2* pour obtenir la liste des plates-formes prises en charge.
- Un domaine de contrôle exécutant un système d'exploitation au moins équivalent au SE Oracle Solaris 11 et, le cas échéant, la SRU (mise à jour du référentiel support) appropriée, ou le SE Oracle Solaris 10 8/11 avec tous les patchs recommandés à la section "Logiciels et patches requis" du manuel *Notes de version d'Oracle VM Server for SPARC 2.2*. Reportez-vous à la section "Mise à niveau du SE Oracle Solaris" à la page 36.
- La version 7.4.2 au minimum du microprogramme système pour votre plate-forme Sun UltraSPARC T2 ou T2 Plus, ou la version 8.2.0 pour vos plates-formes SPARC T3 et SPARC T4. Reportez-vous à la section "Mise à niveau du microprogramme système" à la page 31.
- Logiciel Oracle VM Server for SPARC .2.2 installé et activé sur le domaine de contrôle. Reportez-vous à la section "Installation de Logical Domains Manager" à la page 32.
- (Facultatif) Le package logiciel Oracle VM Server for SPARC Management Information Base (MIB), uniquement disponible dans le SE Oracle Solaris 10. Reportez-vous au Chapitre 15, "Utilisation du logiciel MIB (Management Information Base) Oracle VM Server for SPARC".

Le SE Oracle Solaris et le microprogramme système doivent être installés ou mis à niveau sur votre serveur avant d'installer ou de mettre à niveau Logical Domains Manager. Si votre système utilise déjà le logiciel Oracle VM Server for SPARC, reportez-vous à la section "Mise à niveau d'un système utilisant déjà Oracle VM Server for SPARC" à la page 36. Sinon, reportez-vous à la section "Installation du logiciel Oracle VM Server for SPARC sur un nouveau système" à la page 30.

Ce chapitre aborde les sujets suivants :

Installation du logiciel Oracle VM Server for SPARC sur un nouveau système

Les plates-formes Sun d'Oracle qui prennent en charge le logiciel Oracle VM Server for SPARC sont fournies avec le SE Oracle Solaris 10 ou le SE Oracle Solaris 11. Initialement, la plate-forme apparaît comme un seul système hébergeant uniquement un système d'exploitation. Après l'installation du SE Oracle Solaris, du microprogramme système et de Logical Domains Manager, le système et l'instance d'origine du SE Oracle Solaris deviennent le domaine de contrôle. Ce premier domaine de la plate-forme est nommé `primary` et vous ne pouvez pas modifier ce nom ou détruire ce domaine. A partir de ce moment, la plate-forme peut être reconfigurée pour avoir plusieurs domaines hébergeant différentes instances du SE Oracle Solaris.

Remarque – La version du logiciel 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 10 dans le domaine `primary`, vous pouvez tout de même exécuter le SE Oracle Solaris 11 dans les domaines invités. De même, si vous exécutez le SE Oracle Solaris 11 dans le domaine `primary`, vous pouvez tout de même exécuter le SE Oracle Solaris 10 dans les domaines invités.

Décidez quelle version du SE Oracle Solaris exécuter sur le domaine `primary` en fonction de vos besoins et des éventuelles différences de fonctionnalités entre Oracle Solaris 10 et Oracle Solaris 11. Reportez-vous aux manuels *Oracle Solaris 11 Release Notes* et *Transitioning From Oracle Solaris 10 to Oracle Solaris 11* .

Mise à jour du SE Oracle Solaris

Sur un nouveau système, il peut s'avérer utile de réinstaller le système d'exploitation d'origine, conformément à votre stratégie d'installation. Reportez-vous à la section "SE Oracle Solaris requis et recommandé" du manuel *Notes de version d'Oracle VM Server for SPARC 2.2*. Pour obtenir les instructions complètes d'installation du SE Oracle Solaris, reportez-vous à Oracle Solaris 10 8/11 Information Library (`http://docs.oracle.com/cd/E23823_01/`) et à Oracle Solaris 11 Information Library (`http://docs.oracle.com/cd/E23824_01/`). Vous pouvez personnaliser votre installation en fonction des besoins de votre système.

Si votre système est déjà installé sur le SE Oracle Solaris, vous devez le mettre à niveau par rapport à la version du SE associée au logiciel Oracle VM Server for SPARC .2.2. Reportez-vous à la section "Logiciels et patches requis" du manuel *Notes de version d'Oracle VM Server for SPARC 2.2*. Pour obtenir les instructions complètes de mise à niveau du SE Oracle Solaris, reportez-vous à Oracle Solaris 10 8/11 Information Library (`http://docs.oracle.com/cd/E23823_01/`) et à Oracle Solaris 11 Information Library (`http://docs.oracle.com/cd/E23824_01/`).

Mise à niveau du microprogramme système

Les tâches suivantes décrivent comment mettre à niveau le microprogramme système à l'aide du logiciel ILOM (Integrated Lights Out Manager). Reportez-vous également aux ressources suivantes :

- Pour plus d'informations sur la mise à niveau du microprogramme système à l'aide du logiciel ILOM, reportez-vous aux sections "Mise à jour du microprogramme" du manuel `http://www.oracle.com/technetwork/documentation/sparc-tseries-servers-252697.html` et "Mise à jour du microprogramme ILOM" du manuel *Guide des procédures relatives à la CLI de Sun Integrated Lights Out Manager (ILOM) 3.0*.
- Vous pouvez trouver le microprogramme du système pour votre plate-forme à l'adresse `http://www.oracle.com/technetwork/systems/patches/firmware/index.html`.
- Pour plus d'informations sur le microprogramme système requis pour les serveurs pris en charge, reportez-vous à la section "Patches de microprogramme système requis et recommandés" du manuel *Notes de version d'Oracle VM Server for SPARC 2.2*.
- Pour mettre à niveau le microprogramme du système à partir du domaine de contrôle, reportez-vous aux notes de version du microprogramme du système.
- Reportez-vous aux guides d'administration ou aux notes sur le produit pour les serveurs pris en charge pour plus d'informations sur l'installation et la mise à niveau du microprogramme du système pour ces serveurs.
- Vous pouvez également utiliser l'interface Web d'ILOM pour mettre à niveau un microprogramme système. Reportez-vous à la section "Mise à jour du microprogramme ILOM" du manuel *Guide des procédures relatives à l'interface Web de Sun Integrated Lights Out Manager (ILOM) 3.0*.

Téléchargement de Logical Domains Manager

Vous pouvez obtenir les derniers packages des SE Oracle Solaris 10 et Oracle Solaris 11. Notez que le logiciel Oracle VM Server for SPARC est inclus par défaut dans le système d'exploitation Oracle Solaris 11.

- **SE Oracle Solaris 10.** Téléchargez le package `OVM_Server_SPARC-2_2.zip` à partir de My Oracle support. Reportez-vous à "Procédure de téléchargement du logiciel de Logical Domains Manager (Oracle Solaris 10)" à la page 32.
- **SE Oracle Solaris 11.** Obtenez le package `ldomsmanager` à partir du référentiel support d'Oracle Solaris 11. Reportez-vous à "Procédure de mise à niveau vers le logiciel Oracle VM Server for SPARC .2.2 (Oracle Solaris 11)" à la page 40.

▼ Procédure de téléchargement du logiciel de Logical Domains Manager (Oracle Solaris 10)

1 Téléchargez le fichier zip (`OVM_Server_SPARC-2_2.zip`).

Vous trouverez le logiciel sur la page Web `http://www.oracle.com/virtualization/index.html`.

2 Décompressez le fichier zip.

```
$ unzip OVM_Server_SPARC-2_2.zip
```

Pour plus d'informations sur la structure du fichier et son contenu, reportez-vous à la section "Emplacement du logiciel Oracle VM Server for SPARC .2.2" du manuel *Notes de version d'Oracle VM Server for SPARC 2.2*.

Installation de Logical Domains Manager

Les méthodes d'installation du logiciel de Logical Domains Manager sont énumérées ci-dessous :

- **Oracle Solaris 10 uniquement.** Utilisation du script d'installation pour installer les packages et les patchs. Cette opération installe automatiquement le logiciel de Logical Domains Manager. Reportez-vous à "(Oracle Solaris 10) Installation automatique du logiciel de Logical Domains Manager" à la page 33.
- **Oracle Solaris 10 uniquement.** Utilisation de la fonction JumpStart d'Oracle Solaris pour installer les packages dans le cadre d'une installation en réseau. Pour plus d'informations sur la configuration d'un serveur JumpStart, reportez-vous au manuel *Guía de instalación de Oracle Solaris 10 8/11: instalaciones JumpStart personalizadas y avanzadas*. Reportez-vous également à *Technologie JumpStart : utilisation dans l'environnement d'exploitation Solaris* pour obtenir des informations complètes sur l'utilisation de cette fonction.

- **Oracle Solaris 11 uniquement.** Utilisation du programme d'installation automatisée d'Oracle Solaris 11 pour installer les packages dans le cadre d'une installation en réseau. Reportez-vous à la section "¿Cómo se utiliza Automated Installer?" du manuel *Instalación de sistemas Oracle Solaris 11* et à la section *Transición de Oracle Solaris 10 JumpStart a Oracle Solaris 11 Automated Installer*.
- Installation manuelle du package. Reportez-vous à la section "Installation manuelle du logiciel de Logical Domains Manager" à la page 34.

Remarque – Gardez à l'esprit que vous devez installer manuellement les packages de logiciels Oracle VM Server for SPARC MIB après l'installation des packages Oracle VM Server for SPARC. Il n'est pas installé automatiquement avec les autres packages. Pour plus d'informations sur l'installation et l'utilisation de Oracle VM Server for SPARC MIB, reportez-vous au Chapitre 15, "Utilisation du logiciel MIB (Management Information Base) Oracle VM Server for SPARC".

(Oracle Solaris 10) Installation automatique du logiciel de Logical Domains Manager

Si vous utilisez le script d'installation `install-ldm`, vous avez plusieurs choix pour indiquer comment vous souhaitez exécuter le script. Chaque choix est décrit dans les procédures suivantes.

- **L'utilisation du script `install-ldm` sans aucune option n'effectue pas les opérations suivantes automatiquement :**
  - Vérifie que la version du SE Oracle Solaris est le SE Oracle Solaris 10
  - Vérification que les sous-répertoires du package `SUNWldm/` et `SUNWldmp2v/` sont présents
  - Vérification que les packages du pilote de Logical Domains prérequis, `SUNWldomr` et `SUNWldomu`, sont présents
  - Vérification que les packages `SUNWldm` et `SUNWldmp2v` n'ont pas été installés
  - Installation du logiciel Oracle VM Server for SPARC .2.2
  - Vérification que tous les packages sont installés
  - Si le SST `SUNWjass` est déjà installé, vous êtes invité à renforcer le SE Oracle Solaris sur le domaine de contrôle.
  - Déterminer s'il faut utiliser l'assistant de configuration Oracle VM Server for SPARC (`ldmconfig`) pour effectuer l'installation.
- **L'utilisation du script `install-ldm` avec l'option `-c` exécute automatiquement l'assistant de configuration d'Oracle VM Server for SPARC après l'installation du logiciel.**
- **L'utilisation du script `install-ldm` avec l'option `-s` passe l'exécution de l'assistant de configuration Oracle VM Server for SPARC.**

- **L'utilisation du script `install-ldm` et les options suivantes avec les logiciel SST vous permettent d'effectuer les opérations suivantes :**
  - `install-ldm -d`. Vous permet d'indiquer un pilote SST différent d'un pilote se terminant par `-secure.driver`. Cette option effectue automatiquement toutes les fonctions répertoriées dans le choix précédent et renforce le SE Oracle Solaris sur le domaine de contrôle avec le pilote personnalisé SST que vous indiquez. Par exemple, `server-secure-myname.driver`.
  - `install-ldm -d none`. Indique que vous *ne* voulez pas renforcer le SE Oracle Solaris s'exécutant sur votre domaine de contrôle à l'aide de SST. Cette option effectue automatiquement toutes les fonctions sauf le renforcement répertorié dans les choix précédents. Ne pas utiliser le SST n'est pas recommandé et doit uniquement être effectué lorsque vous avez l'intention de renforcer votre domaine de contrôle à l'aide d'un autre processus.
  - `install-ldm -p`. Indique que vous voulez uniquement effectuer des actions de post-installation pour activer le démon de Logical Domains Manager (`ldmd`) et exécuter le SST. Par exemple, vous devez utiliser cette option si les packages `SUNWldm` et `SUNWjass` sont préinstallés sur votre serveur.

Installation manuelle du logiciel de Logical Domains Manager

La procédure suivante vous guide tout au long de l'installation manuelle du logiciel Oracle VM Server for SPARC .2.2 sur le SE Oracle Solaris 10.

Lors de l'installation du système d'exploitation Oracle Solaris 11, le logiciel Oracle VM Server for SPARC 2.1 est installé par défaut. Si vous souhaitez installer le logiciel Oracle VM Server for SPARC 2.2, reportez-vous à la section "Procédure de mise à niveau vers le logiciel Oracle VM Server for SPARC .2.2 (Oracle Solaris 11)" à la page 40.

▼ Procédure d'installation manuelle du logiciel Oracle VM Server for SPARC .2.2 (Oracle Solaris 10)

Avant de commencer

Téléchargez le logiciel Oracle VM Server for SPARC .2.2 (les packages `SUNWldm` et `SUNWldmp2v`). Pour obtenir des instructions, reportez-vous à la section "Procédure de téléchargement du logiciel de Logical Domains Manager (Oracle Solaris 10)" à la page 32.

1 (Facultatif) Si nécessaire, enregistrez votre configuration sur le processeur de service (SP).

Effectuez uniquement cette opération si vous exécutez une version antérieure du logiciel Oracle VM Server for SPARC.

```
primary# ldm add-config config-name
```

2 Installez les packages `SUNWldm.v` et `SUNWldmp2v`.

```
# pkgadd -Gd . SUNWldm.v SUNWldmp2v
```

Répondez y pour oui à toutes les questions des invites interactives.

L'option `-G` installe le package dans la zone globale uniquement. L'option `-d` définit le chemin vers le répertoire contenant les packages `SUNWldm.v` et `SUNWldmp2v`.

Pour plus d'informations sur la commande `pkgadd`, reportez-vous à la page de manuel `pkgadd`(1M).

3 Vérifiez que les packages `SUNWldm` et `SUNWldmp2v` sont installés.

Les informations de révision (`REV`) sont données à titre d'exemple :

```
# pkginfo -l SUNWldm | grep VERSION
VERSION=2.2,REV=2012.02.01.10.20
```

Pour plus d'informations sur la commande `pkginfo`, reportez-vous à la page de manuel `pkginfo`(1).

Activation du démon de Logical Domains Manager

Le script d'installation `install-ldm` active automatiquement le démon de Logical Domains Manager (`ldmd`). Le démon `ldmd` est également automatiquement activé lorsque le package logiciel Oracle VM Server for SPARC est installé. Une fois le démon activé, vous pouvez créer, modifier et contrôler les domaines logiques.

▼ 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`.

Pour plus d'informations sur la commande `svcadm`, reportez-vous à la page de manuel `svcadm`(1M).

```
# svcadm enable ldmd
```

2 Utilisez la commande `ldm list` pour vérifier 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.

```
# /opt/SUNWldm/bin/ldm list
NAME             STATE    FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active   ---c-   SP      64    3264M    0.3%  19d 9m
```

Mise à niveau d'un système utilisant déjà Oracle VM Server for SPARC

Cette section décrit le processus de mise à niveau du microprogramme du SE Oracle Solaris et les composants de Logical Domains Manager sur un système utilisant déjà le logiciel Oracle VM Server for SPARC.

Si votre système est déjà configuré avec le logiciel Oracle VM Server for SPARC, mettez à niveau le domaine de contrôle et les domaines existants. Cette mise à niveau permet l'utilisation de toutes les fonctionnalités Oracle VM Server for SPARC .2.2 sur ces domaines.

Remarque – Avant de mettre à niveau le logiciel Oracle VM Server for SPARC, procédez comme suit :

- Mettez à niveau le système avec le microprogramme système requis.

  Reportez-vous à la section "Logiciels requis pour activer les fonctionnalités d'Oracle VM Server for SPARC .2.2." du manuel *Notes de version d'Oracle VM Server for SPARC 2.2*.
- Appliquez les patchs du SE Oracle Solaris 10 ou la SRU Oracle Solaris 11 requis.

  Reportez-vous à la section "SE Oracle Solaris requis et recommandé" du manuel *Notes de version d'Oracle VM Server for SPARC 2.2*.
- Enregistrez les configurations sur le processeur de service.

Mise à niveau du SE Oracle Solaris

Reportez-vous à la section "Logiciels et patches requis" du manuel *Notes de version d'Oracle VM Server for SPARC 2.2* pour identifier le SE Oracle Solaris 10 ou Oracle Solaris 11 à utiliser avec cette version du logiciel Oracle VM Server for SPARC ainsi que les patchs nécessaires et recommandés pour les différents domaines. Reportez-vous aux guides d'installation d'Oracle Solaris 10 et d'Oracle Solaris 11 pour obtenir des instructions complètes sur la mise à niveau du SE Oracle Solaris.

Lors de la réinstallation du SE Oracle Solaris dans le domaine de contrôle, vous devez enregistrer et restaurer les données de configuration sauvegardées automatiquement Logical Domains et le fichier de la base de données de contraintes, comme décrit dans cette section.

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. A chaque fois que vous réinstallez le système d'exploitation sur le domaine de contrôle, vous devez

sauvegarder et restaurer les données de configuration enregistrées automatiquement de Logical Domains, qui se trouvent dans les répertoires `/var/opt/SUNWldm/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 Logical Domains" à la page 255.

▼ Procédure d'enregistrement et de restauration des répertoires de sauvegarde automatique

Cette procédure montre comment sauvegarder et restaurer les répertoires enregistrés automatiquement.

1. **Sauvegardez les répertoires enregistrés automatiquement.**

```
# cd /
# tar -cvpf autosave.tar var/opt/SUNWldm/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 de corrompre 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/opt/SUNWldm/autosave-*
```

3. **Restaurez les répertoires enregistrés automatiquement.**

   Ces commandes restaurent les fichiers et les répertoires du répertoire `/var/opt/SUNWldm`.

```
# cd /
# tar -xvpf autosave.tar
```

Sauvegarde et restauration du fichier de la base de données de contraintes de Logical Domains

A chaque fois que vous mettez à niveau le système d'exploitation sur le domaine de contrôle, vous devez sauvegarder et restaurer le fichier de la base de données de contraintes de Logical Domains disponible dans `/var/opt/SUNWldm/ldom-db.xml`.

Remarque – Sauvegardez et restaurez également le fichier `/var/opt/SUNWldm/ldom-db.xml` lorsque vous effectuez une autre opération qui est destructive pour les données du fichier du domaine de contrôle, notamment un swap de disque.

Conservation du fichier de la base de données de contraintes de Logical Domains lors de l'utilisation de la fonction Live Upgrade d'Oracle Solaris 10.

Si vous utilisez la fonction Live Upgrade d'Oracle Solaris 10 sur le domaine de contrôle, envisagez d'ajouter la ligne suivante au fichier `/etc/lu/synclist` :

```
/var/opt/SUNWldm/ldom-db.xml     OVERWRITE
```

Cette ligne copie automatiquement la base de données de l'environnement d'initialisation actif vers le nouvel environnement d'initialisation lorsque vous changez d'environnement d'initialisation. Pour plus d'informations sur `/etc/lu/synclist` et la synchronisation des fichiers entre les environnements de démarrage, reportez-vous à la section "Sincronización de archivos entre entornos de inicio" du manuel *Guía de instalación de Oracle Solaris 10 8/11: Solaris Live Upgrade y planificación de la actualización*.

Mise à niveau de Logical Domains Manager et du microprogramme du système

Cette section décrit comment mettre à niveau le logiciel Oracle VM Server for SPARC .2.2.

Téléchargez d'abord Logical Domains Manager sur le domaine de contrôle. Reportez-vous à la section "Téléchargement de Logical Domains Manager" à la page 32.

Arrêtez ensuite tous les domaines (sauf le domaine de contrôle) s'exécutant sur la plate-forme :

▼ Procédure d'arrêt de tous les domaines s'exécutant sur la plate-forme, à l'exception du domaine de contrôle

Effectuez uniquement à cette tâche si vous comptez procéder à un cycle d'alimentation du système ou mettre à niveau le microprogramme. Elle n'est pas nécessaire si vous mettez uniquement à niveau le logiciel Logical Domains Manager.

1. **Affichez chaque domaine à l'invite `ok`.**

2. **Arrêtez tous les domaines à l'aide de l'option `-a`.**

   ```
   primary# ldm stop-domain -a
   ```

3. **Emettez la sous-commande `unbind-domain` à partir du domaine de contrôle pour chaque domaine.**

   ```
   primary# ldm unbind-domain ldom
   ```

Mise à niveau vers le logiciel Oracle VM Server for SPARC .2.2

Cette section illustre le processus de mise à niveau vers le logiciel Oracle VM Server for SPARC .2.2.

▼ Procédure de mise à niveau vers le logiciel Oracle VM Server for SPARC .2.2 (Oracle Solaris 10)

1. **Effectuez une mise à jour flash du microprogramme du système.**

2. **Désactivez le démon de Logical Domains Manager (`ldmd`).**

   ```
   # svcadm disable ldmd
   ```

3. **Supprimez le package `SUNWldm`.**

   ```
   # pkgrm SUNWldm
   ```

4. **Ajoutez le nouveau package `SUNWldm`.**

   ```
   # pkgadd -Gd . SUNWldm.v
   ```

   L'utilisation de l'option `-d` indique que le package se trouve dans le répertoire courant.

5. **Utilisez la commande `ldm` list pour vérifier que Logical Domains Manager est en cours d'exécution.**

   La commande `ldm listdoit` 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      32    3264M    0.3%  19d 9m
   ```

▼ Procédure de mise à niveau vers le logiciel Oracle VM Server for SPARC .2.2 (Oracle Solaris 11)

1 Préparez votre domaine pour une mise à niveau de Logical Domains Manager.

La procédure décrite ci-après autorise la restauration d'un environnement d'initialisation exécutant le logiciel Oracle VM Server for SPARC 2.1 en cas de besoin.

a. Enregistrez la configuration de votre système sur le processeur de service.

```
# ldm add-config config-name
```

L'exemple suivant enregistre la configuration appelée `ldoms-2.1-config` :

```
# ldm add-config ldoms-2.1-config
```

b. Créez un instantané de l'environnement d'initialisation existant.

```
# beadm create snapshot-name
```

L'exemple suivant crée un instantané nommé `S10811@ldoms-2.1-backup` :

```
# beadm create S10811@ldoms-2.1-backup
```

c. Créez un environnement d'initialisation de sauvegarde à partir de l'instantané.

```
# beadm create -e snapshot-name BE-name
```

L'exemple suivant crée un nouvel environnement d'initialisation `ldoms-2.1-backup` à partir de l'instantané nommé `S10811@ldoms-2.1-backup` :

```
# beadm create -e S10811@ldoms-2.1-backup ldoms-2.1-backup
```

2 Enregistrez-vous pour pouvoir utiliser le référentiel de logiciels en ligne.

Reportez-vous à Certificate Generator Online Help (`https://pkg-register.oracle.com/help/#support`).

3 Installez Product; version 2.2 du package `ldomsmanager` à partir du référentiel de logiciels en ligne dans le cadre d'une mise à jour vers la SRU la plus récente.

```
# pkg update
           Packages to install:  1
            Packages to update: 89
       Create boot environment: No
Create backup boot environment: No
            Services to change:  3

PHASE                                          ACTIONS
Removal Phase                                  517/517
Install Phase                                  806/806
Update Phase                                 5325/5325

PHASE                                            ITEMS
```

```
Package State Update Phase                   179/179
Package Cache Update Phase                     89/89
Image State Update Phase                         2/2
```

Remarque – Pour plus d'informations sur les SRU Oracle Solaris 11, reportez-vous à l'Oracle Solaris 11 Support Repository Updates (SRU) Index (`https://support.oracle.com/CSP/main/article?cmd=show&type=NOT&doctype=REFERENCE&id=1372094.1`).

4 **Assurez-vous que le package a été installé.**

```
# pkg info ldomsmanager
Name: system/ldoms/ldomsmanager
Summary: Logical Domains Manager
Description: LDoms Manager - Virtualization for SPARC T-Series
Category: System/Virtualization
State: Installed
Publisher: solaris
Version: 2.2.0.0
Build Release: 5.11
Branch: 0.175.0.0.0.1.0
Packaging Date: Thu Mar 01 23:06:35 2011
Size: 2.34 MB
FMRI: pkg://solaris/system/ldoms/ldomsmanager@
2.2.0.0,5.11-0.175.0.0.0.1.0:20120221T141945Z
```

5 **Redémarrez le service `ldmd`.**

```
# svcadm restart ldmd
```

6 **Assurez-vous que vous exécutez la version correcte de `ldm`.**

```
# ldm -V
```

7 **Enregistrez la configuration de votre système sur le processeur de service.**

```
# ldm add-config config-name
```

L'exemple suivant enregistre la configuration appelée `ldoms-2.2-config` :

```
# ldm add-config ldoms-2.2-config
```

Configuration usine par défaut et désactivation de Logical Domains

La configuration initiale sur laquelle la plate-forme apparaît en tant que système unique hébergeant uniquement un 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 Logical Domains et remettre la configuration sur les valeurs par défaut d'usine.

▼ Procédure de suppression de tous les domaines invités

1 Arrêtez tous les domaines à l'aide de l'option `-a`.

```
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 de toutes les configurations de domaines logiques

1 Répertoriez toutes les configurations de domaines logiques 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 -i1 -g0 -y
```

3. **Remettez le système sous tension pour charger la configuration usine par défaut.**

```
-> stop /SYS
-> start /SYS
```

▼ Procédure de désactivation de Logical Domains Manager

- **Désactivez Logical Domains Manager à partir du domaine de contrôle.**

```
primary# svcadm disable ldmd
```

Remarque – 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 ou la gestion de l'alimentation. Dans le cas du compte-rendu d'erreurs, si vous êtes dans la configuration `factory-default`, vous pouvez redémarrer le domaine de contrôle pour restaurer le compte-rendu d'erreurs. Cependant, ce n'est pas le cas avec la gestion de l'alimentation. Par ailleurs, certains outils de gestion ou de surveillance du système reposent sur Logical Domains Manager.

▼ Procédure de suppression de Logical Domains Manager

Après la restauration de la configuration usine par défaut et la désactivation de Logical Domains Manager, vous pouvez supprimer le logiciel de Logical Domains Manager.

Remarque – Si vous supprimez Logical Domains Manager avant de restaurer la configuration usine par défaut, vous pouvez restaurer celle-ci à partir du processeur de service comme indiqué dans la procédure suivante.

- **Supprimez le logiciel Logical Domains Manager.**
  - **Supprimez les packages `SUNWldm` et `SUNWldmp2v` d'Oracle Solaris 10.**

    ```
    primary# pkgrm SUNWldm SUNWldmp2v
    ```
  - **Supprimez le package `ldomsmanager` d'Oracle Solaris 11.**

    ```
    primary# pkg uninstall ldomsmanager
    ```

▼ Procédure de restauration de la configuration usine par défaut à partir du processeur de service

Si vous supprimez Logical Domains Manager avant de restaurer la configuration usine par défaut, vous pouvez restaurer celle-ci 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. **Remettez le système sous tension pour charger la configuration usine par défaut.**

   ```
   -> reset /SYS
   ```

◆ ◆ ◆ CHAPITRE 3

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 :

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

Le package Logical Domains Manager ajoute deux profils de droits de contrôle d'accès basé sur les rôles (RBAC) prédéfinis à la configuration RBAC locale. A l'aide de ces profils de droits, vous pouvez déléguer les privilèges administratifs suivants 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 ou les profils de droits en fonction de votre configuration RBAC. Reportez-vous au manuel *Guía de administración del sistema: servicios de seguridad* ou à la Partie III, "Roles, perfiles de derechos y privilegios" du manuel *Administración de Oracle Solaris: servicios de seguridad*.

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 *Guía de administración del sistema: Servicios de nombres y directorios (DNS, NIS y LDAP)*. Tous les exemples de ce chapitre supposent que la configuration RBAC utilise des fichiers locaux. 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 49.

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 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.
- 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`.

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 *Guía de administración del sistema: Servicios de nombres y directorios (DNS, NIS y 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 *Guía de administración del sistema: servicios de seguridad* ou à la Partie III, "Roles, perfiles de derechos y privilegios" du manuel *Administración de Oracle Solaris: servicios de seguridad*.

1. **Prenez le rôle d'administrateur, de superutilisateur ou un rôle équivalent.**

   Pour Oracle Solaris 10, reportez-vous à la section "Configuración de RBAC (mapa de tareas)" du manuel *Guía de administración del sistema: servicios de seguridad*. Pour Oracle Solaris 11, reportez-vous à la Partie III, "Roles, perfiles de derechos y privilegios" du manuel *Administración de Oracle Solaris: servicios de seguridad*.

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 *Guía de administración del sistema: Servicios de nombres y directorios (DNS, NIS y LDAP)*.

Cette procédure présente l'avantage que les rôles ne sont endossés que par les utilisateurs à qui ils ont été affectés. Lorsqu'un utilisateur endosse un rôle, il doit fournir le mot de passe qui a été affecté au rôle, le cas échéant. Ces deux niveaux de sécurité empêchent un utilisateur d'endosser un rôle pour lequel il détient le mot de passe, mais qui ne lui a pas été affecté.

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

1. **Prenez le rôle d'administrateur, de superutilisateur ou un rôle équivalent.**

   Pour Oracle Solaris 10, reportez-vous à la section "Configuración de RBAC (mapa de tareas)" du manuel *Guía de administración del sistema: servicios de seguridad*. Pour Oracle Solaris 11, reportez-vous à la Partie III, "Roles, perfiles de derechos y privilegios" du manuel *Administración de Oracle Solaris: servicios de seguridad*.

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 Endossez le rôle et fournissez le mot de passe, si nécessaire.

```
$ su role-name
```

9 Vérifiez que l'utilisateur a endossé le rôle.

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

Exemple 3–1 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 endosser le rôle `ldm_read`.

```
# roleadd -P "LDoms Review" ldm_read
# passwd ldm_read
New Password: ldm_read-password
Re-enter new Password: ldm_read-password
passwd: password successfully changed for ldm_read
# useradd -R ldm_read user_1
# passwd user_1
New Password: user_1-password
Re-enter new Password: user_1-password
passwd: password successfully changed for user_1
# su user_1
Password: user_1-password
$ id
uid=95555(user_1) gid=10(staff)
$ roles
ldm_read
$ su ldm_read
Password: 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 RBAC suivants dans le fichier `/etc/security/prof_attr` local.

- `LDoms Review:::Review LDoms configuration:auths=solaris.ldoms.read`
- `LDoms Management:::Manage LDoms domains:auths=solaris.ldoms.*`

Le package Logical Domains Manager ajoute également au fichier `/etc/security/exec_attr` l'attribut d'exécution suivant, associé au profil `LDoms Management` :

```
LDoms Management:suser:cmd:::/usr/sbin/ldm:privs=file_dac_read,file_dac_search
```

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

TABLEAU 3–1 Sous-commandes `ldm` et autorisations utilisateur

Sous-commande ldm[1]	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`

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

Contrôle de l'accès à une console de domaine à l'aide de RBAC

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 à 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 `auth_attr` 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 donné. Pour affecter des autorisations à des utilisateurs ou des rôles dans un service de noms, reportez-vous au manuel *Guía de administración del sistema: Servicios de nombres y directorios (DNS, NIS y 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 50 et "Procédure de contrôle de l'accès à toutes les consoles de domaines par le biais de profils de droits" à la page 52
- 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 54 et "Procédure de contrôle de l'accès à une console unique par le biais de profils de droits" à la page 55.

▼ 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 all_cons
```

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

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

Exemple 3–2 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)
-bash-3.2$ su all_cons
Password:
$ telnet 0 5000
Trying 0.0.0.0...
Connected to 0.
Escape character is '^]'.

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

$ telnet 0 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 0 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 Créez un profil de droits comportant l'autorisation `solaris.vntsd.consoles`.

- **Sous Oracle Solaris 10, modifiez le fichier `/etc/security/prof_attr`.**

  Incluez l'entrée suivante :

  ```
  LDoms Consoles:::Access LDoms Consoles:auths=solaris.vntsd.consoles
  ```

- **Sous Oracle Solaris 11, créez un profil à l'aide de la commande `profiles`.**

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

2 Affectez le profil de droits à un utilisateur.

- **Sous Oracle Solaris 10, affectez le profil de droits à un utilisateur.**

  ```
  primary# usermod -P "All,Basic Solaris User,LDoms Consoles" username
  ```

  Veuillez à indiquer les éventuels profils préexistants lorsque vous ajoutez le profil `LDoms Consoles`. La commande qui précède indique que l'utilisateur possédait déjà les profils `All` et `Basic Solaris User`.

- **Sous Oracle Solaris 11, affectez le profil de droits à un utilisateur.**

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

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

```
$ telnet 0 5000
```

Exemple 3–3 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 domaines :

- **Oracle Solaris 10 :** création d'un profil de droits incluant l'autorisation `solaris.vntsd.consoles` par ajout de l'entrée suivante au fichier `/etc/security/prof_attr` :

  ```
  LDoms Consoles:::Access LDoms Consoles:auths=solaris.vntsd.consoles
  ```

  Assignez la profil de droits à *username*.

  ```
  primary# usermod -P "All,Basic Solaris User,LDoms Consoles" username
  ```

  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 0 5000
  Trying 0.0.0.0...
  Connected to 0.
  Escape character is '^]'.

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

- **Oracle Solaris 11 :** utilisez la commande `profiles` pour créer un profil de doits présentant l'autorisation `solaris.vntsd.consoles` dans le fichier `/etc/security/prof_attr`.

  ```
  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 0 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. **Ajoutez une autorisation pour un seul domaine dans le fichier `/etc/security/auth_attr`.**

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

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

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

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

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

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

Exemple 3–4 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 le fichier `/etc/security/auth_attr` :

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

Créez ensuite un rôle possédant la nouvelle autorisation pour n'autoriser l'accès qu'à la console de 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 ldg1cons
Password:
$ id
uid=700303(ldg1cons) gid=1(other)
$ telnet 0 5000
Trying 0.0.0.0...
Connected to 0.
Escape character is '^]'.

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

Le message ci-dessous indique que l'utilisateur `terry` ne peut pas accéder à la console du domaine `ldg2` :

```
$ telnet 0 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 Ajoutez une autorisation pour un seul domaine dans le fichier `/etc/security/auth_attr`.

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

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

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

- **Sous Oracle Solaris 10, modifiez le fichier `/etc/security/prof_attr`.**

  domain-name `Console:::Access` *domain-name*
  `Console:auths=solaris.vntsd.console-`*domain-name*

  Cette entrée doit être placée sur une seule ligne.

- **Sous Oracle Solaris 11, créez un profil à l'aide de la commande `profiles`.**

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

3 Affectez le profil de droits à un utilisateur.

Les commandes suivantes assignent le profil à un utilisateur :

- **Sous Oracle Solaris 10, assignez le profil de droits.**

  `primary# usermod -P "All,Basic Solaris User,`*domain-name* `Console"` *username*

Notez que les profils `All` et `Basic Solaris User` sont indispensables.

- **Sous Oracle Solaris 11, assignez le profil de droits.**

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

Activation et utilisation de l'audit

Logical Domains Manager utilise la fonction d'audit du SE Oracle Solaris pour examiner l'historique des actions et événements qui se sont produits sur votre domaine de contrôle. L'historique est conservé dans un journal consignant ce qui a été fait, à quel moment et par qui, ainsi que les éléments affectés.

Vous pouvez activer et désactiver la fonction d'audit selon la version du SE Oracle Solaris exécutée sur votre système, comme suit :

- **SE Oracle Solaris 10.** Utilisez les commandes `bsmconv` et `bsmunconv`. Reportez-vous aux pages de manuel `bsmconv`(1M) et `bsmunconv`(1M), ainsi qu'à la Partie VII, "Auditoría de Oracle Solaris" du manuel *Guía de administración del sistema: servicios de seguridad*.
- **SE Oracle Solaris 11.** Utilisez la commande `audit`. Reportez-vous à la page de manuel `audit`(1M) et à la Partie VII, "Auditoría en Oracle Solaris" du manuel *Administración de Oracle Solaris: servicios de seguridad*.

▼ Procédure d'activation de l'audit

Vous devez configurer et activer la fonction d'audit d'Oracle Solaris sur votre système. La fonction d'audit du SE Oracle Solaris permet d'examiner l'historique des actions et des événements qui se sont produits sur votre domaine de contrôle. L'historique est conservé dans un journal consignant ce qui a été fait, à quel moment et par qui, ainsi que les éléments affectés. L'audit Oracle Solaris 11 est activé par défaut, mais vous devez effectuer certaines opérations de configuration.

Remarque – Les processus préexistants *ne sont pas* soumis à un audit concernant la classe des logiciels de virtualisation (`vs`). Assurez-vous d'effectuer cette étape *avant* que les utilisateurs standard ne se connectent au système.

---

1. **Ajoutez des personnalisations aux fichiers `/etc/security/audit_event` et `/etc/security/audit_class`.**

   Ces personnalisations sont conservées lors des mises à niveau d'Oracle Solaris, mais elles doivent être ajoutées de nouveau après une réinstallation d'Oracle Solaris.

   a. **Ajoutez l'entrée suivante au fichier `audit_event`, si elle n'y figure pas :**

      ```
      40700:AUE_ldoms:ldoms administration:vs
      ```

b. Ajoutez l'entrée suivante au fichier `audit_class`, si elle n'y figure pas :

```
0x10000000:vs:virtualization_software
```

2 (Oracle Solaris 10) Ajoutez la classe `vs` au fichier `/etc/security/audit_control`.

L'extrait de `/etc/security/audit_control` suivant indique comment spécifier la classe `vs` :

```
dir:/var/audit
flags:lo,vs
minfree:20
naflags:lo,na
```

3 (Oracle Solaris 10) Activez la fonction d'audit.

a. Exécutez la commande `bsmconv`.

```
# /etc/security/bsmconv
```

b. Réinitialisez le système.

4 (Oracle Solaris 11) Présélectionnez la classe d'audit `vs`.

a. Déterminez les classes d'audit déjà sélectionnés.

Assurez-vous que toutes les classes d'audit déjà sélectionnées font partie de l'ensemble de classes mis à jour. L'exemple suivant montre que la classe `lo` est déjà sélectionnée :

```
# auditconfig -getflags
active user default audit flags = lo(0x1000,0x1000)
configured user default audit flags = lo(0x1000,0x1000)
```

b. Ajoutez la classe d'audit `vs`.

```
# auditconfig -setflags [class],vs
```

class représente aucune, une ou plusieurs classes d'audit, séparées par des virgules. Vous pouvez voir la liste des classes d'audit dans le fichier `/etc/security/audit_class`. Il est important d'inclure la classe `vs` sur votre système Oracle VM Server for SPARC.

Par exemple, la commande suivante sélectionne les deux classes `lo` et `vs` :

```
# auditconfig -setflags lo,vs
```

c. (Facultatif) Déconnectez-vous du système si vous souhaitez soumettre les processus à un audit en tant qu'administrateur ou en tant que responsable de la configuration.

Si vous ne souhaitez pas vous déconnecter, reportez-vous à la section "Cómo actualizar la máscara de preselección de usuarios con sesión iniciada" du manuel *Administración de Oracle Solaris: servicios de seguridad*.

5 Vérifiez que le logiciel d'audit est en cours d'exécution.

```
# auditconfig -getcond
```

Si le logiciel d'audit est en cours d'exécution, `audit condition = auditing` s'affiche dans la sortie.

▼ Procédure de désactivation de l'audit

- **Désactivez la fonction d'audit.**
  - **Désactivez la fonction d'audit sur votre système Oracle Solaris 10.**
    a. **Exécutez la commande `bsmunconv`.**

```
# /etc/security/bsmunconv
Are you sure you want to continue? [y/n] y
This script is used to disable the Basic Security Module (BSM).
Shall we continue the reversion to a non-BSM system now? [y/n] y
bsmunconv: INFO: removing c2audit:audit_load from /etc/system.
bsmunconv: INFO: stopping the cron daemon.

The Basic Security Module has been disabled.
Reboot this system now to come up without BSM.
```

    b. **Réinitialisez le système.**
  - **Désactivez la fonction d'audit sur votre système Oracle Solaris 11.**
    a. **Exécutez la commande `audit -t`.**

```
# audit -t
```

    b. **Vérifiez que le logiciel d'audit n'est plus en cours d'exécution.**

```
# auditconfig -getcond
audit condition = noaudit
```

▼ Procédure d'examen des enregistrements d'audit

- **Utilisez l'une des méthodes suivantes pour examiner la sortie d'audit `vs` :**
  - **Utilisez les commandes `auditreduce` et `praudit` pour examiner la sortie d'audit.**

```
# auditreduce -c vs | praudit
# auditreduce -c vs -a 20060502000000 | praudit
```

  - **Utilisez la commande `praudit -x` pour imprimer les enregistrements d'audit au format XML.**

▼ Procédure de rotation des journaux d'audit

- **Utilisez la commande `audit -n` pour faire tourner les journaux d'audit.**

  La rotation des journaux d'audit ferme le fichier d'audit actuel et en ouvre un nouveau dans le répertoire d'audit actuel.

◆ ◆ ◆ CHAPITRE 4

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.

Vous pouvez également utiliser l'assistant de configuration d'Oracle VM Server for SPARC pour configurer les domaines logiques et les services. Reportez-vous au Chapitre 14, “Assistant de configuration d'Oracle VM Server for SPARC (Oracle Solaris 10)”.

Ce chapitre aborde les sujets suivants :

Messages de sortie

A partir de la version 2.0 d'Oracle VM Server for SPARC, si une ressource ne peut pas être configurée de manière dynamique sur le domaine de contrôle, il convient de lancer d'abord une reconfiguration retardée. La reconfiguration retardée reporte les activités de configuration à après le redémarrage 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'au redémarrage :

```
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 de 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.

   - Dans Oracle Solaris 10, ajoutez un service de commutateur virtuel sur le pilote de l'adaptateur réseau dans le domaine de contrôle.

     ```
     primary# ldm add-vsw net-dev=net-driver vsw-service primary
     ```

     Par exemple, la commande suivante ajoute un service de commutateur virtuel (`primary-vsw0`) sur le pilote d'adaptateur réseau (`nxge0`) dans le domaine de contrôle (`primary`) :

```
primary# ldm add-vsw net-dev=nxge0 primary-vsw0 primary
```

- Dans Oracle Solaris 11, ajoutez un service de commutateur virtuel (`primary-vsw0`) sur le pilote de l'adaptateur réseau `net0` dans le domaine de contrôle (`primary`) :

  ```
  primary# ldm add-vsw net-dev=net-driver  vsw-service  primary
  ```

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

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

- La commande suivante s'applique uniquement au SE Oracle Solaris 10 et *ne doit pas* être effectuée sur un système Oracle Solaris 11.

  Cette commande alloue automatiquement une adresse MAC au commutateur virtuel. Vous pouvez définir votre propre adresse MAC sous forme d'option à la commande `ldm add-vsw`. Cependant, dans ce cas, il est de votre responsabilité de vous assurer que l'adresse MAC indiquée ne soit pas en conflit avec une adresse MAC déjà existante.

  Si le commutateur virtuel ajouté remplace l'adaptateur physique sous-jacent en tant qu'interface réseau principale, il doit être associé à l'adresse MAC de l'adaptateur physique, afin que le serveur DHCP (Dynamic Host Configuration Protocol) assigne au domaine la même adresse IP. Reportez-vous à la section "Activation de la mise en réseau entre le domaine de contrôle/service et les autres domaines" à la page 65.

  ```
  primary# ldm add-vsw mac-addr=2:04:4f:fb:9f:0d net-dev=nxge0 primary-vsw0 primary
  ```

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 nxge0     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.

N'essayez *pas* d'utiliser la reconfiguration dynamique (DR) de mémoire pour effectuer la configuration initiale du domaine de contrôle. Bien que vous puissiez utiliser la reconfiguration dynamique de mémoire pour effectuer cette configuration sans redémarrage, cela n'est *pas*

recommandé. L'approche de reconfiguration dynamique de mémoire peut prendre un certain temps (plus long qu'un redémarrage) et peut éventuellement échouer. Utilisez plutôt la commande `ldm start-reconf` pour faire passer le domaine de contrôle en mode de reconfiguration retardée avant de modifier la configuration de la mémoire. Vous pouvez ensuite redémarrer le domaine de contrôle après avoir terminé toutes les étapes de configuration.

▼ Procédure de configuration du domaine de contrôle

Remarque – 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.

1. **Déterminez si vous avez des périphériques cryptographiques dans le domaine de contrôle.**

   Notez que seules les plates-formes UltraSPARC T2, UltraSPARC T2 Plus et SPARC T3 peuvent être équipées de périphériques cryptographiques.

   ```
   primary# ldm list -o crypto primary
   ```

2. **Le cas échéant, assignez des ressources cryptographiques au domaine de contrôle.**

   L'exemple suivant assigne une ressource cryptographique au domaine de contrôle, `primary`. Cette opération laisse le reste des ressources cryptographiques disponibles pour un domaine invité.

   ```
   primary# ldm set-mau 1 primary
   ```

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

   Par exemple, la commande suivante assigne 8 CPU virtuelles au domaine de contrôle `primary`. Cette opération laisse le reste des CPU virtuelles disponibles pour un domaine invité.

   ```
   primary# ldm set-vcpu 8 primary
   ```

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

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

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

   Par exemple, la commande suivante assigne 4 giga-octets de mémoire au domaine de contrôle `primary`. Cette opération laisse le reste de la mémoire disponible pour un domaine invité.

   ```
   primary# ldm set-memory 4G primary
   ```

6. **Ajoutez une configuration de machine de domaine logique au processeur de service (SP).**

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

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

7 **Vérifiez que la configuration est prête à être utilisée au prochain redémarrage.**

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

Cette sous-commande de liste affiche la configuration `initial` définie qui sera utilisée à la prochaine remise sous tension.

Réinitialisation pour utiliser Logical Domains

Vous devez redémarrer 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 redémarrez le domaine de contrôle.**

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

Remarque – Une réinitialisation ou une remise sous tension instancie la nouvelle configuration. Seule une remise sous tension 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 contrôle/service et les autres domaines

Attention – Cette section s'applique *uniquement* au système Oracle Solaris 10. Ne configurez *pas* l'interface `vsw` sur le système Oracle Solaris 11.

Par défaut, la mise en réseau entre le domaine de contrôle et les autres domaines du système est désactivée. Pour l'activer, le périphérique de commutateur virtuel doit être configuré 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 contrôle ou le domaine de service tant que le périphérique backend réseau correspondant est configuré dans le même LAN virtuel ou réseau virtuel.

Remarque – Procédez comme suit à partir de la console du domaine de contrôle, car cette procédure peut interrompre momentanément la connectivité au domaine.

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

Remarque – 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 162

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

   ```
   primary# ifconfig -a
   ```

2. **Créez le commutateur virtuel.**

   ```
   primary# ifconfig vsw0 plumb
   ```

3. **Supprimez le périphérique réseau physique affecté au commutateur virtuel (`net-dev`).**

   ```
   primary# 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.**

     ```
     primary# 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.**

     ```
     primary# ifconfig vsw0 dhcp start
     ```

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

   ```
   primary# mv /etc/hostname.nxge0 /etc/hostname.vsw0
   primary# 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 que vous avez créé le service par défaut `vconscon` (`vcc`) sur le domaine de contrôle avant d'activer `vntsd`. Reportez-vous à la section "Création des services par défaut" à la page 62 pour plus d'informations.

---

1 **Utilisez la commande `svcadm` pour activer le démon du serveur de terminal réseau virtuel, `vntsd`.**

```
primary# svcadm enable vntsd
```

2 **Utilisez la commande `svcs` pour vérifier que le démon `vntsd` est activé.**

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

◆ ◆ ◆ CHAPITRE 5

Configuration des domaines invités

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

Vous pouvez également utiliser l'assistant de configuration d'Oracle VM Server for SPARC pour configurer les domaines logiques et les services. Reportez-vous au Chapitre 14, “Assistant de configuration d'Oracle VM Server for SPARC (Oracle Solaris 10)”.

Ce chapitre aborde les sujets suivants :

Création et démarrage d'un domaine invité

Le domaine invité doit exécuter le système d'exploitation qui comprend la plate-forme `sun4v` et les périphériques virtuels présentés par l'hyperviseur. Actuellement, cela signifie que vous devez exécuter au moins le SE Oracle Solaris 10 11/06. L'exécution du SE Oracle Solaris 10 8/11 vous fournit toutes les fonctions d'Oracle VM Server for SPARC .2.2. Reportez-vous aux *Notes de version d'Oracle VM Server for SPARC 2.2* 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.

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

1 Créez un domaine logique.

Par exemple, la commande suivante crée un domaine invité nommé `ldg1`.

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

2 Ajoutez des CPU au domaine invité.

Par exemple, la commande suivante ajoute huit CPU virtuelles au domaine invité `ldg1`.

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

3 Ajoutez de la mémoire au domaine invité.

Par exemple, la commande suivante ajoute 2 giga-octets de mémoire au domaine invité `ldg1`.

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

4 Ajoutez un périphérique réseau virtuel au domaine invité.

Par exemple, 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 en savoir plus 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 127 et "Utilisation de ZFS avec les disques virtuels" à la page 137.

---

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 le premier 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 du périphérique physique réel. Lors de l'ajout d'un périphérique, le nom de chemin 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 du périphérique réel.
- `primary-vds0` est le nom du serveur de disque virtuel auquel ajouter ce périphérique.

- **Exemple d'un fichier.** Dans le deuxième 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 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 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 : c*N*d*N*s*N*, où c*N* est le contrôleur virtuel, d*N* est le numéro du disque virtuel et s*N* est la tranche.

---

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

Le premier exemple de commande définit `auto-boot?` sur `true` pour le domaine invité `ldg1`.

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

Le second exemple de commande définit `boot-device` sur `vdisk` 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 au Chapitre 7, “Uso de comandos

de configuración de interfaces y enlaces de datos en perfiles" du manuel *Administración de Oracle Solaris: interfaces y virtualización de redes*.

▼ Procédure d'installation d'SE Oracle Solaris sur un domaine invité à partir d'un DVD

1 Insérez le DVD du SE Oracle Solaris 10 ou Oracle Solaris 11 dans le lecteur 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 lecteur DVD dans lequel le média d'Oracle Solaris réside, `dvd_vol@primary-vds0` comme volume secondaire et `vdisk_cd_media` comme disque virtuel.

```
primary# ldm add-vdsdev /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      4     4G       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 027.0.0.1...
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, démarrez 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 dans le menu d'installation du SE Oracle Solaris.

▼ Procédure d'installation du SE Oracle Solaris sur un domaine invité à partir 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      4     4G       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, démarrez 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 dans le menu d'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. Pour effectuer une installation automatisée du système d'exploitation Oracle Solaris 11, vous pouvez utiliser le programme d'installation automatisée (AI). Reportez-vous à la section *Transición de Oracle Solaris 10 JumpStart a Oracle Solaris 11 Automated Installer*.

Cette procédure décrit l'utilisation de la fonction JumpStart d'Oracle Solaris sur un domaine invité. Cette procédure suit la procédure JumpStart classique, mais elle décrit le format de nom de périphérique de disque différent à utiliser dans le profil JumpStart pour le domaine invité. Reportez-vous au manuel *Guía de instalación de Oracle Solaris 10 8/11: instalaciones JumpStart personalizadas y avanzadas*.

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.

- **Modifiez votre profil JumpStart pour prendre en compte cette modification.**

  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 du disque virtuel" à la page 119.

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

    Reportez-vous au manuel *Guía de instalación de Oracle Solaris 10 8/11: instalaciones JumpStart personalizadas y avanzadas*.

    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 racine ZFS.**

  Reportez-vous au Chapitre 9, "Instalación de una agrupación raíz ZFS con JumpStart" du manuel *Guía de instalación de Oracle Solaris 10 8/11: instalaciones JumpStart personalizadas y avanzadas*.

  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 6

Configuration des domaines d'E/S

Ce chapitre décrit les domaines d'E/S et la procédure de configuration de ces derniers dans un environnement de Logical Domains.

Ce chapitre aborde les sujets suivants :

Présentation d'un domaine d'E/S

Un domaine d'E/S a la propriété directe de l'accès direct aux périphériques d'E/S physiques. Il peut être créé en assignant un bus PCIe (PCI EXPRESS) ou un périphérique d'extrémité PCIe à un domaine. Utilisez la commande `ldm add-io` pour assigner un bus ou un périphérique à un domaine.

Vous pouvez vouloir configurer des domaines d'E/S pour les raisons suivantes :

- 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 d'autres domaines invités.

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

- "Assignation de bus PCIe" à la page 81
- "Assignation des périphériques d'extrémité PCIe" à la page 86

Remarque – Vous *ne pouvez pas* migrer un domaine d'E/S configuré avec des périphériques d'extrémité PCIe. Pour plus d'informations sur les autres limitations de migration, reportez-vous au Chapitre 9, “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 interfaces de réseau local (NIU), des périphériques d'extrémité PCIe et des fonctions virtuelles de virtualisation d'E/S à racine unique (SR-IOV) 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, un périphérique Ethernet 1 Gbits/s peut nécessiter un plus petit nombre de coeurs de processeur pour exploiter l'ensemble de la bande passante qu'un périphérique Ethernet 10 Gbits/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.

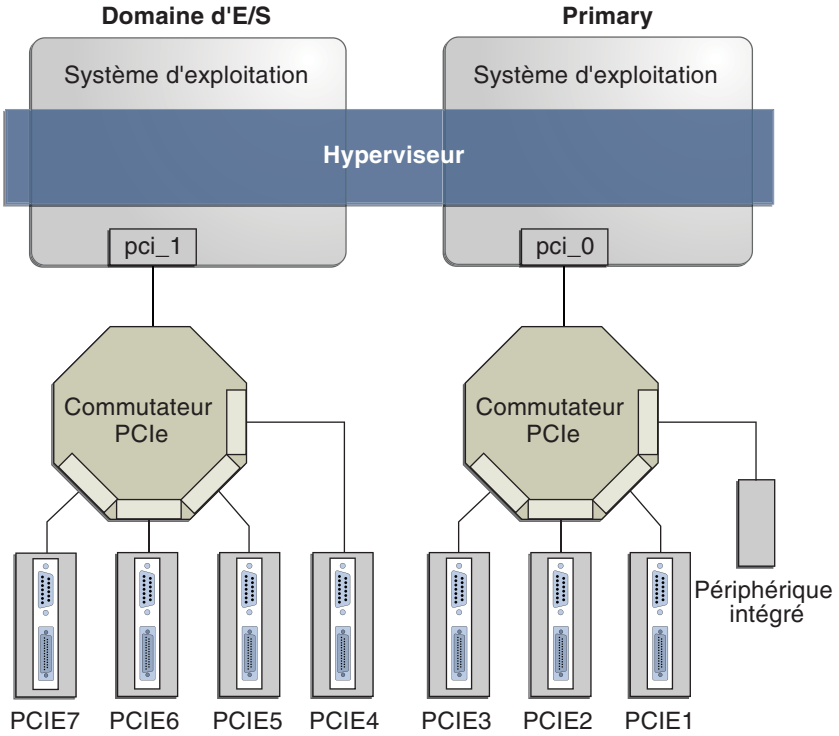
Notez que la création et l'affectation d'un grand nombre de fonctions virtuelles à un domaine qui ne dispose pas d'un nombre suffisant de ressources CPU et de mémoire ont peu de chances d'aboutir à une configuration optimale.

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 suivante représente un système avec deux bus PCIe (`pci_0`et `pci_1`). Chaque bus est assigné à un domaine différent. Ainsi, le système est configuré avec deux domaines d'E/S.

FIGURE 6–1 Assignation d'un bus PCIe à un domaine d'E/S



Le nombre maximum de domaines d'E/S que vous pouvez créer avec des bus PCIe dépend du nombre de bus PCIe disponibles sur le serveur. Par exemple, si vous utilisez un serveur Sun SPARC Enterprise T5440, vous pouvez avoir jusqu'à quatre domaines d'E/S.

Remarque – Certains serveurs Sun UltraSPARC ne peuvent avoir qu'un seul bus PCIe. Dans ce cas, vous pouvez créer un domaine d'E/S en assignant un périphérique d'extrémité PCIe (ou un périphérique pouvant être associé à des E/S directes) à un domaine. Reportez-vous à la section "Assignation des périphériques d'extrémité PCIe" à la page 86. Si le système a une unité d'interface réseau (NIU), vous pouvez également assigner une NIU à un domaine pour créer un domaine d'E/S.

Lorsque vous assignez un bus PCIe à un domaine d'E/S, tous les périphériques sur ce bus sont détenus par le domaine d'E/S. Vous n'êtes pas autorisé à assigner l'un des périphériques d'extrémité PCIe sur ce bus à d'autres domaines. Seuls les périphériques d'extrémité PCIe sur les bus PCIe qui sont assignés au domaine `primary` peuvent être assignés à d'autres domaines.

Si un serveur est configuré dès le départ dans un environnement Logical Domains ou utilise la configuration `factory-default`, le domaine `primary` a accès à toutes les ressources du périphérique physique. Cela signifie que le domaine `primary` est le seul domaine d'E/S configuré sur le système et qu'il possède tous les bus PCIe.

▼ Procédure de création d'un domaine d'E/S par assignation d'un bus PCIe

Cet exemple de procédure montre comment créer un nouveau domaine d'E/S à partir de la configuration initiale dans laquelle plusieurs bus sont détenus par le domaine `primary`. Par défaut, le domaine `primary` détient tous les bus présents sur le système. Cet exemple est pour un serveur Sun SPARC Enterprise T5440. Cette procédure peut également être utilisée sur d'autres serveurs. Les instructions pour les différents serveurs peuvent varier légèrement de celles-ci, mais vous pouvez obtenir les principales de base à partir de cet exemple.

En premier lieu, vous devez conserver le bus ayant le disque d'initialisation du domaine `primary`. Supprimez ensuite un autre bus du domaine `primary` et assignez-le à un autre domaine.

Attention – Tous les disques internes sur les serveurs pris en charge peuvent être connectés à un seul bus PCIe. Si un domaine est démarré à partir d'un disque interne, ne supprimez pas de bus du domaine. Vérifiez également que vous n'êtes pas en train de supprimer un bus avec des périphériques (notamment des ports réseau) 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.

Dans cet exemple, le domaine `primary` n'utilise qu'un pool ZFS (`rpool` (`c0t1d0s0`)) et une interface réseau (`nxge0`). Si le domaine `primary` utilise plusieurs périphériques, répétez les étapes 2 à 4 pour chaque périphérique afin de vous assurer qu'aucun ne soit situé sur le bus qui va être supprimé.

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

```
primary# ldm list-io
NAME                                      TYPE   DOMAIN   STATUS
----                                      ----   ------   ------
pci_0                                     BUS    primary
pci_1                                     BUS    primary
pci_2                                     BUS    primary
pci_3                                     BUS    primary
MB/PCIE0                                  PCIE   -        EMP
MB/PCIE1                                  PCIE   primary  OCC
MB/HBA                                    PCIE   primary  OCC
MB/PCIE4                                  PCIE   -        EMP
MB/PCIE5                                  PCIE   -        EMP
MB/XAUI1                                  PCIE   primary  OCC
MB/PCIE2                                  PCIE   primary  OCC
MB/PCIE3                                  PCIE   primary  OCC
MB/PCIE6                                  PCIE   primary  OCC
MB/PCIE7                                  PCIE   -        EMP
```

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/c0t1d0s0 ): 1309384 blocks   457028 files
  ```

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

  ```
  primary# df /
  /                  (rpool/ROOT/s10s_u8wos_08a):245176332 blocks 245176332 files
  primary# zpool status rpool
  ```

```
 zpool status rpool
  pool: rpool
 state: ONLINE
 scrub: none requested
config:

        NAME        STATE     READ WRITE CKSUM
        rpool       ONLINE       0     0     0
          c0t1d0s0  ONLINE       0     0     0
```

3 Déterminez le périphérique physique auquel le périphérique en mode bloc est connecté.

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`, qui correspond à la liste précédente de `pci_0`. Cela signifie que vous *ne pouvez pas* assigner `pci_0` (`pci@400`) à un autre domaine.

4 Déterminez l'interface réseau qui est utilisée par le système.

- **Oracle Solaris 10. Exécutez la commande ci-dessous :**

```
primary# dladm show-dev
vsw0            link: up        speed: 1000  Mbps       duplex: full
nxge0           link: up        speed: 1000  Mbps       duplex: full
nxge1           link: unknown   speed: 0     Mbps       duplex: unknown
nxge2           link: unknown   speed: 0     Mbps       duplex: unknown
nxge3           link: unknown   speed: 0     Mbps       duplex: unknown
```

- **Oracle Solaris 11. Exécutez la commande ci-dessous :**

```
primary# dladm show-phys
LINK              MEDIA                STATE      SPEED  DUPLEX    DEVICE
net0              Ethernet             unknown    0      unknown   vnet0
net1              Ethernet             up         1000   full      vsw0
net2              Ethernet             up         1000   full      nxge0
```

Les interfaces qui sont à l'état `unknown` ne sont pas configurées, par conséquent elles ne sont pas utilisées. Dans cet exemple, l'interface `nxge0` est utilisée.

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

La commande suivante utilise l'interface réseau `nxge0` :

```
primary# ls -l /dev/nxge0
lrwxrwxrwx   1 root     root          46 Oct  1 10:39 /dev/nxge0 ->
../devices/pci@500/pci@0/pci@c/network@0:nxge0
```

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 trouvait sur un bus que vous voulez assigner à un autre domaine, le domaine `primary` devrait être reconfiguré pour utiliser une interface réseau différente.

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

Dans cet exemple, le bus `pci_2` et le bus `pci_3` sont supprimés du domaine `primary`. Vous risquez de voir un message de la commande `ldm` indiquant que le domaine `primary` passe en mode de reconfiguration retardée.

```
primary# ldm remove-io pci_2 primary
primary# ldm remove-io pci_3 primary
```

7 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 devant être utilisée après le redémarrage.

Remarque – A l'heure actuelle, il est possible d'enregistrer au plus huit configurations dans le SP de systèmes UltraSPARC T2 et UltraSPARC T2 Plus, sans compter la configuration `factory-default`. Sur les systèmes SPARC T3 et SPARC T4, 10 Mo d'espace sont disponibles pour le stockage des configurations. Le nombre total de configurations enregistrées dépend de la taille de chaque configuration stockée dans le processeur de service.

---

8 Redémarrez le domaine `primary` afin que les modifications soient appliquées.

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

9 Arrêtez le domaine auquel vous souhaitez ajouter le bus PCIe.

L'exemple suivant arrête le domaine `ldg1` :

```
primary# ldm stop ldg1
```

10 Ajoutez le bus disponible au domaine ayant besoin d'un accès direct.

Le bus disponible est `pci_2` et le domaine est `ldg1`.

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

11 Redémarrez le domaine afin que les modifications soient appliquées.

Les commandes suivantes redémarre le domaine `ldg1` :

```
primary# ldm start ldg1
```

12 Confirmez 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   DOMAIN   STATUS
----                                      ----   ------   ------
pci_0                                     BUS    primary
pci_1                                     BUS    primary
pci_2                                     BUS    ldg1
pci_3                                     BUS
MB/PCIE0                                  PCIE   -        EMP
MB/PCIE1                                  PCIE   primary  OCC
MB/HBA                                    PCIE   primary  OCC
MB/PCIE4                                  PCIE   -        EMP
MB/PCIE5                                  PCIE   -        EMP
MB/XAUI1                                  PCIE   primary  OCC
MB/PCIE2                                  PCIE   -        UNK
MB/PCIE3                                  PCIE   -        UNK
MB/PCIE6                                  PCIE   -        UNK
MB/PCIE7                                  PCIE   -        UNK
```

Cette sortie confirme que les bus PCIe `pci_0` et `pci_1` et les périphériques situés en dessous sont assignés au domaine `primary`, et que `pci_2` et ses périphériques sont assignés à `ldg1`.

Assignation des périphériques d'extrémité PCIe

A partir de la version 2.0 d'Oracle VM Server for SPARC et du SE Oracle Solaris 10 9/10, vous pouvez assigner un périphérique d'extrémité PCIe (ou un périphérique pouvant être associé à des E/S directes) 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 plus de domaines d'E/S que le nombre de bus PCIe dans un système. Le nombre possible de domaines d'E/S est maintenant limité uniquement au nombre de périphériques d'extrémité PCIe.

Un périphérique d'extrémité PCIe peut être l'un des suivants :

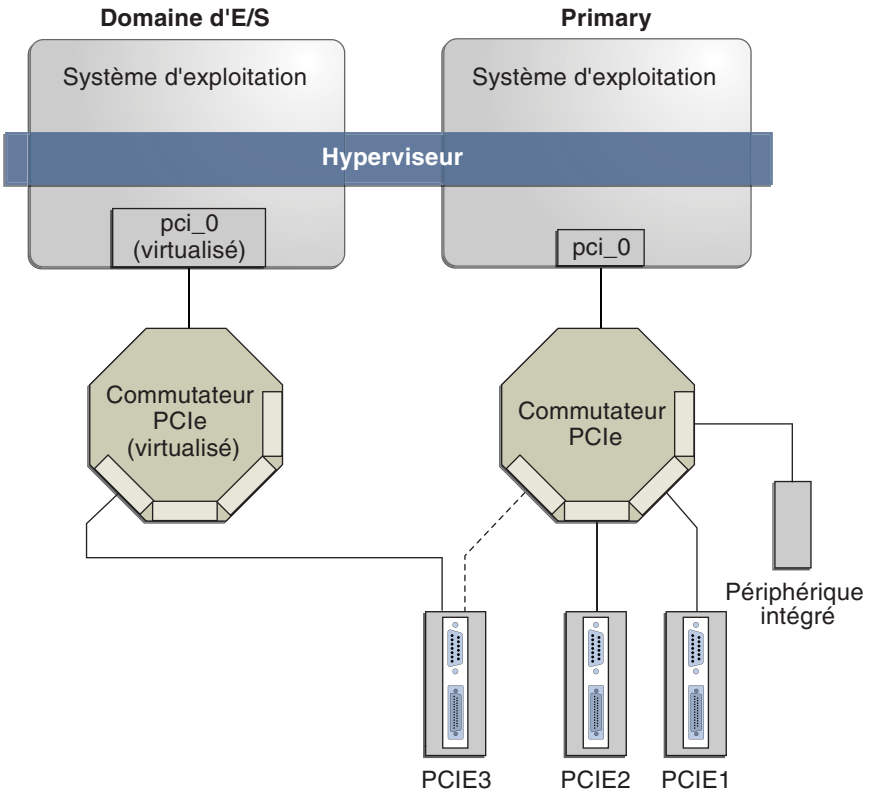
- Une carte PCIe dans un emplacement
- Un périphérique PCIe intégré identifié par la plate-forme

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 sont un complexe racine virtuel et un commutateur PCIe virtuel, respectivement. Ce bloc et ce commutateur sont très semblables au bloc `pci_0` et au commutateur du domaine `primary`. Dans le domaine `primary`, les périphériques de l'emplacement `PCIE3` sont une forme de shadow des périphériques d'origine et sont identifiés par `SUNW,assigned`.

Attention – Vous ne pouvez *pas* utiliser les opérations d'enfichage à chaud d'Oracle Solaris pour “retirer à chaud” un périphérique d'extrémité PCIe après l'avoir supprimé du domaine `primary` en utilisant la commande `ldm rm-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 92.

FIGURE 6–2 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. Reportez-vous à la section “Configuration matérielle et logicielle requise pour les E/S directes” du manuel *Notes de version d'Oracle VM Server for SPARC 2.2*.

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.

Voici quelques détails importants sur la fonction DIO :

- Cette fonction n'est activée que lorsque la configuration logicielle requise est respectée. Reportez-vous à la section "Configuration matérielle et logicielle requise pour les E/S directes" du manuel *Notes de version d'Oracle VM Server for SPARC 2.2*.
- Seules les extrémités connectées à un bus PCIe assigné au domaine `primary` peuvent être assignées à un autre domaine avec la fonction DIO.
- Les domaines d'E/S utilisant les DIO ont accès aux périphériques d'extrémité PCIe uniquement si le domaine `primary` est en cours d'exécution.
- Le redémarrage du domaine `primary` a une incidence sur les domaines d'E/S ayant des périphériques d'extrémité PCIe. Reportez-vous à la section "Redémarrage du domaine `primary`" à la page 91 Le domaine `primary` a également les responsabilités suivantes :
  - Initialise le bus PCIe et gère ce bus.
  - Traite toutes les erreurs de bus déclenchées par les périphériques d'extrémité PCIe assignés aux 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 avec succès la fonction DIO, vous devez exécuter le logiciel approprié et n'assigner que des cartes PCIe prises en charge par la fonction DIO vers les domaines d'E/S. Pour plus d'informations sur la configuration matérielle et logicielle requise, reportez-vous à la section "Configuration matérielle et logicielle requise pour les E/S directes" du manuel *Notes de version d'Oracle VM Server for SPARC 2.2*.

Remarque – Toutes les cartes PCIe prises en charge sur la plate-forme sont prises en charge dans le domaine `primary`. Reportez-vous à la documentation de votre plate-forme pour obtenir la liste des cartes PCIe prises en charge. Cependant, *seules* les cartes PCIe d'E/S directe peuvent être assignées aux domaines d'E/S.

Restrictions actuelles de la fonctionnalité d'E/S directes

Pour plus d'informations sur la manière de contourner les restrictions suivantes, reportez-vous à la section “Planification de la configuration des périphériques d'extrémité PCIe” à la page 89.

- Un reconfiguration retardée est démarrée lorsque vous assignez un périphérique d'extrémité PCIe au domaine `primary` ou que vous le supprimez, ce qui signifie que les modifications ne sont appliquées qu'après le redémarrage du domaine `primary`.

  La réinitialisation du domaine `primary` affecte les E/S directes. En conséquence, planifiez correctement vos modifications de configuration d'E/S directes pour optimiser les modifications relatives aux E/S directes sur le domaine `primary` et pour réduire les réinitialisations du domaine `primary`.
- L'assignation d'un périphérique d'extrémité PCIe à un autre domaine ou sa suppression est uniquement autorisée lorsque ce domaine est arrêté ou inactif.

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 `primary`. Le redémarrage du domaine `primary`a non seulement une incidence sur les services disponibles sur le domaine `primary` lui-même, mais affecte également les domaines d'E/S ayant des périphériques d'extrémité PCIe 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.

La reconfiguration retardée est initiée la première fois que vous assignez ou supprimez un périphérique. En conséquence, vous pouvez continuer à ajouter ou à supprimer plus de périphériques, puis redémarrer le domaine `primary` une seule fois 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 92.

Voici une descriptions de la procédure générale que vous devez suivre pour planifier et effectuer une configuration de périphérique DIO :

1. Comprenez et enregistrez la configuration matérielle de votre système.

   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 une référence ultérieure.

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 de disque d'initialisation
   - Périphérique réseau
   - Autres périphériques que le domaine `primary` offre comme services

3. Supprimez tous les périphériques d'extrémité PCIe qui vous risquez d'utiliser dans les domaines d'E/S.

   Cette étape vous aide à éviter d'effectuer des opérations de redémarrage ultérieures sur le domaine `primary`, car les redémarrages ont une incidence sur les domaines d'E/S.

   Utilisez la commande `ldm rm-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 `rm-io` et `add-io`.

   Remarque – Bien que la première suppression d'un périphérique d'extrémité PCIe risque de démarrer une reconfiguration retardée, vous pouvez poursuivre la suppression des périphériques. Après avoir supprimé tous les périphériques que vous souhaitez, vous devez seulement redémarrer le domaine `primary` une fois pour appliquer toutes les modifications.

   ---

4. Enregistrez cette configuration sur le processeur de service (SP).

   Utilisez la commande `ldm add-config`.

5. Redémarrez le domaine `primary` pour libérer les périphériques d'extrémité PCIe que vous avez supprimés à l'étape 3.

6. Confirmez que les périphériques d'extrémité sur vous supprimez ne sont plus assignés au domaine `primary`.

   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 le 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 redémarrage et en évitant les indisponibilités des services offerts par ce domaine.

9. (Facultatif) Apportez des modifications au matériel PCIe.

Reportez-vous à la section "Procédure de modification matérielle PCIe" à la page 92.

Redémarrage du domaine primary

Le domaine `primary` est le propriétaire du bus PCIe et est responsable de l'initialisation et de la gestion de ce bus. Le domaine `primary` doit être actif et exécuter une version du SE Oracle Solaris prenant en charge la fonction DIO. L'arrêt, l'interruption ou le redémarrage du domaine `primary` interrompt 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 des domaines d'E/S avec des périphériques d'extrémité PCIe est imprévisible lorsque le domaine `primary` est redémarré tandis 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 un redémarrage. Lors du redémarrage du domaine `primary`, vous devrez arrêter et démarrer manuellement chaque domaine.

Pour contourner ces problèmes, procédez comme suit :

- Arrêtez manuellement les domaines du système ayant des périphériques d'extrémité PCIe leur étant assignés *avant* d'arrêter le domaine `primary`.

  Cette étape garantit que les domaines sont arrêtés proprement avant d'arrêter, suspendre ou redémarrer le domaine `primary`.

  Pour rechercher tous les domaines ayant des périphériques d'extrémité PCIe leur étant 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. Utilisez donc ces informations pour vous aider à la planification. 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 `primary` et les domaines ayant des périphériques d'extrémité PCIe leur étant assignés.

  Cette relation de dépendance garantit que les domaines avec des périphériques d'extrémité PCIe sont automatiquement redémarrés lorsque le domaine `primary` redémarre 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 ayant été manuellement arrêtés.

  ```
  # ldm set-domain failure-policy=reset primary
  # ldm set-domain master=primary ldom
  ```

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 spécifique, 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 une nouvelle carte PCIe à un domaine d'E/S, utilisez la commande `ldm rm-io` pour d'abord supprimer la carte du domaine `primary`. 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 `primary`.
- 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 `primary` avant de supprimer 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 rm-io`. Utilisez ensuite la commande `ldm add-io` pour réassigner cette carte PCIe au domaine `primary`. Remplacez ensuite physiquement la carte PCIe que vous avez assignée au domaine `primary` par une carte PCIe différente. Ces étapes vous permettent d'éviter une configuration non prise en charge par la fonction DIO.

▼ 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 89.

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 les commandes `dladm` ou `ipadm` pour gérer la mise en réseau.
- `Automatic`. Vous permet d'utiliser les commandes `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, "Uso de comandos de configuración de interfaces y enlaces de datos en perfiles" du manuel *Administración de Oracle Solaris: interfaces y virtualización de redes*.

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 aux domaines d'E/S, l'identité des périphériques peut uniquement être déterminée dans les domaines d'E/S.

```
# ldm list-io -l
NAME                                       TYPE   DOMAIN   STATUS
----                                       ----   ------   ------
pci_0                                      BUS    primary
[pci@400]
pci_1                                      BUS    primary
[pci@500]
PCIE1                                      PCIE   -        EMP
[pci@400/pci@0/pci@c]
PCIE2                                      PCIE   primary  OCC
[pci@400/pci@0/pci@9]
    network@0
    network@0,1
    network@0,2
    network@0,3
PCIE3                                      PCIE   primary  OCC
[pci@400/pci@0/pci@d]
    SUNW,emlxs/fp/disk
    SUNW,emlxs@0,1/fp/disk
    SUNW,emlxs@0,1/fp@0,0
MB/SASHBA                                  PCIE   primary  OCC
[pci@400/pci@0/pci@8]
    scsi@0/tape
    scsi@0/disk
    scsi@0/sd@0,0
    scsi@0/sd@1,0
PCIE0                                      PCIE   -        EMP
[pci@500/pci@0/pci@9]
PCIE4                                      PCIE   primary  OCC
[pci@500/pci@0/pci@d]
    network@0
    network@0,1
PCIE5                                      PCIE   primary  OCC
[pci@500/pci@0/pci@c]
    SUNW,qlc@0/fp/disk
    SUNW,qlc@0/fp@0,0
    SUNW,qlc@0,1/fp/disk
    SUNW,qlc@0,1/fp@0,0
MB/NET0                                    PCIE   primary  OCC
[pci@500/pci@0/pci@8]
    network@0
    network@0,1
    network@0,2
    network@0,3
```

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/c0t1d0s0 ): 1309384 blocks   457028 files
```

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

```
primary# df /
/                  (rpool/ROOT/s10s_u8wos_08a):245176332 blocks 245176332 files
primary# zpool status rpool
 zpool status rpool
  pool: rpool
 state: ONLINE
 scrub: none requested
config:

        NAME        STATE     READ WRITE CKSUM
        rpool       ONLINE       0     0     0
          c0t1d0s0  ONLINE       0     0     0
```

3 Déterminez le périphérique physique auquel le périphérique en mode bloc est connecté.

L'exemple suivant utilise le périphérique en mode bloc `c0t1d0s0` :

```
primary# ls -l /dev/dsk/c0t1d0s0
lrwxrwxrwx    1 root    root    49 Jul 20 22:17 /dev/dsk/c0t1d0s0 ->
../../devices/pci@400/pci@0/pci@8/scsi@0/sd@0,0:a
```

Dans cet exemple, le périphérique physique du disque d'initialisation du domaine `primary` est connecté au périphérique d'extrémité PCIe (`pci@400/pci@0/pci@8`), qui correspond à la liste de `MB/SASHBA` à l'étape 1. La suppression de ce périphérique empêchera le redémarrage du domaine `primary`. Par conséquent, *ne* supprimez pas ce périphérique du domaine `primary`.

4 Déterminez l'interface réseau qui est utilisée par le système.

- **SE Oracle Solaris 10. Servez-vous de la commande `ifconfig` pour déterminer l'interface réseau utilisée par le domaine `primary` (il s'agit `nxge0` dans cet exemple).**

```
# ifconfig -a
lo0: flags=2001000849<UP,LOOPBACK,RUNNING,MULTICAST,IPv4,VIRTUAL> mtu 8232 index 1
        inet 127.0.0.1 netmask ff000000
nxge0: flags=1004843<UP,BROADCAST,RUNNING,MULTICAST,DHCP,IPv4> mtu 1500 index 2
        inet 10.6.212.149 netmask fffffe00 broadcast 10.6.213.255
        ether 0:21:28:4:27:cc
```

- **SE Oracle Solaris 11. Servez-vous de la commande `ipadm` pour déterminer l'interface réseau utilisée par le domaine `primary` (il s'agit `net0` dans cet exemple).**

```
# ipadm show-addr
ADDROBJ     TYPE     STATE     ADDR
lo0/v4         static     ok     127.0.0.1/8
net0/ipv4     static     ok     10.6.212.149/23
```

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

La commande suivante utilise l'interface réseau `nxge0` :

```
primary# ls -l /dev/nxge0
lrwxrwxrwx   1 root     root          46 Jul 30 17:29 /dev/nxge0 ->
../devices/pci@500/pci@0/pci@8/network@0:nxge0
```

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`.

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 redémarrages.

```
# ldm rm-io PCIE2 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.
# ldm rm-io 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.
--------------------------------------------------------------------------------
# ldm rm-io PCIE4 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.
--------------------------------------------------------------------------------
# ldm rm-io 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` :

```
# ldm add-config dio
```

c. Redémarrez le système pour prendre en compte la suppression des périphériques d'extrémité PCIe.

```
# reboot -- -r
```

7 Connectez-vous au domaine `primary` et vérifiez que les périphériques d'extrémité PCIe ne sont plus assignés au domaine.

```
# ldm list-io
NAME                                      TYPE   DOMAIN   STATUS
----                                      ----   ------   ------
pci_0                                     BUS    primary
pci_1                                     BUS    primary
PCIE1                                     PCIE   -        EMP
PCIE2                                     PCIE            OCC
PCIE3                                     PCIE            OCC
MB/SASHBA                                 PCIE   primary  OCC
PCIE0                                     PCIE   -        EMP
PCIE4                                     PCIE            OCC
PCIE5                                     PCIE            OCC
MB/NET0                                   PCIE   primary  OCC
```

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 `PCI2` au domaine `ldg1`.

```
# ldm add-io PCIE2 ldg1
```

b. Associez et démarrez le domaine `ldg1`.

```
# ldm bind ldg1
# 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.

- **Oracle Solaris 10. Exécutez la commande ci-dessous :**

```
# dladm show-dev
vnet0           link: up        speed: 0     Mbps       duplex: unknown
nxge0           link: unknown   speed: 0     Mbps       duplex: unknown
nxge1           link: unknown   speed: 0     Mbps       duplex: unknown
nxge2           link: unknown   speed: 0     Mbps       duplex: unknown
nxge3           link: unknown   speed: 0     Mbps       duplex: unknown
```

- **Oracle Solaris 11. Exécutez la commande ci-dessous :**

```
# dladm show-phys
LINK              MEDIA               STATE      SPEED  DUPLEX    DEVICE
net0              Ethernet            up         0      unknown   vnet0
net1              Ethernet            unknown    0      unknown   vsw0
net2              Ethernet            unknown    0      unknown   nxge0
```

Utilisation des fonctions virtuelles SR-IOV PCIe

A partir de la version 2.2 d'Oracle VM Server for SPARC, la fonction SR-IOV (virtualisation d'E/S à racine unique) PCIe (Peripheral Component Interconnect Express) est prise en charge par les plates-formes SPARC T3 et SPARC T4.

Présentation de SR-IOV

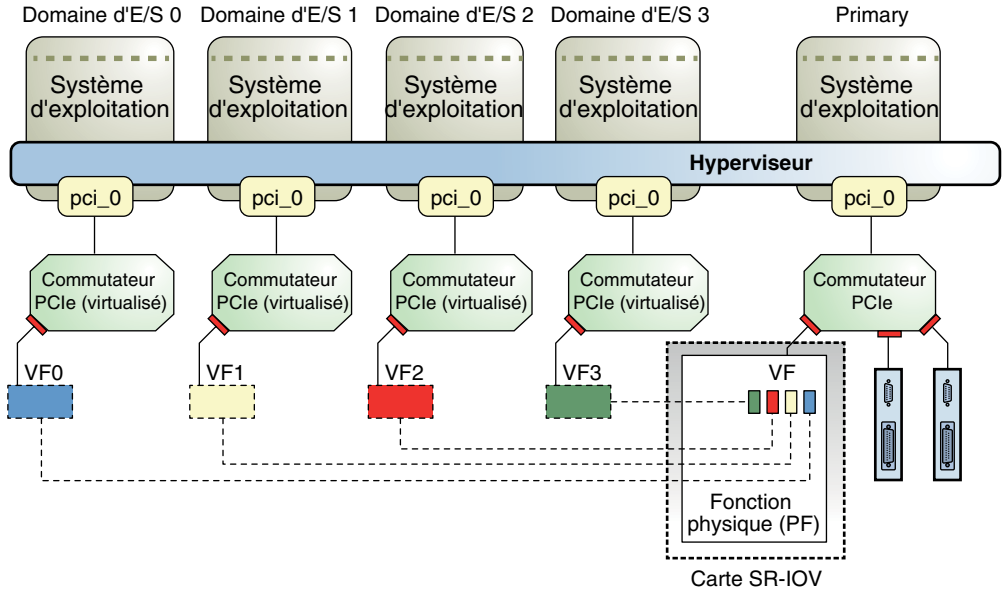
L'implémentation SR-IOV est basée sur la version 1.1 de la norme telle que définie par le consortium PCI-SIG. La norme SR-IOV permet le partage efficace de périphériques PCIe entre des machines virtuelles et est implémenté dans le matériel afin d'atteindre des performances d'E/S comparables à des performances 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 6–3 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 64 000 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.

Les périphériques dotés de fonctionnalités SR-IOV présentent 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

Configuration matérielle et logicielle requise pour SR-IOV

A partir de la version 2.2 d'Oracle VM Server for SPARC, la fonction SR-IOV PCIe est prise en charge par les plates-formes SPARC T3 et SPARC T4. Pour plus d'informations sur les versions requises du matériel, des logiciels et du microprogramme, reportez-vous à la section "Configuration matérielle et logicielle SR-IOV PCIe" du manuel *Notes de version d'Oracle VM Server for SPARC 2.2*.

Restrictions actuelles de la fonction SR-IOV

Remarque – Pour réduire le nombre de réinitialisations au strict minimum, réalisez plusieurs opérations au cours d'une même reconfiguration retardée.

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

- La migration est désactivée pour tout domaine auquel une ou plusieurs fonctions virtuelles sont affectées.
- La création et la destruction de fonctions virtuelles déclenchent une reconfiguration retardée.
- Vous ne pouvez pas assigner une fonction virtuelle à un domaine actif. Une reconfiguration retardée est initiée lorsque vous affectez une fonction virtuelle au domaine `primary`.
- 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.

- Seules les cartes SR-IOV Ethernet sont prises en charge.
- La fonction SR-IOV est uniquement activée pour les cartes SR-IOV installées sur le domaine `primary`. Si une carte SR-IOV est assignée à un domaine par le biais de l'affectation de bus PCIe ou la fonction d'E/S directes (DIO), la fonction SR-IOV n'est pas activée pour cette carte.
- 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.

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

Avant de créer et de détruire des fonctions virtuelles, planifiez à l'avance les fonctions virtuelles que vous souhaitez utiliser dans votre configuration. Lors de la création et de la destruction de fonctions virtuelles, le domaine `primary` doit être réinitialisé. Cette réinitialisation a une incidence négative sur tous les domaines d'E/S comportant des extrémités PCIe ou sur lesquels les fonctions virtuelles SR-IOV sont configurées. Il est donc important de réduire au strict minimum le nombre de réinitialisations du domaine `primary`. Déterminez le nombre de fonctions virtuelles dont vous avez besoin à partir des différents périphériques SR-IOV pour satisfaire vos besoins de configuration actuels et à venir.

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 80.

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.
`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 maximum 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. Reportez-vous à la section "Rubriques SR-IOV avancées" à la page 112.

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

Utilisez la commande suivante pour créer la fonction virtuelle :

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

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 les propriétés spécifiques au réseau `mac-addr`, `alt-mac-addrs` et `mtu` peuvent être modifiées en procédant comme suit :

- **Lorsque la fonction virtuelle est assignée au domaine** `primary` : une demande de modification de propriété déclenche une reconfiguration retardée.
- **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 autre que le domaine** `primary` **et qu'une reconfiguration retardée est déjà en vigueur** : une demande de modification de propriété échoue avec affichage d'un message d'erreur.

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

La création d'une fonction virtuelle étant susceptible de déclencher une reconfiguration retardée, vous pouvez créer d'autres fonctions virtuelles et n'effectuer qu'une seule réinitialisation du domaine `primary` afin d'appliquer les modifications. Vous n'avez pas besoin de réinitialiser le domaine `primary` après la création de chaque fonction virtuelle.

Une fonction physique SR-IOV donnée peut prendre en charge un grand nombre de fonctions virtuelles. Créez *uniquement* les fonctions virtuelles dont vous avez besoin. Pour connaître le nombre de configurations maximal recommandé, reportez-vous à la section “Rubriques SR-IOV avancées” à la page 112.

3. Utilisez la commande `ldm add-config` pour enregistrer la configuration sur le processeur de service.
4. Réinitialisez le domaine `primary` pour créer la fonction virtuelle.
5. Vous devez arrêter un domaine actif avant de lui affecter une fonction virtuelle à l'aide de la commande `ldm add-io`. Réduisez au minimum l'indisponibilité des domaines d'E/S en effectuant simultanément toutes les modifications de domaines d'E/S. En procédant ainsi, vous réduisez le nombre de réinitialisations du domaine `primary` requises pour définir de telles configurations.
6. Initialisez les domaines d'E/S et configurez les fonctions virtuelles comme s'il s'agissait de n'importe quel autre périphérique réseau.

   Pour plus d'informations sur les restrictions des fonctions virtuelles, reportez-vous à la section “Rubriques SR-IOV avancées” à la page 112.

Création, modification et destruction de fonctions virtuelles

Cette section décrit la création, la modification et la destruction de fonctions virtuelles.

▼ Procédure de création d'une fonction virtuelle

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 Créez une fonction virtuelle à partir d'une fonction physique.

```
primary# 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 – Pour les périphériques réseau, l'adresse MAC est automatiquement attribuée.

Vous pouvez utiliser soit le nom du chemin d'accès, soit le pseudonyme pour indiquer des fonctions virtuelles. Toutefois, il est recommandé d'utiliser le pseudonyme.

Exemple 6–1 Création d'une fonction virtuelle

L'exemple suivant montre 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   DOMAIN   STATUS
----                                    ----   ------   ------
pci_0                                   BUS    primary
niu_0                                   NIU    primary
/SYS/MB/RISER0/PCIE0                    PCIE   -        EMP
/SYS/MB/RISER1/PCIE1                    PCIE   -        EMP
/SYS/MB/RISER2/PCIE2                    PCIE   -        EMP
/SYS/MB/RISER0/PCIE3                    PCIE   -        EMP
/SYS/MB/RISER1/PCIE4                    PCIE   primary  OCC
/SYS/MB/RISER2/PCIE5                    PCIE   primary  OCC
/SYS/MB/SASHBA0                         PCIE   primary  OCC
/SYS/MB/SASHBA1                         PCIE   primary  OCC
/SYS/MB/NET0                            PCIE   primary  OCC
/SYS/MB/NET2                            PCIE   primary  OCC
/SYS/MB/RISER1/PCIE4/IOVNET.PF0         PF     -
/SYS/MB/RISER1/PCIE4/IOVNET.PF1         PF     -
```

```
/SYS/MB/RISER2/PCIE5/P0/P2/IOVNET.PF0     PF     -
/SYS/MB/RISER2/PCIE5/P0/P2/IOVNET.PF1     PF     -
/SYS/MB/RISER2/PCIE5/P0/P4/IOVNET.PF0     PF     -
/SYS/MB/RISER2/PCIE5/P0/P4/IOVNET.PF1     PF     -
/SYS/MB/NET0/IOVNET.PF0                   PF     -
/SYS/MB/NET0/IOVNET.PF1                   PF     -
/SYS/MB/NET2/IOVNET.PF0                   PF     -
/SYS/MB/NET2/IOVNET.PF1                   PF     -
```

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   DOMAIN   STATUS
----                                      ----   ------   ------
/SYS/MB/NET0/IOVNET.PF0                   PF     -
[pci@400/pci@2/pci@0/pci@6/network@0]
    maxvfs = 7
```

Les exemples suivants indiquent comment créer une fonction virtuelle :

- Cet exemple crée une fonction virtuelle sans définir de propriété facultative. Dans ce cas, l'adresse MAC pour une fonction virtuelle de classe réseau est automatiquement attribuée.

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

  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.
  ```

- Cet exemple crée une fonction virtuelle 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
  ```

- Cet exemple crée une fonction virtuelle disposant de 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
  ```

▼ Procédure de modification d'une fonction virtuelle

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" à la page 112.

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

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

  Pour modifier ces propriétés de fonction, arrêtez tout d'abord le domaine propriétaire de la fonction virtuelle. Si la fonction virtuelle correspondante est assignée au domaine `primary`, il en résulte une reconfiguration retardée, et une réinitialisation est nécessaire pour que les modifications prennent effet.

- `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. Plus précisément, la définition de la propriété `pvid` active 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. Une modification de ces propriétés entraîne une reconfiguration retardée, et une réinitialisation du domaine `primary` est nécessaire pour que les modifications prennent effet. Pour plus d'informations sur les propriétés spécifiques aux périphériques, reportez-vous à la section "Rubriques SR-IOV avancées" à la page 112.

- **Modifiez une fonction virtuelle.**

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

Exemple 6–2 Modification d'une fonction virtuelle

Les exemples suivants illustrent l'utilisation de la commande `ldm set-io` pour définir les propriétés d'une fonction virtuelle :

- L'exemple suivant modifie la fonction virtuelle indiquée, `/SYS/MB/NET0/IOVNET.PF0.VF0`, et spécifie qu'elle fait partie des VLAN d'ID 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

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.
```

▼ Procédure de destruction d'une fonction virtuelle

Vous pouvez détruire une fonction virtuelle si elle n'est pas affectée à un domaine. Notez également que seule la dernière fonction virtuelle créée peuvent être détruite. La configuration résultante est validée par le pilote de la fonction physique. Lorsque l'opération est effectuée correctement, une reconfiguration retardée est déclenchée et une réinitialisation est nécessaire pour que les modifications prennent effet.

- **Détruisez une fonction virtuelle.**

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

Exemple 6–3 Destruction d'une fonction virtuelle.

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

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

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.
```

Ajout et suppression de fonctions virtuelles sur des domaines d'E/S

▼ Procédure d'ajout d'une fonction virtuelle à un domaine d'E/S

La commande suivante ajoute une fonction virtuelle à un domaine logique :

```
ldm add-io vf-name ldom
```

vf-name est le pseudonyme du nom du chemin d'accès de la fonction virtuelle. Il est préférable d'utiliser le pseudonyme. *ldom* représente le nom du domaine auquel vous ajoutez la fonction virtuelle. L'invité spécifié doit avoir l'état inactif ou lié. Si vous spécifiez le domaine `primary`, cette commande déclenche une reconfiguration retardée.

- **Ajoutez une fonction virtuelle.**

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

  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`.

Exemple 6–4 Ajout d'une fonction virtuelle.

L'exemple suivant illustre l'ajout de la fonction virtuelle `/SYS/MB/NET0/IOVNET.PF0.VF0` au domaine `ldg1`. Pour que la commande aboutisse, il faut que le domaine spécifié ait l'état inactif ou lié. S'il s'agit du domaine `primary`, une reconfiguration retardée est déclenchée.

```
primary# ldm add-io /SYS/MB/NET0/IOVNET.PF0.VF0 ldg1
```

Si la commande s'exécute correctement, la fonction virtuelle est ajoutée au domaine `ldg1`. Si le domaine `ldg1` est déjà associé (ou s'il est associé par la suite), le domaine peut être réinitialisé et le SE invité peut utiliser la fonction virtuelle ajoutée pour les opérations d'E/S.

▼ Procédure de suppression d'une fonction virtuelle d'un domaine d'E/S

La commande suivante supprime une fonction virtuelle SR-IOV d'un domaine logique :

```
ldm remove-io vf-name ldom
```

vf-name est le pseudonyme du nom du chemin d'accès de la fonction virtuelle. Il est préférable d'utiliser le pseudonyme du périphérique. *ldom* représente le nom du domaine duquel vous supprimez la fonction virtuelle. L'invité spécifié doit avoir l'état inactif ou lié. Si vous spécifiez le domaine `primary`, cette commande déclenche une reconfiguration retardée.

Remarque – Avant de supprimer la fonction virtuelle du domaine, assurez-vous qu'elle n'est pas indispensable à l'initialisation du domaine.

- **Supprimez une fonction virtuelle.**

  ```
  primary# ldm rm-io vf-name ldom
  ```

Exemple 6–5 Suppression d'une fonction virtuelle

L'exemple suivant illustre la suppression de la fonction virtuelle `/SYS/MB/NET0/IOVNET.PF0.VF0` du domaine `ldg1` :

```
primary# ldm rm-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 le domaine comprenant la fonction virtuelle est le domaine `primary`, une reconfiguration retardée est déclenchée.

SR-IOV : réinitialisation du domaine primary

Faites très attention lorsque vous réinitialisez le domaine `primary`. Reportez-vous à la section "Redémarrage du domaine `primary`" à la page 91 Comme pour les emplacements PCIe dans le domaine d'E/S, les points évoqués dans cette section s'appliquent également aux fonctions virtuelles assignées à un domaine d'E/S.

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 `primary` pour réduire les indisponibilités.

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. **Obtenez des informations détaillées sur la fonction physique que vous souhaitez partager.**

   ```
   primary# ldm list-io -l pf-name
   ```

3. **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. Si vous exécutez ces commandes en batch, vous ne devez réinitialiser qu'une seule fois le domaine `primary`.

4. **Arrêtez tous les domaines d'E/S auxquels une fonction virtuelle ou un périphérique d'extrémité PCIe est affecté.**

   ```
   primary# ldm stop ldom
   ```

5. **Redémarrez le domaine `primary`.**

   ```
   primary# reboot
   ```

6. **Affichez la liste des fonctions virtuelles disponibles sur le domaine `primary`.**

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

7. **Affectez la fonction virtuelle à un domaine.**

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

8. **Associez et démarrez le domaine.**

   ```
   primary# ldm bind ldom
   primary# ldm start ldom
   ```

9. **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 6–6 Création d'un domaine d'E/S par affectation d'une fonction virtuelle SR-IOV

L'exemple suivant décrit 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`.

La sortie `ldm list-io` suivante indique que la fonction physique `/SYS/MB/NET0/IOVNET.PF0` est disponible :

```
primary# ldm list-io
NAME                                      TYPE   DOMAIN   STATUS
----                                      ----   ------   ------
pci_0                                     BUS    primary
niu_0                                     NIU    primary
/SYS/MB/RISER0/PCIE0                      PCIE   -        EMP
/SYS/MB/RISER1/PCIE1                      PCIE   -        EMP
/SYS/MB/RISER2/PCIE2                      PCIE   -        EMP
/SYS/MB/RISER0/PCIE3                      PCIE   -        EMP
/SYS/MB/RISER1/PCIE4                      PCIE   primary  OCC
/SYS/MB/RISER2/PCIE5                      PCIE   primary  OCC
/SYS/MB/SASHBA0                           PCIE   primary  OCC
/SYS/MB/SASHBA1                           PCIE   primary  OCC
/SYS/MB/NET0                              PCIE   primary  OCC
/SYS/MB/NET2                              PCIE   primary  OCC
/SYS/MB/RISER1/PCIE4/IOVNET.PF0           PF     -
/SYS/MB/RISER1/PCIE4/IOVNET.PF1           PF     -
/SYS/MB/RISER2/PCIE5/P0/P2/IOVNET.PF0     PF     -
/SYS/MB/RISER2/PCIE5/P0/P2/IOVNET.PF1     PF     -
/SYS/MB/RISER2/PCIE5/P0/P4/IOVNET.PF0     PF     -
/SYS/MB/RISER2/PCIE5/P0/P4/IOVNET.PF1     PF     -
/SYS/MB/NET0/IOVNET.PF0                   PF     -
/SYS/MB/NET0/IOVNET.PF1                   PF     -
/SYS/MB/NET2/IOVNET.PF0                   PF     -
/SYS/MB/NET2/IOVNET.PF1                   PF     -
```

La commande suivante affiche des détails supplémentaires sur la fonction physique `/SYS/MB/NET0/IOVNET.PF0`, notamment le nombre maximal de fonctions virtuelles pouvant être créées :

```
primary# ldm list-io -l /SYS/MB/NET0/IOVNET.PF0
NAME                                      TYPE   DOMAIN   STATUS
----                                      ----   ------   ------
/SYS/MB/NET0/IOVNET.PF0                   PF     -
[pci@400/pci@2/pci@0/pci@6/network@0]
    maxvfs = 7
```

La commande suivante crée une fonction virtuelle appelée `/SYS/MB/NET0/IOVNET.PF0.VF0` pour la fonction physique `/SYS/MB/NET0/IOVNET.PF0` :

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

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

Etant donné que le domaine d'E/S `ldg1` comporte un périphérique d'extrémité PCIe créé à l'aide de la fonction DIO, vous devez arrêter le domaine `ldg1` et réinitialiser le domaine `primary` en procédant comme suit :

```
primary# ldm stop ldg1
primary# reboot
```

La commande suivante permet de vérifier que la nouvelle fonction virtuelle `/SYS/MB/NET0/IOVNET.PF0.VF0` existe :

```
primary# ldm list-io
NAME                                    TYPE   DOMAIN   STATUS
----                                    ----   ------   ------
pci_0                                   BUS    primary
niu_0                                   NIU    primary
/SYS/MB/RISER0/PCIE0                    PCIE   -        EMP
/SYS/MB/RISER1/PCIE1                    PCIE   -        EMP
/SYS/MB/RISER2/PCIE2                    PCIE   -        EMP
/SYS/MB/RISER0/PCIE3                    PCIE   -        EMP
/SYS/MB/RISER1/PCIE4                    PCIE   primary  OCC
/SYS/MB/RISER2/PCIE5                    PCIE   primary  OCC
/SYS/MB/SASHBA0                         PCIE   primary  OCC
/SYS/MB/SASHBA1                         PCIE   primary  OCC
/SYS/MB/NET0                            PCIE   primary  OCC
/SYS/MB/NET2                            PCIE   primary  OCC
/SYS/MB/RISER1/PCIE4/IOVNET.PF0         PF     -
/SYS/MB/RISER1/PCIE4/IOVNET.PF1         PF     -
/SYS/MB/RISER2/PCIE5/P0/P2/IOVNET.PF0   PF     -
/SYS/MB/RISER2/PCIE5/P0/P2/IOVNET.PF1   PF     -
/SYS/MB/RISER2/PCIE5/P0/P4/IOVNET.PF0   PF     -
/SYS/MB/RISER2/PCIE5/P0/P4/IOVNET.PF1   PF     -
/SYS/MB/NET0/IOVNET.PF0                 PF     -
/SYS/MB/NET0/IOVNET.PF1                 PF     -
/SYS/MB/NET2/IOVNET.PF0                 PF     -
/SYS/MB/NET2/IOVNET.PF1                 PF     -
/SYS/MB/NET0/IOVNET.PF0.VF0             VF
```

La commande suivante ajoute la fonction virtuelle `/SYS/MB/NET0/IOVNET.PF0.VF0` au domaine `ldg1` :

```
primary# ldm add-io /SYS/MB/NET0/IOVNET.PF0.VF0 ldg1
```

Les commandes suivantes associent et redémarrent le domaine `ldg1` :

```
primary# ldm bind ldg1
primary# ldm start ldg1
```

La commande suivante vérifie que la fonction virtuelle est disponible et peut être utilisée :

```
guest# dladm show-phys
LINK              MEDIA               STATE      SPEED  DUPLEX    DEVICE
net0              Ethernet            up         0      unknown   vnet0
net1              Ethernet            up         1000   full      igbvf0
```

Rubriques SR-IOV avancées

Cette section décrit plusieurs éléments à prendre en compte lorsque vous utilisez des périphériques prenant en charge la fonction SR-IOV PCIe.

- **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 fonctionnalités 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 – Lors de l'initialisation du SE Oracle Solaris à partir d'une fonction virtuelle, assurez-vous que le SE Oracle Solaris en cours de chargement prend en charge les périphériques de fonction virtuelle. Si tel est le cas, vous pouvez poursuivre l'installation comme prévu.

- **Nombre maximal de domaines d'E/S et de fonctions virtuelles pris en charge.** Les périphériques d'extrémité PCIe et les fonctions virtuelles SR-IOV d'un bus PCIe donné peuvent être affectés à 15 domaines au maximum. Les ressources PCIe telles que les vecteurs d'interruption de chaque bus PCIe sont réparties entre le domaine racine 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 affecter un grand nombre de fonctions virtuelles à un même domaine d'E/S. Pour une description des problèmes liés à SR-IOV, reportez-vous au manuel *Notes de version d'Oracle VM Server for SPARC 2.2*.

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. Chaque propriété possède un nom, une brève description, 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 la définition d'une propriété spécifique au périphérique déclenche une reconfiguration retardée.

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
```

Configuration réseau avancée pour les fonctions virtuelles

- 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 liens 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 liens, 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 des fonctions virtuelles à partir de ce même périphérique de fonction physique.

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 donc 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 et initialise le SE Oracle Solaris 11 dans le domaine d'E/S. Dans cet exemple, `ldg1` est 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 créer quatre 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 show-phys
LINK              MEDIA                STATE      SPEED  DUPLEX    DEVICE
net0              Ethernet             up         0      unknown   vnet0
net1              Ethernet             up         1000   full      igbvf0
guest# dladm create-vnic -l net1 vnic0
guest# dladm create-vnic -l net1 vnic1
guest# dladm create-vnic -l net1 vnic2
guest# dladm create-vnic -l net1 vnic3
guest# dladm show-link
LINK                CLASS     MTU    STATE    OVER
net0                phys      1500   up       --
net1                phys      1500   up       --
vnic0               vnic      1500   up       net1
vnic1               vnic      1500   up       net1
vnic2               vnic      1500   up       net1
vnic3               vnic      1500   up       net1
```

◆ ◆ ◆ CHAPITRE 7

7 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 se termine. 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.

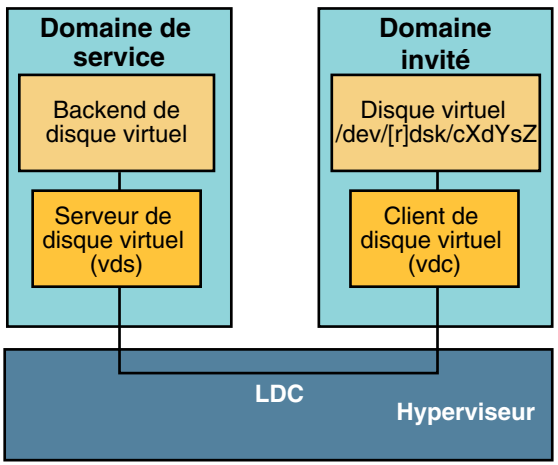
Le backend du disque virtuel peut être physique ou logique. Les périphériques physiques peuvent comprendre :

- Un disque physique ou un numéro d'unité logique (LUN)
- Une tranche de disque physique

Les périphériques logiques peuvent être les suivants :

- Fichier sur un système de fichiers, notamment ZFS ou UFS
- Volume logique à partir d'un gestionnaire de volumes, notamment ZFS, VxVM ou Solaris Volume Manager
- Tout pseudo-périphérique de disque accessible à partir du domaine de service

FIGURE 7–1 Disques virtuels avec Logical Domains



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 120 pour obtenir une description des options du disque virtuel. Reportez-vous à la section "Délai d'attente du disque virtuel" à la page 135 pour obtenir une description du délai d'attente du disque virtuel.

▼ Procédure d'ajout d'un disque virtuel

1 **Exportez le backend du disque virtuel à partir d'un domaine de service.**

```
# ldm add-vdsdev [-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 ldom
```

Vous pouvez indiquer un ID 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 119.

Remarque – Un backend est généralement exporté à partir du domaine de service et assigné au domaine invité lorsque ce dernier (*ldom*) est lié.

▼ Procédure d'exportation multiple du backend d'un 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`.

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 à l'aide des commandes suivantes.**

```
# ldm add-vdsdev [options={ro,slice}] backend volume1@service-name
# ldm add-vdsdev -f [options={ro,slice}] backend volume2@service-name
```

Notez que la seconde commande `ldm add-vdsdev` 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 à l'aide des commandes suivantes.**

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 120.

- **Après l'exportation d'un backend à partir du domaine de service, vous pouvez modifier les options du disque virtuel à l'aide de la commande suivante.**

```
# 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 120.

- **Après l'assignation d'un disque virtuel à un domaine invité, vous pouvez modifier le délai d'attente du disque virtuel à l'aide de la commande suivante.**

```
# ldm set-vdisk timeout=seconds disk-name ldom
```

▼ Procédure de suppression d'un disque virtuel

1 **Supprimez un disque virtuel d'un domaine invité à l'aide de la commande suivante.**

```
# ldm rm-vdisk disk-name ldom
```

2 **Arrêtez l'exportation du backend correspondant à partir du domaine de service à l'aide de la commande suivante.**

```
# ldm rm-vdsdev volume-name@service-name
```

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 ldom
```

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.

Apparence du 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.

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`). Un tel 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. Un tel 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`). Un tel 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 la commande `ldm add-vdsdev` sous forme de liste séparée 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 échouera.

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. Cela empêche les applications s'exécutant sur le domaine de service d'utiliser par inadvertance le backend étant 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 peut fonctionner ou non avec les pseudo-périphériques, 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` ou `luxadm` 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. Reportez-vous à la section "Utilisation de disques virtuels au sommet de Solaris Virtual Manager" à la page 143 pour plus d'informations.
- 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 143 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 117 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. Notez qu'il s'agit d'un comportement nouveau à partir de la version du SE Oracle Solaris 10 5/08. Avant la version du SE Oracle Solaris 10 5/08, les moteurs de traitement de disque s'ouvraient toujours en exclusivité, et il était impossible de les ouvrir autrement.

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 122.

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

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.

▼ 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-vdsdev /dev/dsk/c1t57d0s0 c1t57d0s0@primary-vds0
```

Il est inutile de spécifier l'option `slice`, car une tranche est toujours exportée comme un 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 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 (disque `c1t57d0s2`, par exemple), vous devez spécifier l'option `slice`. Sinon, tout le disque est exporté.**

```
# ldm add-vdsdev options=slice /dev/dsk/c1t57d0s2 c1t57d0s2@primary-vds0
```

Fichier et volume

Un fichier ou un volume (par exemple provenant de ZFS ou Solaris Volume Manager) est exporté soit comme un disque complet, soit comme un disque à tranche unique selon que vous définissez 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 – Avant la version du SE Oracle Solaris 5/08, si un fichier vide était exporté en tant que disque virtuel, le système écrivait une étiquette de disque par défaut et créait un partitionnement par défaut. Ce n'est plus la cas avec la version du SE Oracle Solaris 5/08, et vous devez exécuter `format`(1M)dans le domaine invité pour créer les 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 méga-octets pour obtenir un disque virtuel de 100 méga-octets.

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 8 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 vide de 100 méga-octets pour obtenir un disque virtuel de 100 méga-octets.

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

Avant la version du SE Oracle Solaris 10 5/08, l'option `slice` n'existait pas et les volumes étaient exportés en tant que disques à tranche unique. Si vous avez une configuration exportant les volumes en tant que disques virtuels et si vous mettez à niveau le système vers le SE Oracle Solaris 10 5/08, les volumes sont maintenant exportés comme des disques complets plutôt que 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 .2.2 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)` pour plus d'informations sur cette commande.
- Ajoutez la ligne suivante au fichier `/etc/system` sur le domaine de service.

  ```
  set vds:vd_volume_force_slice = 1
  ```

  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[1]	Disque à tranche unique[2]
Tranche de disque (pas la tranche 2)	Disque à tranche unique[3]	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

[1] Exportez tout le disque.

[2] Exportez uniquement la tranche 2.

[3] 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)

Il est possible d'utiliser le pilote de fichier loopback (`lofi`) pour exporter un fichier en tant que disque virtuel. Cependant, cette opération ajoute une couche de pilote supplémentaire et a une incidence 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 "Fichier et volume" à la page 124.

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

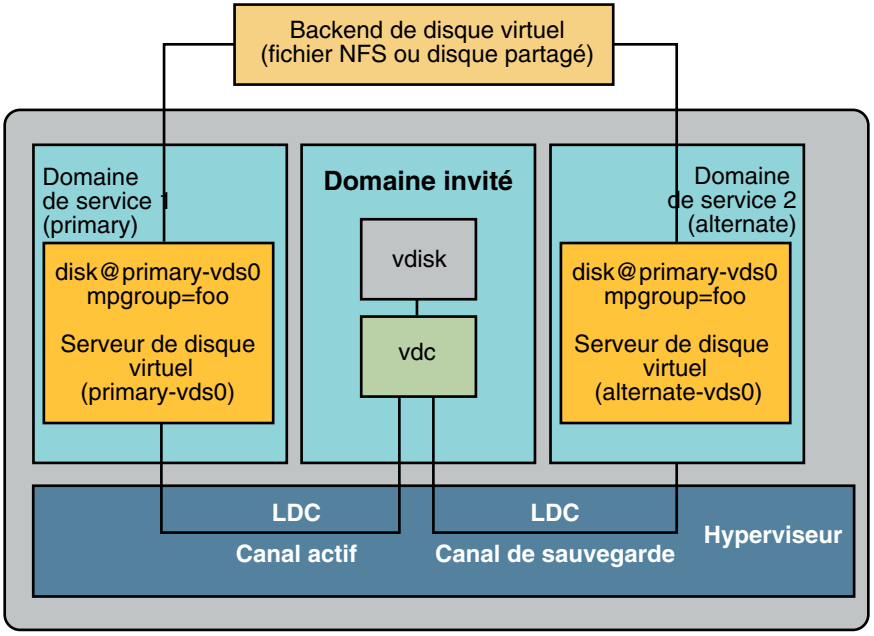
La fonctionnalité de chemins d'accès multiples de disque virtuel (fonctionnalité multipathing) vous permet de configurer un disque virtuel sur un domaine invité pour accéder à son stockage backend par plusieurs chemins. Les chemins traversent différents domaines de service qui fournissent l'accès au 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 – A partir de la version Oracle VM Server for SPARC 2.0, la fonctionnalité multipathing 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 131. Dans cet exemple, un groupe multipathing nommé `foo` est utilisé pour créer un disque virtuel, dont le backend est accessible à partir de deux domaines de service : `primary` et `alternate`.

FIGURE 7–2 Configuration de la fonctionnalité de chemins d'accès multiples d'un disque virtuel



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 lorsque la fonctionnalité multipathing de disque virtuel est configurée peut empêcher le fonctionnement correct de cette fonctionnalité, en particulier lorsque la valeur du délai d'attente 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 135.

▼ Procédure de configuration de la fonctionnalité multipathing de disque virtuel

1. **Exportez le backend du disque virtuel à partir du domaine de service `primary`.**

   `# ldm add-vdsdev mpgroup=foo` *backend-path1 volume*`@primary-vds0`

   où *backend-path1* est le chemin vers le backend du disque virtuel à partir du domaine `primary`.

2. **Exportez le même backend du disque virtuel à partir du domaine de service `alternate`.**

   `# ldm add-vdsdev mpgroup=foo` *backend-path2 volume*`@alternate-vds0`

   où *backend-path2* est le chemin vers le 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.**

   `# ldm add-vdisk` *disk-name volume*`@primary-vds0` *ldom*

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.

Informations supplémentaires

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 le domaine de service auquel il a été associé (le domaine `primary` dans cet exemple). 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 backens 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 corrompre les données stockées sur les moteurs de traitement.

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 le lecteur de CD ou de DVD lui-même. Vous pouvez uniquement exporter le CD ou le DVD se trouvant dans le lecteur. Par conséquent, un CD ou un DVD doit se trouver dans le lecteur 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 133 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 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 démarrer 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 démarrez 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 Effectuez l'une des opérations suivantes.

- Si le volume de gestion des volumes n'est pas en cours d'exécution ou en ligne, passez à l'étape 3.
- 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 commentez 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 le lecteur 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.**

Voici comment assigner le CD ou le DVD exporté au domaine `ldg1` :

```
primary# ldm add-vdisk cdrom cdrom@primary-vds0 ldg1
```

Informations supplémentaires

Exportation multiple d'un CD ou d'un DVD

Un CD ou un DVD peut être exporté plusieurs fois et assigné à des domaines invités différents. Reportez-vous à la section "Procédure d'exportation multiple du backend d'un disque virtuel" à la page 117 pour plus d'informations.

▼ Procédure d'exportation d'une image ISO à partir du domaine primary pour installation sur un domaine invité

Cette procédure montre comment exporter une image ISO à partir du domaine `primary` et l'installer sur un domaine invité. 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 :

```
# ldm list
NAME             STATE    FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active   -n-cv   SP      4     4G       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`.

```
# 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`.

```
# 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`.

```
# ldm add-vdisk s10-dvd dvd-iso@primary-vds0 ldom1
```

4. **Redémarrez le domaine invité.**

   Dans cet exemple, le domaine logique est `ldom1`.

```
# ldm start-domain ldom1
LDom ldom1 started
# ldm list
NAME             STATE    FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active   -n-cv   SP      4     4G       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.**

```
# 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. **Démarrez le domaine invité pour effectuer l'installation à partir de l'image ISO.**

   Dans cet exemple, démarrez 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 échoue et signale une erreur lorsque le domaine de service est en panne pour une longue période. Il est maintenant possible de 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.

Ce délai d'attente peut être défini en procédant comme suit :

- A l'aide de la commande `ldm add-vdisk`.

  `ldm add-vdisk timeout=`*seconds disk-name volume-name@service-name ldom*

- A l'aide de la commande `ldm set-vdisk`.

  `ldm set-vdisk timeout=`*seconds disk-name ldom*

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).

Sinon, le délai d'attente peut être défini en ajoutant la ligne suivante au fichier `/etc/system` sur le domaine invité.

```
set vdc:vdc_timeout=seconds
```

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.

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.

En conséquence, les applications ou fonctions de produit utilisant les commandes SCSI (notamment `metaset` de Solaris Volume Manager ou `shared devices` d'cluster Oracle Solaris) peuvent être utilisés dans les domaines invités uniquement avec les disques virtuels ayant 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. Les réservations SCSI sont effectuées par 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. Sinon, cela peut aboutir à un état de disque inattendu.

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 :

- De stocker des images de disque dans des volumes ZFS ou des fichiers ZFS
- D'utiliser des instantanés pour sauvegarder les images de disque
- d'utiliser des clones pour dupliquer les images de disque et mettre à disposition d'autres domaines.

Reportez-vous au*Guía de administración de Oracle Solaris ZFS* pour plus d'informations sur l'utilisation de ZFS.

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 stockés 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`. La commande peut prendre un certain temps pour se terminer, surtout si le fichier à créer est très volumineux, ce qui est souvent la cas lors de la création d'une image 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 pseudo-périphérique 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 giga-octets pour accueillir tout type d'installation de n'importe quelle version du SE Oracle Solaris.

Exemples de stockage d'images de disque avec ZFS

Les exemples suivants :

1. Créent une image de 20 giga-octets sur un volume ou un fichier ZFS.
2. Exportent le volume ou le fichier ZFS en tant que disque virtuel. La syntaxe pour exporter un volume ou un fichier ZFS est identique, mais le chemin au moteur de traitement est différent.
3. Assignent le volume ou le fichier ZFS exporté à un domaine invité.

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.

▼ Procédure de création d'une image de disque à l'aide d'un volume ZFS

- **Par exemple, créez une image de disque de 20 giga-octets sur un volume ZFS.**

```
primary# zfs create -V 20gb ldmpool/ldg1/disk0
```

▼ Procédure de création d'une image de disque à l'aide d'un fichier ZFS

- **Par exemple, créez une image de disque de 20 giga-octets sur un volume ZFS.**

```
primary# zfs create ldmpool/ldg1/disk0
primary# mkfile 20g /ldmpool/ldg1/disk0/file
```

▼ Procédure d'exportation du volume ZFS

- **Exportez le volume ZFS en tant que disque virtuel.**

```
primary# ldm add-vdsdev /dev/zvol/dsk/ldmpool/ldg1/disk0 ldg1_disk0@primary-vds0
```

▼ Procédure d'exportation du fichier ZFS

- **Exportez le fichier ZFS en tant que disque virtuel.**

```
primary# ldm add-vdsdev /ldmpool/ldg1/disk0/file ldg1_disk0@primary-vds0
```

▼ Procédure d'assignation du volume ou fichier ZFS à un domaine invité

- **Assignez le volume ou le fichier ZFS à un domaine invité, dans cet exemple `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 d'instantané 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. Il y a plusieurs méthodes pour vérifier qu'un disque n'est pas en cours d'utilisation dans un domaine invité. Vous pouvez :

- Arrêter et dissocier le domaine invité. C'est la solution la plus sûre et c'est la seule solution disponible si vous voulez créer un instantané d'une image de disque utilisée comme disque d'initialisation d'un domaine invité.
- Vous pouvez également démonter les tranches du disque dont vous souhaitez créer un instantané dans le domaine invité et vérifier 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.

▼ Procédure de création d'un instantané d'une image de disque

- **Créez un instantané de l'image de disque qui a été créée pour le domaine `ldg1`, par exemple.**

```
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 de l'image de disque, vous pouvez dupliquer cette image de disque à l'aide de la commande de clonage 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 `sys-unconfig`. Pour éviter ce problème, vous pouvez également créer un instantané d'une image de disque du 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. **Exécutez la commande `sys-unconfig`.**
3. **A la fin de la commande `sys-unconfig`, le domaine s'arrête.**
4. **Arrêtez et dissociez le domaine. Ne le redémarrez *pas*.**
5. **Prenez un instantané de l'image du disque d'initialisation du domaine.**

   Par exemple :

   ```
   primary# zfs snapshot ldmpool/ldg1/disk0@unconfigured
   ```

   A ce moment, vous avez l'instantané de l'image du disque d'initialisation d'un système non configuré.

6 **Clonez cette image pour créer un nouveau domaine qui, au premier démarrage, demande la configuration du système.**

Utilisation du gestionnaire de volumes dans un environnement Logical Domains

Cette section décrit l'utilisation des gestionnaires de volumes dans un environnement Logical Domains.

Utilisation de disques virtuels au sommet de gestionnaires de volumes

Tous les volumes ZFS (Zettabyte File System), Solaris Volume Manager ou Veritas Volume Manager (VxVM) peuvent être exportés 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 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 de disques virtuels au sommet de Solaris Virtual 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 `metastart` 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.

```
# 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/c10t600C0FF00000000000015153295A4B100d0s1  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 Veritas Volume Manager (VxVM) est installé sur votre système, et si 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` (valeur l'étant pas celle 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 si Veritas DMP est activé en vérifiant les informations sur la fonctionnalité multipathing dans la sortie de la commande `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 des gestionnaires de volumes au sommet de disques virtuels

Cette section décrit l'utilisation des gestionnaires de volumes au sommet des disques virtuels.

Utilisation de ZFS au sommet de 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 au sommet de 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 au sommet de disques virtuels

Pour la prise en charge de VxVM dans les domaines invités, reportez-vous à la documentation VxVM de Symantec.

◆ ◆ ◆ CHAPITRE 8

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 148, "Présentation de la gestion de réseau dans Oracle Solaris 11" à la page 150 et "Différences liées aux fonctions de gestion réseau d'Oracle Solaris 11" à la page 189.

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.

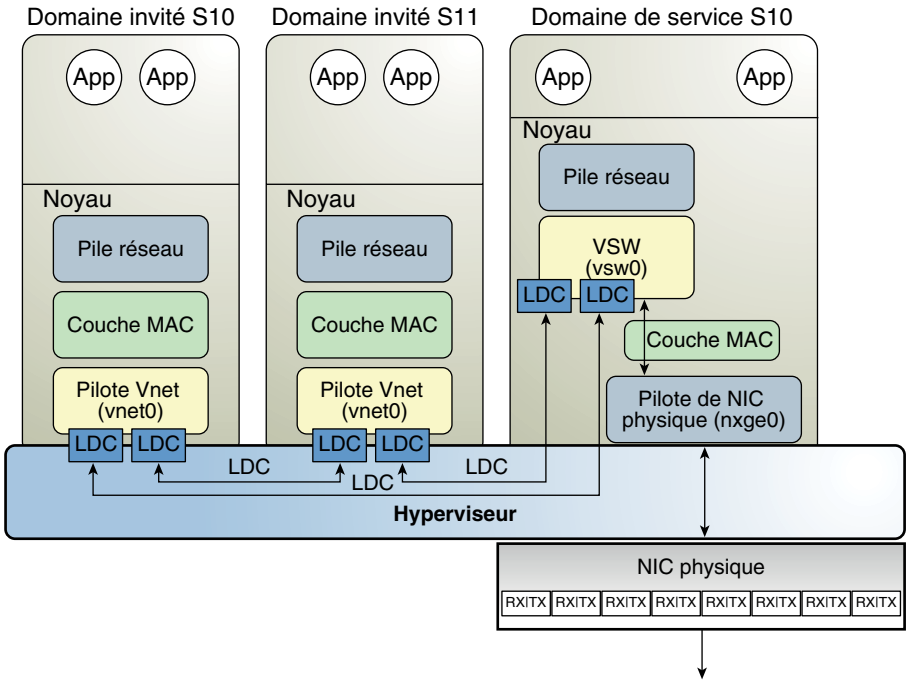
Remarque – La gestion de réseau d'Oracle Solaris 10 fonctionne de la même manière que dans un domaine ou un système. Il en va de même pour la gestion de réseau d'Oracle Solaris 11. Pour plus d'informations sur la gestion de réseau dans le SE Oracle Solaris, reportez-vous à la Documentation Oracle Solaris 10 et à la Documentation Oracle Solaris 11.

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 150.

Présentation de la gestion de réseau dans Oracle Solaris 10

Le diagramme 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 8–1 Présentation du réseau Oracle VM Server for SPARC pour le SE Oracle Solaris 10



Vous trouverez ci-dessous une explication du diagramme précédent dans lequel des noms d'interface tels que `nxge0`, `vsw0` et `vnet0` sont utilisés. Ces noms d'interfaces s'appliquent uniquement au SE Oracle Solaris 10.

- 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.

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 Documentation Oracle Solaris 11.

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.

  Pour identifier le périphérique à utiliser en tant que périphérique backend pour le commutateur virtuel, recherchez `vsw` dans la sortie `dladm show-phys`.

  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.

  Remarque – La création de cartes d'interface réseau virtuelles sur des commutateurs virtuels (`vsw`*n*) ou des périphériques réseau virtuels (`vnet`*n*) ne sont *pas* prises en charge. Reportez-vous à la section “Oracle Solaris 11 : les zones configurées à l'aide d'une interface réseau automatique risquent de ne pas pouvoir démarrer” du manuel *Notes de version d'Oracle VM Server for SPARC 2.2*.

- Utilisez le profil de configuration réseau (NCP) `DefaultFixed` d'Oracle Solaris 11 lors de la configuration du logiciel Oracle VM Server for SPARC.

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 les commandes `dladm` ou `ipadm` pour gérer la mise en réseau.
- `Automatic`**.** Vous permet d'utiliser les commandes `netcfg` ou `netadm` pour gérer la mise en réseau.

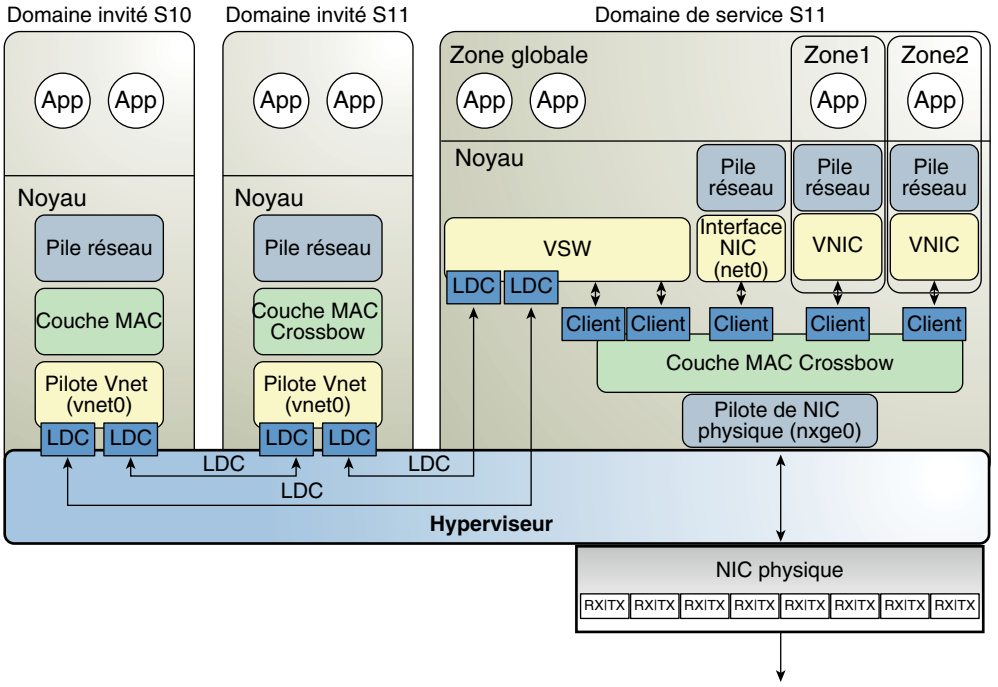
Assurez-vous à l'aide de la commande `netadm list` que le NCP `DefaultFixed` est activé. Reportez-vous au Chapitre 7, "Uso de comandos de configuración de interfaces y enlaces de datos en perfiles" du manuel *Administración de Oracle Solaris: interfaces y virtualización de redes*.

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 contrôle 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.

Le diagramme 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 8–2 Présentation du réseau Oracle VM Server for SPARC pour le SE Oracle Solaris 11



Vous trouverez ci-dessous une explication du diagramme précédent qui 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.

- Le commutateur virtuel du domaine de service est connecté aux domaines invités. Cela permet aux domaines invités de communiquer entre eux.
- Le commutateur virtuel est également connecté au périphérique réseau physique `nxge0`. Cela 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`. Par conséquent, vous n'avez pas besoin de configurer `vsw` en tant qu'interface réseau dans un domaine de service Oracle Solaris 11 en raison des améliorations de la gestion de réseau apportées dans la couche MAC d'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 système auxquels il est connecté. Un système peut être un domaine invité, un domaine de service ou un réseau physique.

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, les communications réseau entre les domaines invités et le réseau physique sont autorisées via l'interface réseau physique. 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 ou la commande `ipadm` d'Oracle Solaris 11.

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 contrôle/service et les autres domaines" à la page 65 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)`.

Les exemples suivants illustrent la procédure de création d'un commutateur virtuel sur un adaptateur réseau physique :

- **Oracle Solaris 10 :** la commande suivante crée un commutateur virtuel sur un adaptateur réseau physique nommé `nxge0` :

  ```
  primary# ldm add-vsw net-dev=nxge0 primary-vsw0 primary
  ```

  Pour plus d'informations sur la configuration d'un commutateur virtuel en tant qu'interface réseau, reportez-vous à la section "Activation de la mise en réseau entre le domaine de contrôle/service et les autres domaines" à la page 65.

- **Oracle Solaris 11 :** la commande suivante crée un commutateur virtuel sur un adaptateur réseau physique nommé `net0` :

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

Périphérique réseau virtuel

Un périphérique réseau virtuel (`vnet`) 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).

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)`.

Reportez-vous aux informations concernant la gestion de réseau Oracle VM Server for SPARC pour Oracle Solaris 10 et Oracle Solaris 11, respectivement à la Figure 8–1 et à la Figure 8–2.

Canaux LDC inter-Vnet

Jusqu'à la version Oracle VM Server for SPARC 2.1, Logical Domains Manager assignait les canaux LDC de la manière suivante :

- Un canal LDC entre les périphérique 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 exponentielle.

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 :

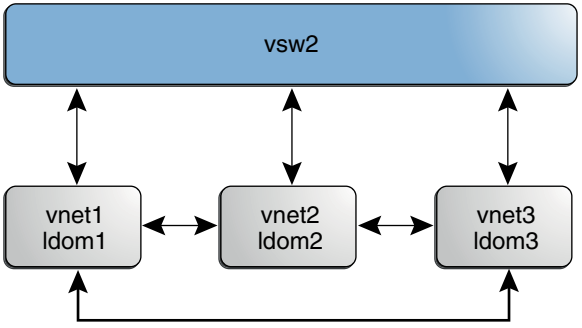
- Lorsque les performances de communication entre invités n'est pas de prime importance
- Lorsqu'un grand nombre de périphériques réseau virtuel sont requis dans un périphérique de commutateur virtuel

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.

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.

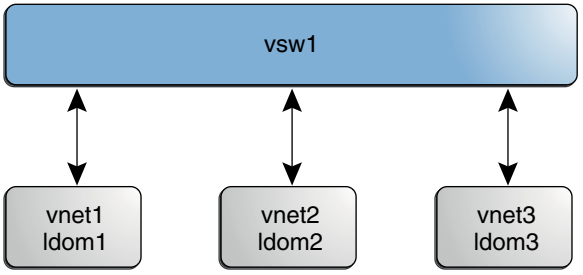
FIGURE 8–3 Configuration du commutateur virtuel utilisant les canaux inter-vnet



La figure suivante présente la même configuration de commutateur virtuel avec la propriété `inter-vnet-link` définie sur `off`. Cela signifie que les canaux LDC inter-vnet ne sont pas alloués. Vous pouvez voir qu'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`.

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 8–4 Configuration de commutateur virtuel n'utilisant pas les canaux inter-vnet



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`.

```
# ldm add-vsw [id=switch-id] vswitch-name ldom
# ldm add-vnet [id=network-id] if-name vswitch-name ldom
```

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 nxge0   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 rompre la configuration existante du système. 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.

▼ Procédure d'identification du nom de l'interface réseau du SE Oracle Solaris

Dans cet exemple de procédure, le domaine invité `ldg1` contient deux périphériques de réseau virtuel, `net-a` et `net-c`. Pour rechercher le nom de l'interface réseau du SE Oracle Solaris dans

`ldg1` qui correspond à `net-c`, procédez comme suit. 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.

1 Utilisez la commande `ldm` pour rechercher le numéro du périphérique réseau virtuel pour `net-c`.

```
# 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 pour ce numéro de périphérique sous `/devices`.

```
# uname -n
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.

- **SE Oracle Solaris 10. Utilisez la commande `ifconfig`.**

  ```
  # 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
  ```

- **SE Oracle Solaris 11. Utilisez la commande `dladm`.**

  Tout d'abord, vous devez déterminer le nom de l'interface à indiquer pour `vnet1` à l'aide de la commande `dladm show-phys`.

  ```
  primary# dladm show-phys |grep vnet1
  net2              Ethernet             up         0      unknown   vnet1

  primary# dladm show-linkprop -p mac-address net2
  LINK     PROPERTY        PERM VALUE          DEFAULT        POSSIBLE
  net2     mac-address     rw   00:14:4f:f8:dd:68 00:14:4f: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 (`vnet`), un commutateur virtuel (`vsw`) 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 son`tadd-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. Cela 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 à Logical Domains

Le bloc suivant d'adresses MAC 512K est assigné à Logical Domains :

```
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.

Avant de sélectionner une adresse potentielle, Logical Domains Manager recherche d'abord s'il a des adresses assignées automatiquement libérées récemment enregistrées dans une base de données à ces fins (voir "Adresses MAC libérées" à la page 161). Dans ce cas, Logical Domains Manager sélectionne son adresse candidate dans la base de données.

Si aucune adresse libérée récemment n'est disponible, l'adresse MAC est sélectionnée de manière aléatoire dans la plage 256K d'adresses mise de côté dans ce but. 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 Manager 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 160. Si l'adresse est déjà assignée, Logical Domains Manager itère en choisissant une autre adresse et en vérifiant à nouveau les collisions. Cela continue jusqu'à ce qu'une adresse MAC non déjà allouée soit trouvée ou 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, un Logical Domains Manager effectue une vérification auprès les autres Logical Domains Manager sur les autres systèmes en envoyant un message multidiffusion par 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 de Logical Domains a déjà été associé à cette adresse MAC, Logical Domains Manager de ce système renvoie une réponses contenant l'adresse MAC en question. Si Logical Domains Manager demandant reçoit une réponse, il sait que l'adresse MAC choisir a déjà été allouée, en choisit une autre, puis itère.

Par défaut, ces messages multidiffusion ne sont envoyés qu'aux autres gestionnaires sur le même sous-réseau : le TTL (time-to-live) par défaut 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 :

- De l'écoute des messages multidiffusion
- Du suivi des adresses MAC assignées à ses domaines
- De la recherche des 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.

Adresses MAC libérées

Lorsqu'un domaine logique ou un périphérique associé à une adresse MAC automatique est supprimé, l'adresse MAC est enregistrée dans une base de données des adresses MAC libérées récemment pour une utilisation ultérieure potentielle sur ce système. Les adresses MAC sont enregistrées pour éviter l'épuisement des adresses IP (Internet Protocol) d'un serveur DHCP (Dynamic Host Configuration Protocol). Lorsque les serveurs DHCP allouent des adresses IP, ils effectuent cette opération pour une certaine période (durée de location). La durée de location est souvent configurée comme très longue, généralement des heures ou des jours. Si les périphériques réseau sont créés et supprimés à une vitesse élevée sans que Logical Domains Manager réutilise automatiquement les adresses MAC, le nombre d'adresses MAC allouées peut rapidement submerger un serveur DHCP configuré de manière classique.

Lorsque Logical Domains Manager est invité à obtenir automatiquement l'adresse MAC pour un domaine logique ou un périphérique réseau, il recherche d'abord dans la base de données des adresses MAC libérées pour voir s'il y a une adresse MAC assignée précédemment qu'il peut utiliser. S'il y a une adresse MAC disponible dans cette base de données, l'algorithme de détection d'adresse MAC en doublon est exécuté. Si l'adresse MAC n'a pas été assignée à quelqu'un d'autre depuis sa libération, elle sera réutilisée et supprimée de la base de données. Si une collision est détectée, l'adresse est tout simplement supprimée de la base de données. Logical Domains Manager essaye ensuite l'adresse suivante dans la base de données ou s'il n'y en a aucune disponible, sélectionne de manière aléatoire une nouvelle adresse MAC.

Utilisation des adaptateurs réseau avec Logical Domains

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 162 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 des groupements de liens, reportez-vous à la section "Utilisation du groupement de liens avec un commutateur virtuel" à la page 183.

▼ Procédure de détermination de la compatibilité GLDv3 d'un adaptateur réseau (Oracle Solaris 10)

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 auront un type `non-vlan` ou `vlan`.
  - Les pilotes non compatibles GLDv3 auront un 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 le réseau virtuel (`vnet`) 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.

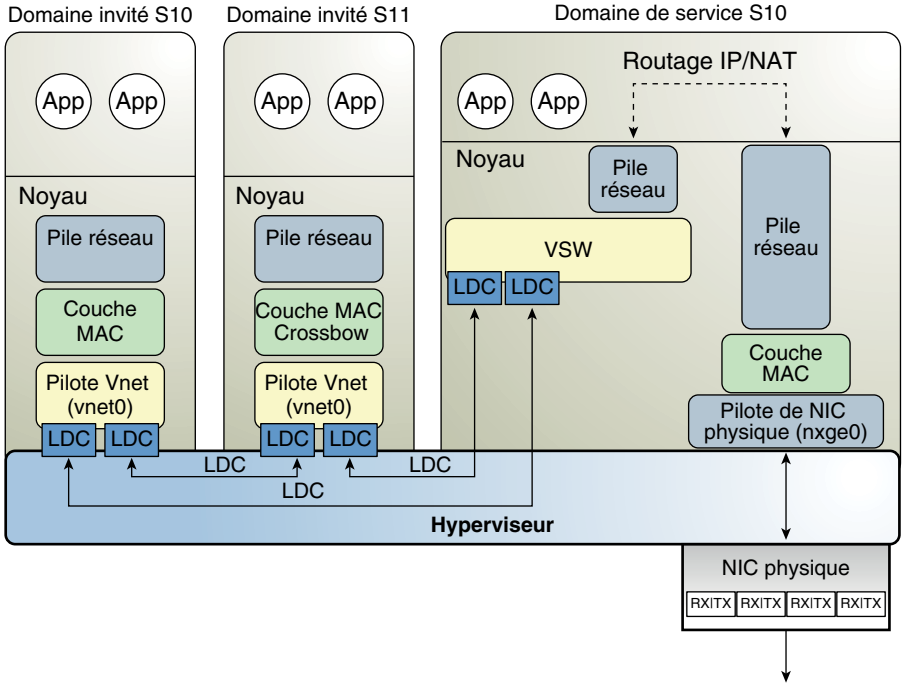
Les avantages de cette configuration sont :

- 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 10

Le diagramme 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 8–5 Routage du réseau virtuel



▼ Procédure de configuration d'un commutateur virtuel pour fournir une connectivité externe aux domaines (Oracle Solaris 10)

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] primary-vsw0 primary
```

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 66.

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 66 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.**

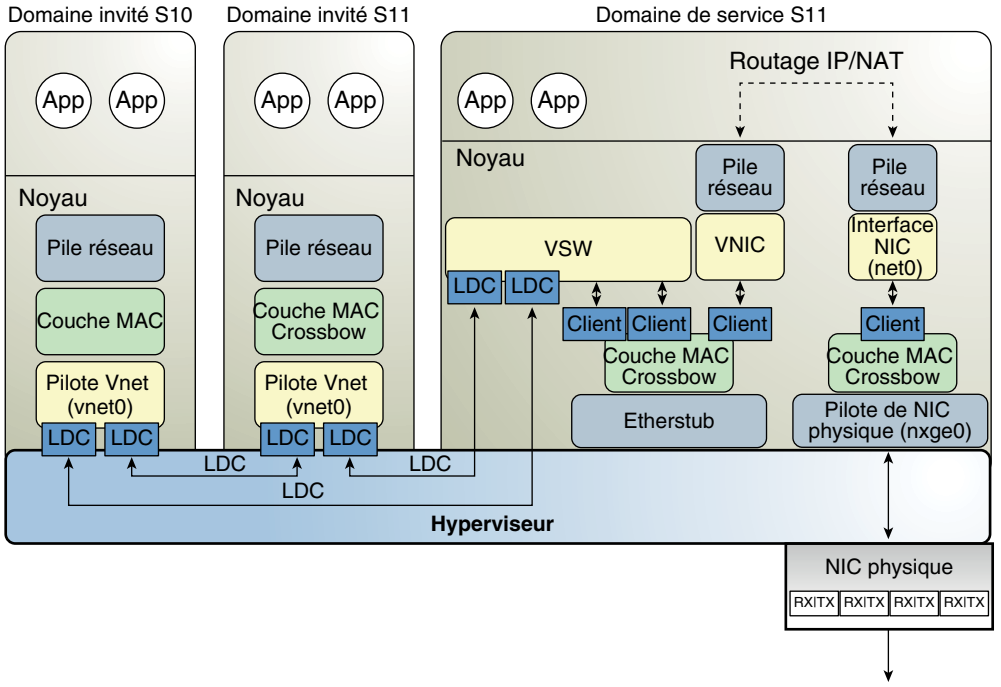
Pour plus d'informations sur le routage IP, reportez-vous à la section "Reenvío de paquetes y rutas en redes IPv4" du manuel *Guía de administración del sistema: servicios IP*.

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 diagramme 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 8–6 Routage du réseau virtuel



▼ 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 vnicstub0
primary# ipadm create-addr -T static -a 192.168.100.1/24  vnicstub0/v4static
```

5 Activez le transfert IPv4 et créez des règles NAT.

Reportez-vous aux sections "Configuración de las propiedades de la interfaz IP" du manuel *Administración de Oracle Solaris: interfaces y virtualización de redes* et "Reenvío de paquetes y rutas en redes IPv4" du manuel *Guía de administración del sistema: servicios IP*.

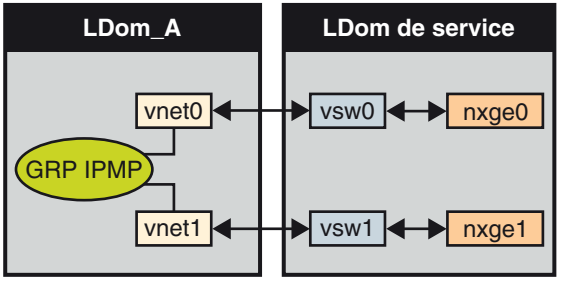
Configuration d'IPMP dans un environnement Logical Domains

Le logiciel Oracle VM Server for SPARC prend en charge l'IPMP (IP network multipathing) 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

Le diagramme suivant 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 Oracle Solaris 10 et `net0` et `net1` dans Oracle Solaris 11. Le diagramme présente les noms des interfaces physiques 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.

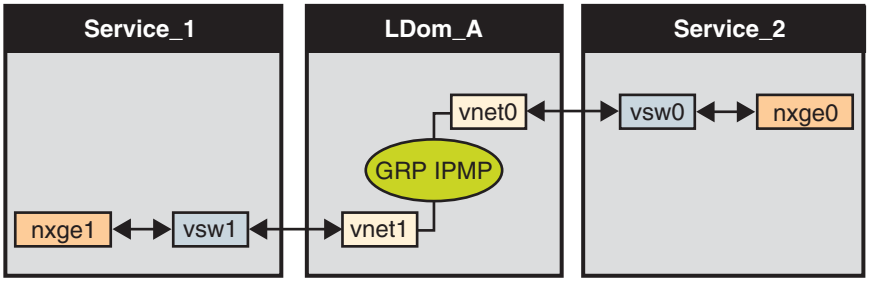
FIGURE 8–7 Deux réseaux virtuels connectés à des instances de commutateur virtuel distinctes



Remarque – La Figure 8–7 présente la configuration sur un système Oracle Solaris 10. Pour un système Oracle Solaris 11, seuls les noms des interfaces sont modifiés et remplacés par des noms génériques tels que `net0` et `net1` respectivement pour `nxge0` et `nxge1`.

On peut 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 (comme représenté dans le schéma suivant). 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 8–8 Connexion de chaque périphérique réseau virtuel à des domaines de service différents



Remarque – La Figure 8–8 présente la configuration sur un système Oracle Solaris 10. Pour un système Oracle Solaris 11, seuls les noms des interfaces sont modifiés et remplacés par des noms génériques tels que `net0` et `net1` respectivement pour `nxge0` et `nxge1`.

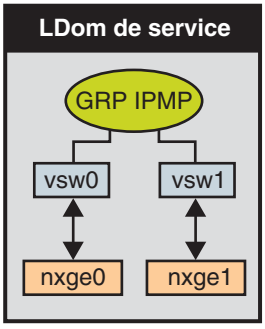
Pour plus d'informations, reportez-vous au manuel Oracle Solaris 10 *Guía de administración del sistema: servicios IP* ou au manuel Oracle Solaris 11 *Administración de Oracle Solaris: servicios IP*.

Configuration et utilisation d'IPMP dans le domaine de service

IPMP peut être configuré dans le domaine de service en configurant des interfaces de commutateur virtuel dans un groupe. Le schéma suivant représente deux instances de commutateur virtuel (`vsw0` et `vsw1`) 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 `nxge0` et `nxge1` dans Oracle Solaris 10 et `net0` et `net1` dans Oracle Solaris 11. Le diagramme suivant présente les noms des interfaces physiques Oracle Solaris 10.

FIGURE 8–9 Configuration de deux interfaces de commutateur virtuel en tant que membres d'un groupe IPMP



Remarque – La Figure 8–9 présente la configuration sur un système Oracle Solaris 10. Pour un système Oracle Solaris 11, seuls les noms des interfaces sont modifiés et remplacés par des noms génériques tels que `net0` et `net1` respectivement pour `nxge0` et `nxge1`.

Utilisation de l'IPMP basé sur liaison dans la mise en réseau virtuelle de Logical Domains

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). Cette configuration est activée par défaut et ne nécessite pas d'autre étape de configuration de votre part.

Parfois, il peut être nécessaire de détecter les changements d'état de la liaison au réseau physique. Par exemple, si un périphérique physique a été assigné à un commutateur virtuel, la liaison du réseau physique à partir du domaine de service vers le réseau externe peut ne pas fonctionner même si la liaison du périphérique réseau virtuel à son périphérique de commutateur virtuel fonctionne. Dans ce cas, il peut être nécessaire d'obtenir et de signaler l'état de la liaison physique au périphérique réseau virtuel et à sa pile.

L'option `linkprop=phys-state`peut être utilisée pour configurer le suivi de l'état de la liaison physique pour les périphériques du réseau virtuel ainsi que pour les périphériques du commutateur virtuel. Lorsque cette option est activée, le périphérique virtuel (réseau virtuel ou

commutateur virtuel) signale son état de liaison en fonction de l'état de liaison physique pendant qu'il est créé en tant qu'interface dans le domaine. 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 Logical Domains. 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` au SE Oracle Solaris 10 8/11, 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 décrit comment activer les mises à jour de l'état de liaison pour les périphériques du 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 Prenez le rôle d'administrateur, de superutilisateur ou un rôle équivalent.

Pour Oracle Solaris 10, reportez-vous à la section "Configuración de RBAC (mapa de tareas)" du manuel *Guía de administración del sistema: servicios de seguridad*. Pour Oracle Solaris 11, reportez-vous à la Partie III, "Roles, perfiles de derechos y privilegios" du manuel *Administración de Oracle Solaris: servicios de seguridad*.

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.

  ```
  # ldm add-vnet linkprop=phys-state if-name vswitch-name ldom
  ```

  L'exemple suivant active les mises à jour de l'état de liaison physique pour `vnet0` connecté à `primary-vsw0` sur le domaine logique `ldom1` :

  ```
  # ldm add-vnet linkprop=phys-state 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`.

  ```
  # ldm set-vnet linkprop=phys-state if-name ldom
  ```

  L'exemple suivant active les mises à jour de l'état de liaison physique pour `vnet0` sur le domaine logique `ldom1` :

  ```
  # ldm set-vnet linkprop=phys-state 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 `vnet0` sur le domaine logique `ldom1` :

```
# ldm set-vnet linkprop= vnet0 ldom1
```

Exemple 8–1 Configuration d'IPMP basé sur liaison

Les exemples suivants montrent comment configurer l'IPMP basé sur liaison avec et sans activer les 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.

```
# ldm add-vnet vnet0 primary-vsw0 ldom1
# ldm add-vnet 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.

- **SE 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
  ```

- **SE 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
  # 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
  ```

- 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.

  Notez que `net0` et `net1` sont respectivement les noms propres Oracle Solaris 11 de `vnet0` et `vnet1`.

  - **SE Oracle Solaris 10.** Utilisez les commandes suivantes :

    ```
    # ldm add-vnet linkprop=phys-state vnet0 primary-vsw0 ldom1
    # ldm add-vnet linkprop=phys-state vnet1 primary-vsw1 ldom1
    ```

  - **SE Oracle Solaris 11.** Utilisez les commandes suivantes :

    ```
    # ldm add-vnet linkprop=phys-state net0 primary-vsw0 ldom1
    # ldm add-vnet linkprop=phys-state net1 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 :

- **SE 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
```

- **SE 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
```

Configuration et utilisation d'IPMP dans les versions antérieures à Logical Domains 1.3

Dans les versions de Logical Domains antérieures à la version 1.3, les périphériques de commutateur virtuel et de réseau virtuel ne sont pas capables d'effectuer la détection des pannes de liaison. Dans ces versions, la détection et la récupération des pannes réseau peut être configurée à l'aide d'IPMP basé sur sonde.

Configuration d'IPMP dans le domaine invité

Les périphériques réseau virtuels dans un domaine invité peuvent être configurés en groupe IPMP, comme illustré à la Figure 8–7 et à la Figure 8–8. La seule différence est que la détection des pannes par sonde est utilisée en configurant des adresses de test sur les périphériques de réseau virtuel. Reportez-vous au *Guía de administración del sistema: servicios IP* pour plus d'informations sur la configuration d'IPMP par sonde.

Configuration d'IPMP dans le domaine de service

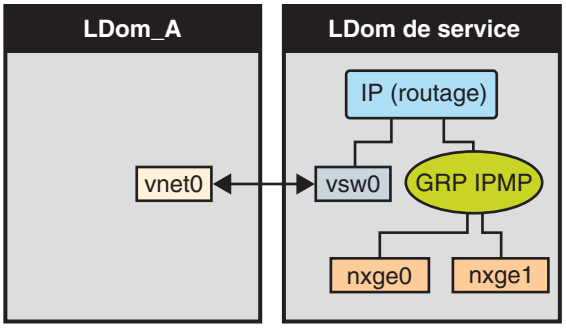
Dans les version de Logical Domains antérieures à la version 1.3, le périphérique de commutateur virtuel n'est pas capable de détecter les pannes de liaison physique. Dans ce cas, la détection et la récupération des pannes réseau peuvent être définies en configurant les interfaces

physiques dans le domaine de service en groupe IPMP. Pour ce faire, configurez le commutateur virtuel du domaine de service sans assigner de périphérique réseau physique à ce dernier. N'indiquez aucune valeur pour la propriété `net-dev` (`net-dev=`) lorsque vous utilisez la commande `ldm add-vswitch` pour créer le commutateur virtuel. Créez l'interface de commutateur virtuel dans le domaine de service et configurez le domaine de service lui-même pour qu'il agisse comme un routeur IP. Reportez-vous au manuel *Guía de administración del sistema: servicios IP* d'Oracle Solaris 10 pour plus d'informations sur la configuration du routage IP.

Une fois configuré, le commutateur virtuel envoie tous les paquets provenant des réseaux virtuels (et destinés à une machine externe) à sa couche IP, au lieu d'envoyer les paquets directement au moyen du périphérique physique. En cas de panne de l'interface physique, la couche IP détecte la panne et réachemine automatiquement les paquets via l'interface secondaire.

Comme les interfaces physiques sont directement configurées en groupe IPMP, le groupe peut être configuré pour une détection basée sur la liaison ou sur la sonde. Le schéma suivant représente deux interfaces réseau (`nxge0` et `nxge1`) configurées en tant que membre du groupe IPMP. L'instance de commutateur virtuel (`vsw0`) a été créée en tant que périphérique réseau pour envoyer des paquets à sa couche IP.

FIGURE 8–10 Configuration de deux interfaces réseau en tant que membres d'un groupe IPMP



Remarque – La Figure 8–10 présente la configuration sur un système Oracle Solaris 10. Pour un système Oracle Solaris 11, seuls les noms des interfaces sont modifiés et remplacés par des noms génériques tels que `net0` et `net1` respectivement pour `nxge0` et `nxge1`.

▼ Procédure de configuration de la route d'un hôte pour l'IPMP basé sur sonde

Remarque – Cette procédure s'applique uniquement aux domaines invités et aux versions antérieures à la version 1.3, où seul l'IPMP basé sur sonde est pris en charge.

Si aucune route explicite n'est configurée pour un routeur dans le réseau correspondant aux interfaces IPMP, une ou plusieurs routes hôtes explicites vers les systèmes cibles doivent être configurées pour que la détection IPMP basée sur sonde fonctionne comme prévu. Sinon, la détection par sonde peut ne pas parvenir à détecter les défaillances réseau.

- **Configurez une route hôte.**

  ```
  # route add -host destination-IP gateway-IP -static
  ```

  Par exemple :

  ```
  # route add -host 192.168.102.1 192.168.102.1 -static
  ```

  Reportez-vous à la section "Configuración de sistemas de destino" du manuel *Guía de administración del sistema: servicios IP* pour plus d'informations.

Utilisation du balisage VLAN

Le logiciel Oracle VM Server for SPARC prend en charge le balisage VLAN 802.1Q dans l'infrastructure réseau.

Remarque – Les VLAN balisés ne sont pas pris en charge dans les versions précédentes des composants de mise en réseau de Logical Domains.

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 `vnet` 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 Logical Domains est que vous devez assigner le `vnet` aux VLAN correspondants à l'aide des commandes de la CLI de Logical Domains Manager. Reportez-vous au `ldm(1M)` pour obtenir des informations complètes sur les commandes de la CLI de Logical Domains Manager.

De même, vous pouvez configurer des interfaces VLAN sur un périphérique de commutateur virtuel dans le domaine de service. Les ID de VLAN 2 à 4094 sont valides, l'ID de VLAN 1 est réservées en tant que `default-vlan-id`.

Lorsque vous créez un périphérique `vnet` sur un domaine invité, vous devez lui assigner les VLAN requis en définissant l'ID de VLAN du port et zéro ou plusieurs ID de VLAN pour ce `vnet` à l'aide des arguments `pvid=` et `vid=` de la commande `ldm add-vnet`. Cette opération configure le commutateur virtuel afin qu'il prenne en charge plusieurs VLAN dans le réseau Logical Domains et commute les paquets à l'aide de l'adresse MAC et des ID de VLAN dans le réseau.

De même, les VLAN auxquels le périphérique `vsw` lui-même doit appartenir, s'il est créé en tant qu'interface réseau, doivent être configurés dans le périphérique `vsw` à l'aide des arguments `pvid=` et `vid=` de la commande `ldm add-vsw`.

Vous pouvez modifier les VLAN auxquels un périphérique appartient à l'aide de la commande `ldm set-vnet` ou `ldm set-vsw`.

ID du VLAN du port (PVID)

Le PVID définit le VLAN dont le périphérique réseau virtuel doit être membre 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 `vnet` 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. Vous ne pouvez avoir qu'un seul PVID pour un périphérique `vnet`.

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 `vnet 0` à l'aide de l'une des commandes suivantes et si 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`.

- **SE Oracle Solaris 10.** Utilisez la commande `ifconfig`.

  ```
  # ifconfig vnet0 plumb
  ```

- **SE Oracle Solaris 11.** Utilisez la commande `ipadm`.

  ```
  # ipadm create-ip net0
  ```

ID de VLAN (VID)

Le VID indique le VLAN dont un périphérique réseau virtuel ou un commutateur virtuel doit être membre 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.

▼ Procédure d'assignation de VLAN à un commutateur virtuel et à un périphérique réseau virtuel

1 Assignez le commutateur virtuel (`vsw`) à deux VLAN.

Par exemple, configurez le VLAN 21 comme non balisé et le VLAN 20 comme balisé. Assignez le réseau virtuel (`vnet`) à trois VLAN. Configurez le VLAN 20 comme non balisé et les VLAN 21 et 22 comme balisés.

```
# ldm add-vsw net-dev=nxge0 pvid=21 vid=20 primary-vsw0 primary
# ldm add-vnet pvid=20 vid=21,22 vnet01 primary-vsw0 ldom1
```

2 Créez les interfaces VLAN.

Cet exemple suppose que le numéro d'interface de ces périphériques est `0` dans les domaines et que les VLAN sont mappés sur les sous-réseaux suivants :

VLAN 20 Sous-réseau `192.168.1.0` (masque réseau : `255.255.255.0`)

VLAN 21 Sous-réseau `192.168.2.0` (masque réseau : `255.255.255.0`)

VLAN 22 Sous-réseau `192.168.3.0` (masque réseau : `255.255.255.0`)

a. Créez l'interface VLAN dans le domaine de service (`primary`).

- **SE Oracle Solaris 10. Utilisez la commande `ifconfig`.**

```
primary# ifconfig vsw0 plumb
primary# ifconfig vsw0 192.168.2.100 netmask 0xffffff00 broadcast + up
primary# ifconfig vsw20000 plumb
primary# ifconfig vsw20000 192.168.1.100 netmask 0xffffff00 broadcast + up
```

- **SE Oracle Solaris 11. Utilisez les commandes `dladm` et `ipadm`.**

```
primary# dladm create-vlan -l vsw0 -v20
primary# ipadm create-ip net0
primary# ipadm create-addr -T static -a 192.168.2.100/24 net0/ipv4
primary# ipadm create-ip net20000
primary# ipadm create-addr -T static -a 192.168.1.100/24 net20000/ipv4
```

b. Créez l'interface VLAN dans le domaine invité (`ldom1`).

- **SE Oracle Solaris 10. Utilisez la commande `ifconfig`.**

```
ldom1# ifconfig vnet0 plumb
ldom1# ifconfig vnet0 192.168.1.101 netmask 0xffffff00 broadcast + up
ldom1# ifconfig vnet21000 plumb
ldom1# ifconfig vnet21000 192.168.2.101 netmask 0xffffff00 broadcast + up
ldom1# ifconfig vnet22000 plumb
ldom1# ifconfig vnet22000 192.168.3.101 netmask 0xffffff00 broadcast + up
```

  Pour plus d'informations sur la procédure de configuration des interfaces VLAN dans le système d'exploitation Oracle Solaris 10, reportez-vous à la section "Administración de redes de área local virtuales" du manuel *Guía de administración del sistema: servicios IP*.

- **SE Oracle Solaris 11. Utilisez les commandes `dladm` et `ipadm`.**

```
ldom1# dladm create-vlan -l net0 -v21
ldom1# ipadm create-ip net0
ldom1# ipadm create-addr -T static -a 192.168.1.101/24 net0/ipv4
ldom1# ipadm create-ip net21000
ldom1# ipadm create-addr -T static -a 192.168.2.101/24 net21000/ipv4
ldom1# ipadm create-ip net22000
ldom1# ipadm create-addr -T static -a 192.168.3.101/24 net22000/ipv4
```

  Pour plus d'informations sur la procédure de configuration des interfaces VLAN dans le système d'exploitation Oracle Solaris 11, reportez-vous à la section "Administración de redes de área local virtuales" du manuel *Administración de Oracle Solaris: interfaces y virtualización de redes*.

▼ 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 est *uniquement* prise en charge sur les systèmes Oracle Solaris 10. Définissez l'ID de VLAN associée au serveur d'installation comme le PVID du périphérique réseau virtuel et ne configurez pas de VLAN balisés (`vid`) pour ce périphérique réseau virtuel. Vous devez effectuer cette opération, car l'OBP ne connaît pas les VLAN et ne peut pas traiter les paquets réseau balisés VLAN. Le commutateur virtuel traite le non balisage et le balisage des paquets de et vers le domaine invité au cours de l'installation réseau. A la fin de l'installation réseau et après la démarrage du SE Oracle Solaris, vous pouvez configurer le périphérique du réseau virtuel pour qu'il soit balisé dans ce VLAN. Vous pouvez ensuite ajouter le périphérique réseau virtuel à d'autres VLAN en mode balisé.

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 76.

1. **Configurez le périphérique réseau en mode non balisé au départ.**

   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
   ```

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
   ```

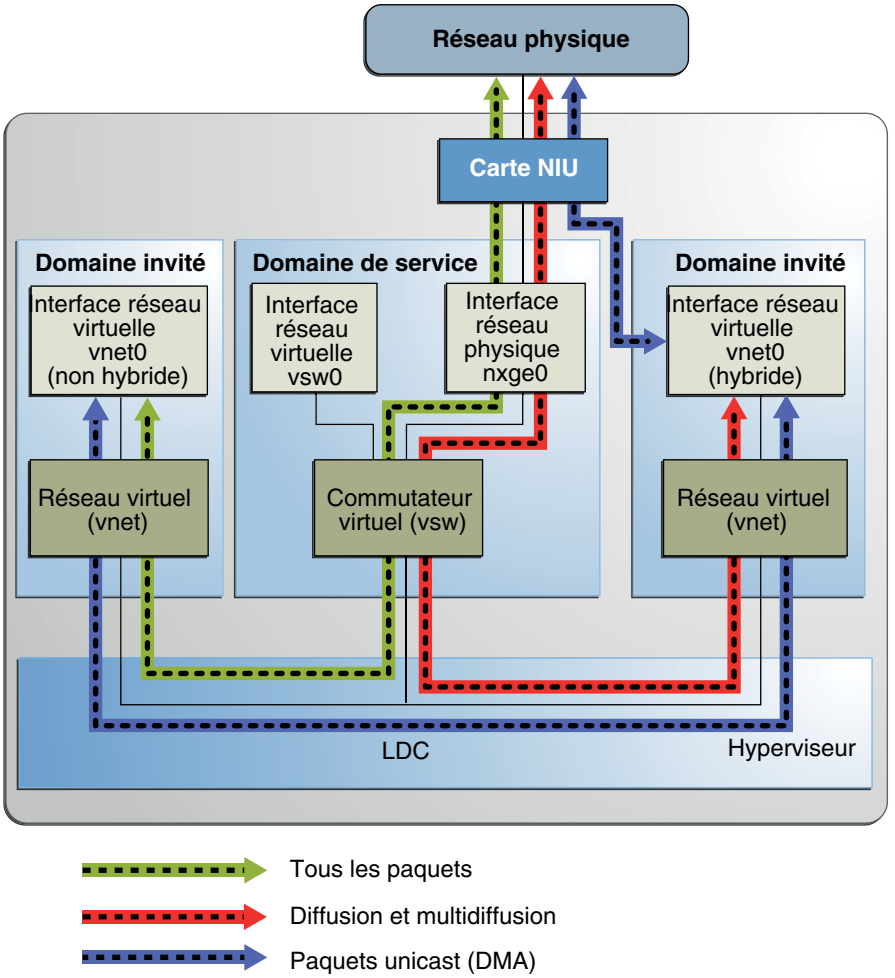
Utilisation des E/S hybrides NIU

Le cadre d'E/S virtuel implémente un modèle d'E/S *hybride* pour optimiser les fonctionnalités et les performances. Le modèle d'E/S hybride associe des E/S directes et virtualisées pour permettre un déploiement flexible des ressources d'E/S sur les machines virtuelles. Cela est particulièrement utile lorsque les E/S directes ne fournissent pas une fonctionnalité complète pour la machine virtuelle ou que les E/S directes ne sont pas disponibles de manière continue ou cohérente pour la machine virtuelle. Cela peut arriver en raison de la disponibilité des ressources ou de la migration de la machine virtuelle. L'architecture d'E/S hybride est bien adaptée pour l'unité d'interface réseau (NIU) sur les plates-formes Sun UltraSPARC T2, SPARC T3 et SPARC T4. Une NIU est une interface d'E/S réseau intégrée sur la puce. Cette architecture permet l'assignation dynamique de ressources DMA (Direct Memory Access) aux périphériques de mise en réseau virtuels et, fournit ainsi des performances cohérentes aux applications du domaine.

Les E/S hybrides NIU sont disponibles pour les plates-formes Sun UltraSPARC T2, SPARC T3 et SPARC T4. Cette fonction est activée par un mode hybride facultatif qui fournit un périphérique réseau virtuel (`vnet`) où les ressources matérielles DMA sont louées à un périphérique `vnet` dans un domaine invité pour améliorer les performances. En mode hybride, un périphérique vnet dans un domaine invité peut envoyer et recevoir du trafic unicast à partir d'un réseau externe directement dans le domaine invité à l'aide des ressources matérielles DMA. Le trafic de diffusion ou de multidiffusion et le trafic unicast vers les autres domaines invités du même système continuent à être envoyés à l'aide du mécanisme de communication d'E/S.

Remarque – Les E/S hybrides NIU ne sont pas disponibles sur les plates-formes UltraSPARC T2 Plus.

FIGURE 8–11 Mise en réseau virtuelle hybride



Remarque – La Figure 8–11 présente la configuration sur un système Oracle Solaris 10. Pour un système Oracle Solaris 11, seuls les noms des interfaces sont modifiés et remplacés par des noms génériques tels que `net0` pour `nxge0`.

Le mode hybride s'applique uniquement aux périphériques `vnet` associés à un commutateur virtuel (`vsw`) configuré pour utiliser un périphérique réseau NIU. Comme les ressources matérielles DMA partageables sont limitées, jusqu'à trois périphériques `vnet` par `vsw` peuvent se voir assigner des ressources matérielles DMA sur une période donnée. Si plus de trois périphériques `vnet` ont le mode hybride activé, l'assignation est effectué sur la base du premier

arrivé, premier servi. Comme il y a deux périphériques réseau NIU dans un système, il peut y avoir un total de six périphériques `vnet` sur deux commutateurs virtuels différents avec des ressources matérielles DMA assignées.

Vous trouverez ci-dessous des points dont vous devez avoir connaissance lors de l'utilisation de cette fonction :

- L'option de mode hybride pour un périphérique `vnet` est traitée comme une suggestion uniquement. Cela signifie que les ressources DMA sont assignées uniquement lorsqu'elles sont disponibles et que le périphérique est capable de les utiliser.
- Les commandes de la CLI de Logical Domains Manager ne valident pas l'option de mode hybride, c'est-à-dire qu'il est possible de définir le mode hybride sur un périphérique `vnet` quelconque ou sur un certain nombre de périphériques `vnet`.
- Les domaines invités et le domaine de service doivent exécuter le SE Oracle Solaris 10 10/08 au moins.
- Un maximum de trois périphériques `vnet` par `vsw` peut disposer des ressources matérielles DMA louées à un moment donné. Comme il y a deux périphériques réseau NIU, il peut y avoir un total de six périphériques `vnet` avec des ressources matérielles DMA louées.

  Remarque – Ne définissez le mode hybride que pour trois périphériques `vnet` par `vsw` afin qu'ils soient sûrs d'avoir des ressources matérielles DMA assignées.

- Le mode hybride est désactivé par défaut pour un périphérique `vnet`. Il doit être activé de manière explicite à l'aide de la commande `ldm`. Reportez-vous à la section "Procédure d'activation du mode hybride" à la page 182 et à la page de manuel `ldm`(1M).
- L'option de mode hybride ne peut pas être modifiée de manière dynamique lorsque le domaine invité est actif.
- Les ressources matérielles DMA ne sont assignées que lorsqu'un périphérique `vnet` créé dans le domaine invité est actif.
- Le pilote Ethernet NIU 10 gigabits (`nxge`) est utilisé pour la carte NIU. Le même pilote est également utilisé pour les autres cartes réseau 10 gigabits. Cependant, la fonction d'E/S hybride NIU n'est disponible que pour les périphériques réseau NIU.

▼ Procédure de configuration d'un commutateur virtuel avec un périphérique réseau NIU

1. **Déterminez un périphérique réseau NIU.**

   L'exemple suivant montre la sortie sur un serveur UltraSPARC T2 :

   ```
   # grep nxge /etc/path_to_inst
   "/niu@80/network@0" 0 "nxge"
   "/niu@80/network@1" 1 "nxge"
   ```

   L'exemple suivant montre la sortie sur un serveur SPARC T3-1 ou SPARC T4-1 :

   ```
   # grep nxge /etc/path_to_inst
   "/niu@480/network@0" 0 "nxge"
   "/niu@480/network@1" 1 "nxge"
   ```

2. **SE Oracle Solaris 11 uniquement : identifiez le lien correspondant au périphérique réseau NIU, comme par exemple `nxge0`.**

   ```
   primary# dladm show-phys -L |grep nxge0
   net2              nxge0         /SYS/MB
   ```

3. **Configurez un commutateur virtuel.**

   - **SE Oracle Solaris 10. Utilisez la commande suivante :**

     ```
     # ldm add-vsw net-dev=nxge0 primary-vsw0 primary
     ```

   - **SE Oracle Solaris 11. Utilisez la commande suivante :**

     L'exemple suivant utilise `net0` au lieu de `nxge0`.

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

▼ Procédure d'activation du mode hybride

- **Par exemple, activez le mode hybride pour un périphérique `vnet` lors de sa création.**

  ```
  # ldm add-vnet mode=hybrid vnet01 primary-vsw0 ldom01
  ```

▼ Procédure de désactivation du mode hybride

- **Par exemple, désactivez le mode hybride pour un périphérique `vnet`.**

  ```
  # ldm set-vnet mode= vnet01 ldom01
  ```

Utilisation du groupement de liens avec un commutateur virtuel

Il est possible de configurer un commutateur virtuel de manière à ce qu'il utilise un groupement de liens. Un groupement de liens 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 procédure de configuration du groupement de liaisons, reportez-vous au *Guía de administración del sistema: servicios IP*.

Après avoir créé un groupement de liens, 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`.

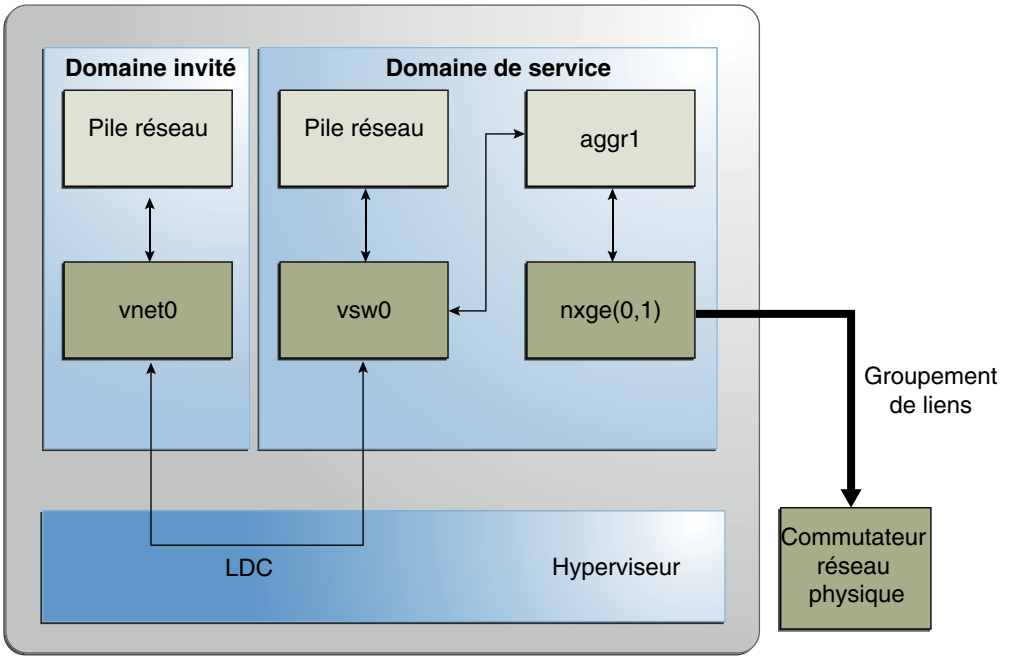
Lorsque le groupement de liens est assigné au commutateur virtuel, le trafic de et vers le réseau physique s'écoule au travers du groupement. Tout équilibrage de charge ou basculement nécessaire est traité de manière transparente par le cadre de groupement sous-jacent. Le groupement de liens est complètement 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 liens.

Vous pouvez créer et utiliser le commutateur virtuel configuré pour utiliser un groupement de liens dans le domaine de service. Reportez-vous à la section “Procédure de configuration du commutateur virtuel en tant qu'interface primary” à la page 66.

La figure suivante représente un commutateur virtuel configuré pour utiliser un groupement `aggr1` sur les interfaces physiques `nxge0` et `nxge1`.

FIGURE 8–12 Configuration d'un commutateur virtuel pour utiliser un groupement de liens



Remarque – La Figure 8–12 présente la configuration sur un système Oracle Solaris 10. Pour un système Oracle Solaris 11, seuls les noms des interfaces sont modifiés et remplacés par des noms génériques tels que `net0` et `net1` respectivement pour `nxge0` et `nxge1`.

Configuration de trames géantes

Le commutateur virtuel de Logical Domains (`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. Ce changement a pour conséquence que ces pilotes peuvent augmenter la capacité de traitement du réseau.

▼ Procédure de configuration du réseau virtuel et des périphériques de commutateur virtuels pour l'utilisation de trames géantes

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.

Dans certains cas, vous pouvez définir une valeur MTU directement sur un périphérique réseau virtuel. Vous pouvez effectuer cette opération si la valeur MTU requise pour le périphérique du réseau virtuel doit être inférieure à celle prise en charge par le commutateur virtuel.

Remarque – Sur le SE Oracle Solaris 10 5/09, 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).

---

1 **Connectez-vous au domaine de contrôle.**

2 **Prenez le rôle d'administrateur, de superutilisateur ou un rôle équivalent.**

Pour Oracle Solaris 10, reportez-vous à la section "Configuración de RBAC (mapa de tareas)" du manuel *Guía de administración del sistema: servicios de seguridad*. Pour Oracle Solaris 11, reportez-vous à la Partie III, "Roles, perfiles de derechos y privilegios" du manuel *Administración de Oracle Solaris: servicios de seguridad*.

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`.

```
# ldm add-vsw 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`.

  ```
  # ldm set-vsw 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.

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 :

```
# ldm set-vnet mtu= vnet-name ldom
```

Notez que l'activation des trames géantes pour un périphérique réseau virtuel active automatiquement les trames géantes pour une ressource d'E/S hybride assignée à un périphérique réseau virtuel.

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.

Exemple 8–2 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 un périphérique de commutateur virtuel, `primary-vsw0`, avec une valeur MTU de `9000`. Notez que l'instance 0 du périphérique réseau `nxge0` est définie comme une valeur de la propriété `net-dev`.

```
# ldm add-vsw net-dev=nxge0 mtu=9000 primary-vsw0 primary
```

Ensuite, la commande `ldm add-vnet` ajoute un périphérique réseau virtuel client à ce commutateur virtuel, `primary-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`.

```
# ldm add-vnet vnet01 primary-vsw0 ldom1
```

Selon la version du SE Oracle Solaris en cours d'exécution, procédez comme suit :

- **SE Oracle Solaris 10.** La commande `ifconfig` crée l'interface du commutateur virtuel dans le domaine de service, `primary`. La sortie de la commande `ifconfig vsw0` montre que la valeur de la propriété `mtu` est `9000`.

  ```
  # ifconfig vsw0 plumb
  # ifconfig vsw0 192.168.1.100/24 up
  # 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, `ldom1`. La sortie de la commande `ifconfig vnet0` montre que la valeur de la propriété `mtu` est `9000`.

  ```
  # ifconfig vnet0 plumb
  # ifconfig vnet0 192.168.1.101/24 up
  # 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
  ```

- **SE Oracle Solaris 11.** Utilisez les commandes `ipadm` et `dladm` pour modifier la valeur de la propriété `mtu` de l'interface principale et la définir sur `9000`.

  ```
  # 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 `ldom1`. La sortie de la commande `ipadm show-ifprop` montre que la valeur de la propriété `mtu` est `9000`.

  ```
  # ipadm create-ip net0
  # ipadm create-addr -T static -a 192.168.1.101/24 net0/ipv4
  # ipadm show-ifprop -p mtu net0
  IFNAME PROPERTY PROTO PERM CURRENT PERSISTENT DEFAULT POSSIBLE
  net0   mtu      ipv4  rw   9000    --         9000    68-9000
  ```

- 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.

  - **SE Oracle Solaris 10.** Utilisez la commande `ifconfig`.

    ```
    # ifconfig vnet0 mtu 4000
    # 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
    ```

  - **SE Oracle Solaris 11.** Utilisez la commande `ipadm`.

    ```
    # ipadm set-ifprop -p mtu=4000 net0
    # ipadm show-ifprop -p mtu net0
    IFNAME PROPERTY PROTO PERM CURRENT PERSISTENT DEFAULT POSSIBLE
    net0   mtu      ipv4  rw   4000    --         9000    68-9000
    ```

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 système d'exploitation 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 Logical Domains 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. Cela s'applique 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 Logical Domains sont 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.

Différences liées aux fonctions de gestion réseau d'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. Cette section décrit les différences fonctionnelles concernant le périphérique réseau virtuel et le commutateur virtuel Oracle VM Server for SPARC lorsque le système d'exploitation 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* pour les domaines qui exécutent le système d'exploitation 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.

- **Utilisation d'un périphérique `etherstub` d'Oracle Solaris 11 en tant que périphérique backend pour créer un commutateur virtuel privé**

  L'utilisation de ce périphérique permet à un domaine invité de communiquer avec une zone configurée dans un domaine de service Oracle Solaris 11.

- **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 le périphérique backend qui soit 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 des VNIC sur des périphériques `vsw`*n* et `vnet`*n*. Toute tentative de création de VNIC sur `vsw`*n* et `vnet`*n* échoue. Reportez-vous à la section "Oracle Solaris 11 : les zones configurées à l'aide d'une interface réseau automatique risquent de ne pas pouvoir démarrer" du manuel *Notes de version d'Oracle VM Server for SPARC 2.2*.

◆ ◆ ◆ CHAPITRE 9

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 2.2* 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

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), et le domaine sur la machine source est 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 2.1 ou une version supérieure de Logical Domains Manager doit être en cours d'exécution sur les deux machines.
- Les machines source et cible doivent posséder une version compatible du microprogramme installé afin de prendre en charge la migration en direct. Reportez-vous à la section "Configuration requise pour la migration en direct" du manuel *Notes de version d'Oracle VM Server for SPARC 2.2*.

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, 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).

  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 l'Exemple 9–2. 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 194.
- **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

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.

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 195. 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 201.

Pour plus d'informations sur les options et les opérandes de migration, reportez-vous à la page de manuel `ldm(1M)`.

Réalisation d'une simulation

Lorsque vous indiquez l'option `-n` dans la sous-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, il est possible qu'une simulation aboutisse et qu'une migration échoue, et inversement.

Réalisation de migrations non interactives

Vous pouvez utiliser la commande `ldm migrate-domain -p` *filename* pour démarrer une opération de migration non interactive.

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 droits d'accès au fichier sont définis de sorte que seul le propriétaire root, ou un utilisateur avec des droits, puisse 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 de la migration de domaine" du manuel *Notes de version d'Oracle VM Server for SPARC 2.2*.

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 préférable, mais pas obligatoire, de disposer d'au moins 16 CPU dans chaque 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 "Gestión del protocolo de hora de red (tareas)" du manuel *Guía de administración del sistema: servicios de red*. Pour configurer un domaine en tant que client NTP Oracle Solaris 11, reportez-vous à la section "Gestión del protocolo de hora de red (tareas)" du manuel *Oracle Administración Solaris: Servicios de red*.

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.
- Pour les systèmes exécutant le SE Oracle Solaris 10, les machines source et cible doivent posséder le même type de processeur.
- Pour le SE Oracle Solaris 11, la définition de la propriété `cpu-arch` permet d'effectuer la migration entre des systèmes possédant des processeurs de type différent. Les valeurs de la propriété `cpu-arch` prises en charge sont les suivantes :

- `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.
- `generic` utilise des fonctions matérielles de la CPU pour permettre à un domaine invité d'effectuer une migration indépendante du type de CPU.

L'utilisation de la valeur `generic` peut entraîner une détérioration des performances par rapport à 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.

Utilisez la commande `psrinfo -pv` pour déterminer le type de processeur, comme suit :

```
# psrinfo -pv
The physical processor has 8 virtual processors (0-7)
  SPARC-T4 (chipid 0, clock 2548 MHz)
```

- Lorsque le domaine à migrer exécute le SE Oracle Solaris 11, vous pouvez faire migrer le domaine entre un système source et un système cible dont les processeurs ont des fréquences et des valeurs de fréquence `STICK` différentes. Ce type de migration est possible même lorsque la valeur de la propriété `cpu-arch` n'est pas définie. Toutefois, lorsque le domaine à migrer exécute le SE Oracle Solaris 10 la fréquence des processeurs et les valeurs de fréquence `STICK` des deux machines *doivent* correspondre.

  Utilisez la commande `prtconf -pv` pour déterminer la fréquence `STICK`, comme suit :

```
# prtconf -pv | grep stick-frequency
    stick-frequency:  05f4bc08
```

Remarque – La fréquence à laquelle le registre `STICK` augmente est dérivée de la fréquence de la CPU à pleine vitesse. Cependant, même si la fréquence de la CPU sur les deux machines est identique, la fréquence de registre `STICK` exacte peut légèrement différer et ainsi bloquer la migration.

Configuration requise pour la mémoire

Il doit y avoir suffisamment de mémoire libre sur la machine cible pour permettre la migration d'un domaine. Par ailleurs, voici les quelques propriétés qui doivent être conservées pendant la migration :

- Il doit être possible de créer le même nombre de blocs de mémoire de taille identique.
- Les adresses physiques des blocs de mémoire ne doivent pas nécessairement correspondre, mais les mêmes adresses réelles doivent être conservées pendant la migration.

Par ailleurs, la disposition de la mémoire disponible sur la machine cible doit être compatible avec la disposition de la mémoire du domaine à migrer ou la migration échouera. Plus précisément, si la mémoire de la machine cible est fragmentée en plusieurs petites plages d'adresse, mais si le domaine à migrer nécessite une seule large plage d'adresse, la migration échouera. L'exemple suivant illustre ce scénario. La machine cible possède 2 Go de mémoire libre dans deux blocs de mémoire :

```
# ldm list-devices memory
MEMORY
    PA                   SIZE
    0x108000000          1G
    0x188000000          1G
```

Le domaine à migrer, `ldg-src`, possède également 2 Go de mémoire libre, mais il est disposé sur un seul bloc de mémoire :

```
# ldm list -o memory ldg-src
NAME
ldg-src

MEMORY
    RA               PA               SIZE
    0x8000000        0x208000000      2G
```

Etant donné cette disposition de la mémoire, la migration échoue :

```
# ldm migrate-domain ldg-src t5440-sys-2
Target Password:
Unable to bind 2G memory region at real address 0x8000000
Domain Migration of LDom ldg-src failed
```

Remarque – Après une migration, la reconfiguration dynamique (DR) de la mémoire est désactivée pour le domaine migré tant qu'il n'a pas été redémarré. Après le redémarrage, la reconfiguration dynamique de la mémoire est réactivée 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.

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. 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 à l'aide d'un port disponible sur le premier périphérique vcc du domaine de contrôle. La migration échoue s'il y a un conflit avec le nom de groupe par défaut.

Configuration requise pour les E/S hybrides NIU

Vous pouvez migrer un domaine qui utilise des ressources d'E/S hybride NIU. Une contrainte définissant les ressources d'E/S hybride NIU n'est pas une contrainte matérielle d'un domaine logique. Si un tel domaine est migré sur une machine ne disposant pas de ressources NIU disponibles, le contrainte est préservée, mais non remplie.

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.

Les versions suivantes du SE Oracle Solaris prennent en charge la reconfiguration dynamique des unités cryptographiques :

- Au moins le SE Solaris 10 10/09
- Au moins le SE Solaris 10 5/08 OS plus patch ID 142245-01

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. Les opérations de reconfiguration retardée sont bloquées lorsqu'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

La migration de domaines n'est pas prise en charge pour une machine source ou cible sur laquelle la stratégie de gestion de d'alimentation (PM) élastique est en cours d'application. Si la stratégie PM sur la machine source ou cible est commutée de performance à élastique alors qu'une migration est en cours, la commutation de stratégie est différée jusqu'à la fin de la migration. La commande de migration renvoie une erreur si une migration de domaine est tentée lorsque la stratégie élastique est en cours d'application sur la machine source ou 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 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 à moins que le domaine à migrer dispose d'une seule CPU. Lorsque le domaine à migrer dispose d'une seule CPU, la migration est effectuée lorsque certaines contraintes et limitations sont respectées. Reportez-vous à la section "Restrictions de la migration de domaine" du manuel *Notes de version d'Oracle VM Server for SPARC 2.2*.

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.

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 peut satisfaire les contraintes du domaine migré lorsqu'une liaison est tentée ultérieurement, ou la liaison du domaine échouera.

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 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.

Pour plus d'informations sur la fonction d'E/S directes (DIO), reportez-vous à la section “Assignation des périphériques d'extrémité PCIe” à la page 86.

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 :

- Le domaine en cours de migration présente un `s` pour indiquer qu'il est la source de la migration.
- Le domaine migré présente un `t` pour indiquer qu'il est la cible de la migration.
- Si une erreur se produit et nécessite une intervention de l'utilisateur, un `e` s'affiche.

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%         t5440-sys-2
```

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 et 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.

Une opération de migration peut également être annulée à l'aide de la commande `ldm cancel-operation`. Cette commande met fin à la migration en cours et le domaine en cours de migration reprend en tant que domaine actif. La commande `ldm cancel-operation` doit être lancée à partir de la machine source. Sur une machine donnée, une commande relative à la migration a une incidence sur l'opération de migration qui a été démarrée à partir de cette machine. Une machine cible *ne peut pas* contrôler une opération de migration.

Remarque – Une fois une migration lancée, la suspension du processus `ldm` n'interrompt pas l'opération, En effet, le démon Logical Domains Manager (`ldmd`) sur les machines source et cible, et non le processus `ldm`, affecte la migration. Le processus `ldm` attend un signal de `ldmd` indiquant que la migration est terminée avant de reprendre.

Récupération sur un échec de migration

L'opération de migration prend fin si la connexion réseau est perdue de la manière suivante :

- *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 migration

EXEMPLE 9–1 Migration d'un domaine invité

Cet exemple décrit la migration du domaine `ldg1` vers une machine appelée `t5440-sys-2`.

```
# ldm migrate-domain ldg1 t5440-sys-2
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 t5440-sys-2
```

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, `t5440-sys-2`.

EXEMPLE 9–2 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 (`t5440-sys-2`) 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@t5440-sys-2:ldg-tgt
Target Password:
```

EXEMPLE 9–3 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 9–4 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 `t5440-sys-1`.

EXEMPLE 9–4 Obtention de l'état de migration pour le domaine sur la machine cible *(Suite)*

```
# ldm list -o status ldg-tgt
NAME
ldg-tgt

STATUS
    OPERATION    PROGRESS    SOURCE
    migration    55%         t5440-sys-1
```

EXEMPLE 9–5 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 `t5440-sys-2`.

```
# ldm list -o status -p ldg-src
VERSION 1.6
DOMAIN|name=ldg-src|
STATUS
|op=migration|progress=42|error=no|target=t5440-sys-2
```

◆ ◆ ◆ CHAPITRE 10

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 le redémarrage 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
- **Périphériques d'E/S virtuels –** Pris en charge au moins dans le SE Oracle Solaris 10 10/08
- **Unités cryptographiques –** Prises en charge au moins dans le SE Oracle Solaris 10 8/11
- **Mémoire –** Prise en charge à partir de la version Oracle VM Server for SPARC 2.0 (voir "Utilisation de la reconfiguration dynamique de la mémoire" à la page 228)
- **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 le redémarrage du SE
- Après un arrêt ou un démarrage d'un domaine logique

Les opérations de reconfiguration retardées sont limitées au domaine de contrôle. Pour tous les autres domaines, vous devez arrêter le domaine pour modifier la configuration, à moins que les ressources ne puissent être reconfigurées de manière dynamique.

Lorsqu'une reconfiguration retardée est en cours sur le domaine de contrôle, les autres demandes de reconfiguration du domaine de contrôle sont retardées jusqu'à la réinitialisation de celui-ci, ou à son arrêt et son redémarrage.

La commande `ldm cancel-reconf` annule les opérations de reconfiguration retardée sur le domaine de contrôle. 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 "Suppression d'un grand nombre de CPU d'un domaine de contrôle" du manuel *Notes de version d'Oracle VM Server for SPARC 2.2*. 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 230.

Allocation des ressources

A partir de la version Oracle VM Server for SPARC 2.0, 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.

Allocation de CPU

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 alloués 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 est dynamique, ce qui signifie qu'elle 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.

Remarque – La propriété `max-cores` *ne peut pas* être modifiée, à moins d'arrêter et de dissocier le domaine, ou encore de lancer une reconfiguration retardée sur le domaine de contrôle. Par conséquent, pour augmenter le nombre maximal de coeurs spécifié lors de l'activation de la contrainte whole-core, vous devez tout d'abord arrêter et dissocier le domaine.

▼ Procédure d'application de la contrainte whole-core

Assurez-vous que la contrainte whole-core est activée sur le domaine de contrôle avant de définir la contrainte max-cores.

1 Appliquez la contrainte whole-core sur le domaine `primary`.

```
# ldm set-core 1 primary
```

2 Assurez-vous que la contrainte whole-core est activée sur le domaine de contrôle.

```
# ldm ls -o re primary
```

Notez que `max-cores` est défini sur `unlimited`. Comme n'importe quel autre domaine, le domaine de contrôle ne peut pas être utilisé avec le partitionnement forcé tant que la contrainte max-cores n'est pas activée.

3 Déclenchez une reconfiguration retardée sur le domaine `primary`.

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

4 Activez la contrainte max-cores sur le domaine `primary`.

```
# ldm set-domain max-cores=number-of-CPU-cores primary
```

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”.

Vous ne pouvez désactiver la contrainte max-cores que sur un domaine inactif, non pas sur un domaine associé ou actif. Avant de désactiver la contrainte max-cores sur le domaine de contrôle, vous devez d'abord déclencher une reconfiguration retardée.

5 Assurez-vous que la contrainte whole-core est activée.

```
# ldm ls -o re primary
```

6 Redémarrez le domaine `primary`.

```
# reboot
```

Lors de la réinitialisation, vous pouvez utiliser le domaine de contrôle avec un partitionnement forcé.

Exemple 10–1 Application de la contrainte whole-core

L'exemple suivant illustre l'application de la contrainte whole-core au domaine `primary`. La première commande applique la contrainte, tandis que la deuxième commande vérifie qu'elle est activée :

```
# ldm set-core 1 primary
# ldm ls -o re primary
NAME
primary

CONSTRAINT
    cpu=whole-core
    max-cores=unlimited
    threading=max-throughput
```

Les commandes suivantes limitent max-cores à trois coeurs en déclenchant une reconfiguration retardée, en définissant la propriété `max-cores` et en vérifiant que la contrainte est activée :

```
# 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.

# ldm set-domain max-cores=3 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.
--------------------------------------------------------------------------------

# ldm ls -o re primary
NAME
primary

FLAGS
normal,delayed(modify),control,vio-service

CONSTRAINT
    cpu=whole-core
    max-cores=3
    threading=max-throughput
```

Lors de la réinitialisation, vous pouvez utiliser le domaine de contrôle avec un partitionnement forcé.

L'exemple suivant supprime la contrainte max-cores, mais conserve la contrainte whole-core sur le domaine `ldg1` :

```
# ldm set-domain max-cores=unlimited ldg1
```

Pour supprimer à la fois la contrainte max-cores et la contrainte whole-core du domaine `ldg1`, affectez des CPU virtuelles au lieu de coeurs de la manière suivante :

```
# ldm set-vcpu 8 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 212
- "Gestion dynamique des ressources" à la page 212
- "Migration de domaine" à la page 213
- "Gestion de l'alimentation" à la page 213

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 maximum 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 n'est pas compatible avec la gestion dynamique des ressources (DRM). Si une stratégie DRM est activée sur un domaine qui utilise la contrainte whole-core, cette stratégie est automatiquement désactivée. La contrainte whole-core reste activée.

Même si une stratégie DRM ne peut pas être activée lorsqu'une contrainte whole-core est en vigueur, vous pouvez toujours définir un stratégie DRM pour le domaine. Notez que lorsqu'une stratégie est désactivée automatiquement, elle reste toujours active. La stratégie est automatiquement réactivée si le domaine est redémarré sans la contrainte whole-core.

Voici les interactions attendues entre la contrainte whole-core et la DRM :

- Si la contrainte whole-core est définie sur un domaine, un message d'avertissement est émis lorsque vous essayez d'activer une stratégie DRM sur ce domaine.
- Si une stratégie DRM est en vigueur sur un domaine inactif, vous êtes autorisé à activer la contrainte whole-core sur ce domaine. Lorsque le domaine devient actif et que la stratégie est activé, le système désactive automatiquement la stratégie DRM pour ce domaine.
- Si une stratégie DRM est activée sur un domaine actif ou lié, vous n'êtes pas autorisé à activer la contrainte whole-core.

Migration de domaine

La configuration whole-core de la CPU n'est pas compatible avec la migration de domaine. Cependant, vous pouvez toujours migrer un domaine qui est configuré avec des coeurs complets de CPU. Après une telle migration, le partitionnement forcé n'est pas appliqué sur le système cible. De même, la configuration whole-core et le nombre maximal de coeurs de CPU ne sont pas conservés sur le système cible.

Si vous migrez un domaine configuré avec des coeurs complets, vous devez reconfigurer le domaine cible de manière à ce qu'il utilise le partitionnement forcé une fois la migration terminée. Vous devez également vous assurer que votre contrat de licence vous autorise à utiliser le domaine à la fois sur le système source et sur le système cible.

Gestion de l'alimentation

La contrainte whole-core est parfaitement compatible avec les stratégies performance et élastique de gestion de l'alimentation (PM). Lorsque la stratégie élastique est activée, le sous-système PM peut ajouter ou supprimer des coeurs de CPU dans des domaines configurés avec la contrainte whole-core. Dans ce cas, la contrainte whole-core continue à être honorée, et les domaines utilisant cette contrainte restent configurés avec des coeurs complets uniquement.

Réglage de la CPU SPARC afin d'optimiser les performances de charge de travail sur les systèmes SPARC T4

Vous pouvez utiliser des contrôles de thread de CPU dynamiques pour optimiser les performances de charge de travail sur les systèmes SPARC T4.

Ces contrôles de thread vous permettent de spécifier le nombre de threads matériels à activer par coeur. Les applications existantes peuvent tirer parti des avantages en termes de performances des threads dynamiques pour les CPU SPARC sans requérir de réécriture ou de recompilation.

Cette section indique comment utiliser les contrôles de thread de CPU afin d'optimiser les performances de la CPU sur les systèmes SPARC T4. Les performances de la CPU peuvent être optimisées et le débit maximisé en réglant les coeurs de CPU de manière à ce qu'ils utilisent le

plus grand nombre de threads de CPU possibles. Par défaut, la CPU est réglée de manière à assurer un débit maximal. Les performances de la CPU peuvent également être optimisées pour les charges de travail liées à la CPU en réglant les coeurs de CPU de manière à ce qu'ils maximisent le nombre d'instructions par cycle (IPC).

Modes de thread de la CPU et charges de travail

Sur les systèmes SPARC T4, vous pouvez optimiser les performances de la CPU en spécifiant le mode de thread de la CPU. Le mode de thread peut être défini de manière dynamique et indépendante pour chaque domaine sur le système. Aucune réinitialisation n'est requise pour modifier le mode de thread, et le mode défini persiste lors des réinitialisations du domaine et des cycles d'alimentation de la plate-forme.

En sélectionnant le mode de thread de la CPU approprié, vous pouvez améliorer les performances des applications et des charges de travail qui s'exécutent sur un domaine. Vous pouvez opter pour un mode de thread qui optimise le débit ou un mode de thread qui maximise le nombre d'instructions par cycle :

- **Optimisation du débit (`max-throughput`).** Un débit élevé bénéficie avant tout aux charges de travail qui exécutent un grand nombre de logiciels et effectuent un grand nombre d'opérations d'E/S. Lorsque vous optimisez en vue d'un débit maximal, vous permettez aux coeurs de CPU d'exécuter simultanément autant de threads matériels que possible. Ce mode est idéal pour exécuter des charges de travail d'applications mixtes ou des charges de travail à niveau multithread élevé, telles que celles effectuées par les serveurs Web, les serveurs de bases de données et les serveurs de fichiers. Ce mode est utilisé par défaut ainsi que sur d'anciennes plates-formes SPARC de série T, telles que les plates-formes SPARC T3.
- **Optimisation de l'IPC (`max-ipc`).** Un IPC élevé bénéficie avant tout aux charges de travail consistant en des applications liées à la CPU et à thread unique, telles que des systèmes exécutant des calculs arithmétiques de manière intensive. Lorsque vous optimisez l'IPC, vous permettez à un thread de CPU d'exécuter un plus grand nombre d'instructions par cycle de CPU. Cette optimisation est obtenue au moyen d'une réduction du nombre de threads de CPU actifs simultanément sur le même coeur de CPU.

Sélection du mode de thread de la CPU

Sélectionnez le mode de thread de la CPU d'un domaine à l'aide de la commande `ldm add-domain` ou `ldm set-domain` pour définir la propriété `threading`.

`ldm add-domain [threading=max-throughput|max-ipc]` *ldom*

`ldm set-domain [threading=max-throughput|max-ipc]` *ldom*

La propriété `threading` permet de modifier de façon dynamique le mode de thread en spécifiant l'une des valeurs suivantes :

- `max-throughput`. Utilisez cette valeur pour sélectionner le mode de thread qui optimise le débit. Ce mode active tous les threads assignés au domaine. Ce mode est utilisé par défaut et est sélectionné si vous ne spécifiez aucun mode (`threading=`).
- `max-ipc`. Utilisez cette valeur pour sélectionner le mode de thread qui optimise le nombre d'instructions par cycle (IPC). Lorsque vous utilisez ce mode sur la plate-forme SPARC T4, un seul thread est actif pour chaque coeur de CPU assigné au domaine. Pour que ce mode puisse être sélectionné, il faut que le domaine soit configuré avec la contrainte whole-core.

  Utilisez la commande `ldm add-core` ou la commande `ldm set-core` pour configurer la contrainte whole-core. Reportez-vous à la page de manuel `ldm(1M)`.

Notez que la modification dynamique du mode de thread active ou désactive des threads de CPU. Par conséquent, le nombre de CPU virtuelles disponibles dans le domaine change également de façon dynamique.

Le mode de thread `max-ipc` s'appuie sur la contrainte whole-core, donc vous devez vous conformer aux exigences et aux restrictions de la contrainte whole-core pour effectuer les opérations suivantes :

- Modifier le nombre de coeurs alloués à un domaine.
- Activer ou désactiver la contrainte whole-core.

Par conséquent, pour changer de façon dynamique le mode de thread d'un domaine en cours d'exécution en `max-ipc`, vous devez configurer le domaine avec la contrainte whole-core.

Pour plus d'informations sur les restrictions, reportez-vous à la section “Restrictions des contrôles de thread” à la page 216. Pour plus d'informations sur les sous-commandes `add-domain` et `set-domain`, reportez-vous à la page de manuel `ldm(1M)`.

Affichage de la valeur de la propriété threading

Vous pouvez utiliser la commande suivante pour afficher la valeur de la propriété `threading` :

- La commande `ldm list -o resmgmt` affiche les contraintes. Dans l'exemple de sortie suivant, la propriété `threading` est définie sur `max-ipc` :

```
# ldm list -o resmgmt ldg1
NAME
ldg1
CONSTRAINT
whole-core
max-cores=3
threading=max-ipc
```

- La commande `ldm list -o cpu` affiche les CPU virtuelles désactivées en indiquant la valeur `0` dans la colonne `UTIL`. Le texte en gras dans l'exemple de `max-ipc` suivant montre qu'un seul thread est activé par CPU :

```
# ldm list -o cpu ldg1
NAME
ldg1
VCPU
VID PID CID UTIL STRAND
0    8  1   0.3% 100%
1    9  1   0    100%
2   10  1   0    100%
3   11  1   0    100%
4   12  1   0    100%
5   13  1   0    100%
6   14  1   0    100%
7   15  1   0    100%
8   24  2   0.4% 100%
...
```

- La commande `ldm list -l` inclut toutes les informations relatives au domaine spécifié. Le texte en gras dans l'exemple suivant montre que la propriété `threading` est définie sur `max-ipc` :

```
# ldm list -l ldg1
...
VID PID CID UTIL STRAND
0    8  1   0.6% 100%
1    9  1   0    100%
2   10  1   0    100%
3   11  1   0    100%
4   12  1   0    100%
5   13  1   0    100%
6   14  1   0    100%
...
CONSTRAINT
whole-core
max-cores=3
threading=max-ipc
...
```

Restrictions des contrôles de thread

La fonctionnalité de contrôle de thread est soumise aux restrictions suivantes :

- Les restrictions de la contrainte whole-core s'appliquent. Reportez-vous à la section “Allocation de CPU” à la page 209.
- La valeur de la propriété `threading` n'est pas conservée en cas de migration du domaine.
- La propriété `threading` ne peut pas être définie sur `max-ipc` lorsque de la gestion de l'alimentation (PM) est activée.

Lorsque la PM est en cours d'exécution, la propriété `threading` de tous les domaines doit être définie sur `max-throughput`.

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.

- **Coeurs et threads de CPU.** Le logiciel Oracle VM Server for SPARC s'exécute sur les serveurs SPARC de série T d'Oracle utilisant des processeurs SPARC de série T. Les processeurs SPARC de série T sont équipés de plusieurs coeurs de CPU dont chacun contient plusieurs threads de CPU.
- **Partitionnement forcé et coeurs complets de CPU.** A partir de la version Oracle VM Server for SPARC 2.0, 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 fournit au domaine le nombre de coeurs de CPU spécifié et tous ses threads de CPU. 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, vous devez utiliser au moins la version 2.0 d'Oracle VM Server for SPARC. Vous devez également utiliser des coeurs complets de CPU de la manière suivante :
  - Si un domaine exécute des applications utilisant l'octroi de licence de partitionnement forcé d'Oracle, ce domaine doit être configuré avec des coeurs complets de CPU.
  - Si un domaine n'exécute aucune application utilisant l'octroi de licence de partitionnement forcé d'Oracle, il n'est pas nécessaire de configurer ce domaine 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

Les tâches décrites dans cette section expliquent comment déterminer si un domaine est configuré avec des coeurs complets de CPU et comment dresser la liste des coeurs de CPU affectés à un domaine.

▼ Procédure permettant de déterminer si un domaine a été configuré avec des coeurs complets de CPU

- **Déterminez si le domaine est configuré avec des coeurs complets de CPU.**

  ```
  # ldm list -o resmgmt domain
  ```

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)`.

Exemple 10–2 Détermination de la configuration avec ou sans coeurs complets de CPU d'un domaine

La commande suivante indique que le domaine `ldg1` est configuré avec des coeurs complets et avec un maximum de cinq coeurs :

```
# ldm list -o resmgmt ldg1
NAME
ldg1

CONSTRAINT
   whole-core
   max-cores=5
```

▼ Procédure de création de la liste des coeurs de CPU affectés à un domaine

Lorsqu'un domaine est associé, des coeurs de CPU sont affectés au domaine.

- **Etablissez la liste des coeurs de CPU affectés à un domaine.**

  `# ldm list -o core` *domain*

Exemple 10–3 Etablissement de la liste des coeurs de CPU affectés à un domaine

La commande suivante dresse la liste des coeurs affectés au domaine `ldg1` :

```
# 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.

Remarque – Les sous-commandes `ldm` utilisées pour affecter des coeurs complets ont été modifiées dans la version Oracle VM Server for SPARC 2.2.

Les tâches et les exemples de cette section font appel aux nouvelles commandes introduites avec le logiciel Oracle VM Server for SPARC 2.2.

Si vous utilisez la version 2.0 ou 2.1 de Logical Domains Manager pour affecter des coeurs complets à des domaines, utilisez les commandes `ldm add-vcpu -c`, `ldm set-vcpu -c` et `ldm remove-vcpu -c` respectivement au lieu des commandes `ldm add-core`, `ldm set-core` et `ldm remove-core`.

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, lequel correspond au plafond de CPU. Reportez-vous à la page de manuel `ldm(1M)`.

A partir de la version 2.2 d'Oracle VM Server for SPARC, le plafond de CPU et l'allocation de coeurs de CPU sont gérées 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 plafond de CPU 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 plafond de CPU 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

1 Créez le domaine.

```
# ldm create domain
```

2 Définissez le nombre de coeurs complets de CPU pour le domaine.

```
# ldm set-core number-of-cpu-cores domain
```

Cette commande définit également le nombre maximal de coeurs de CPU du domaine sur *number-of-cpu-cores*.

3 Configurez le domaine.

Pendant la configuration, assurez-vous d'utiliser la commande `ldm add-core`, `ldm set-core` ou `ldm rm-core`.

4 Associez et démarrez le domaine.

```
# ldm bind domain
# ldm start domain
```

Exemple 10–4 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. Elle fixe également à 2 le nombre maximal de coeurs de CPU de `ldg1`.

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 219.

Les troisième et quatrième commandes indiquent comment associer et démarrer le domaine `ldg1`, suite à quoi vous pouvez utiliser le domaine `ldg1`.

```
# ldm create ldg1
# ldm set-core 2 ldg1
...
# ldm bind ldg1
# ldm start ldg1
```

▼ Procédure de configuration d'un domaine existant avec des coeurs complets de CPU

Si un domaine existant est configuré pour utiliser des threads de CPU, vous pouvez modifier sa configuration et paramétrer l'utilisation de coeurs complets de CPU.

1 Arrêtez et dissociez le domaine.

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

2 Définissez le nombre de coeurs complets de CPU pour le domaine.

```
# ldm set-core number-of-cpu-cores domain
```

Cette commande définit également le nombre maximal de coeurs de CPU du domaine sur *number-of-cpu-cores*.

3 Associez à nouveau et redémarrez le domaine.

```
# ldm bind domain
# ldm start domain
```

Exemple 10–5 Configuration d'un domaine existant avec quatre coeurs complets de CPU

Cet exemple met à jour la configuration d'un domaine existant, `ldg1`. Les première et deuxième commandes arrêtent et dissocient le domaine `ldg1`. La troisième commande configure le domaine `ldg1` avec quatre coeurs complets de CPU. Cette commande fixe également à quatre le nombre maximal de coeurs de CPU du domaine `ldg1`. Les quatrième et cinquième commandes associent et redémarrent le domaine `ldg1`.

```
# ldm stop ldg1
# ldm unbind ldg1
# ldm set-core 4 ldg1
# ldm bind ldg1
# ldm start 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 et paramétrer l'utilisation de coeurs complets de CPU.

1 Placez le domaine `primary` en mode de reconfiguration retardée.

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

2 Définissez le nombre de coeurs complets de CPU pour le domaine `primary`.

```
# ldm set-core number-of-cpu-cores primary
```

Cette commande définit également le nombre maximal de coeurs de CPU du domaine `primary` sur *number-of-cpu-cores*.

3 Redémarrez le domaine `primary`.

Utilisez la procédure appropriée pour réinitialiser le domaine `primary` ; celle-ci dépend de la configuration système. Reportez-vous à la section "Redémarrage du domaine `primary`" à la page 91

Exemple 10–6 Configuration du domaine primary 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. Cette commande fixe également à deux le nombre maximal de coeurs de CPU du domaine `primary`. La troisième commande réinitialise le domaine `primary`.

```
# ldm start-reconf primary
# ldm set-core 2 primary
# shutdown -i 5
```

Interaction avec d'autres fonctions d'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 *uniquement* ajouter ou supprimer des coeurs de CPU entiers, pas des threads de CPU individuels. Ainsi, 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 CPU DR 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, définir ou supprimer de façon dynamique des coeurs complets de CPU dans un domaine associé ou actif :

`ldm add-core` *number-of-cpu-cores domain*
`ldm set-core` *number-of-cpu-cores domain*
`ldm rm-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 10–7 Ajout dynamique de deux coeurs complets de CPU à un domaine

Cet exemple illustre l'ajout dynamique de 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`.

EXEMPLE 10–7 Ajout dynamique de deux coeurs complets de CPU à un domaine *(Suite)*

```
# ldm list ldg1
NAME    STATE   FLAGS   CONS  VCPU  MEMORY UTIL  UPTIME
ldg1    active  -n----  5000  16    2G     0.4%  5d 17h 49m
# ldm list -o resmgmt ldg1
NAME
ldg1

CONSTRAINT
   whole-core
        max-cores=4
# 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)
# ldm add-core 2 ldg1
# 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. Si la DRM est utilisée, les stratégies DRM ne s'appliquent *pas* aux domaines configurés avec des coeurs complets de CPU.

Une stratégie DRM peut inclure un domaine configuré avec des coeurs complets de CPU. Toutefois, lorsqu'une telle stratégie est activée, elle est automatiquement désactivée pour le domaine concerné. Le domaine reste configuré avec des coeurs complets de CPU, à moins que le domaine ne soit ultérieurement reconfiguré avec des threads de CPU au lieu de coeurs complets. Lorsque le domaine est configuré pour l'utilisation de threads de CPU, la stratégie DRM est automatiquement réactivée pour le domaine concerné.

Gestion de l'alimentation de la CPU

Vous pouvez utiliser la gestion de l'alimentation (PM) de la CPU en activant la politique performance ou élastique pour des domaines configurés avec des coeurs complets de CPU. L'utilisation de la PM préserve le partitionnement forcé du système.

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.

Incompatibilité de la migration de domaine

Une configuration whole-core n'est pas compatible avec une migration de domaine. Cependant, vous pouvez toujours migrer un domaine qui est configuré avec des coeurs complets de CPU. Après une telle migration, le partitionnement forcé n'est pas appliqué sur le système cible. De même, la configuration whole-core et le nombre maximal de coeurs de CPU ne sont pas conservés sur le système cible lors de la migration.

Si vous migrez un domaine configuré avec des coeurs complets, vous devez reconfigurer le domaine cible de manière à ce qu'il utilise le partitionnement forcé une fois la migration terminée. Vous devez également vous assurer que votre contrat de licence vous autorise à utiliser le domaine à la fois sur le système source et sur le système cible.

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 .2.2 vous permet également de sélectionner de façon explicite les ressources physiques à affecter ou dont annuler l'affectation à un domaine. Cette fonctionnalité est *uniquement* disponible lorsque le domaine de contrôle exécute le SE Oracle Solaris 11.

Les ressources affectées de manière explicite sont appelées *ressources nommées*. Les ressources affectées de manière automatique sont appelées *ressources anonymes*.

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 226. 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 227.

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[,...]] ldom
  # ldm set-core cid=core-ID[,core-ID[,...]] ldom
  # ldm rm-core [-f] cid=core-ID[,core-ID[,...]] ldom
  ```

  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.

- **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[,...]] ldom
  # ldm set-mem mblock=PA-start:size[,PA-start:size[,...]] ldom
  # ldm rm-mem mblock=PA-start:size[,PA-start:size[,...]] ldom
  ```

  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 non associé. 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 226.

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. Lorsque la propriété `mblock` est définie, la contrainte `physical-bindings` est définie

sur `memory`. De même, lorsque la propriété `cid` est définie, la contrainte `physical-bindings` est définie sur `core`. Lorsque les propriétés `cid` et `mblock` sont toutes deux définies, la contrainte `physical-bindings` est définie sur `core, memory`.

Pour modifier la contrainte `physical-bindings` pour un type de ressources donné sur le domaine de contrôle, vous *devez* tout d'abord supprimer toutes les ressources du type concerné en définissant le nombre de ressources sur zéro de la manière suivante :

- Définissez sur 0 le nombre de ressources à l'aide de la commande `ldm set-core 0` ou de la commande `ldm set-mem 0`.
- Supprimez toutes les contraintes `physical-bindings` spécifiées pour un type de ressources donné.

  Pour supprimer tous les coeurs et blocs de mémoire nommés, exécutez respectivement les commandes `ldm set-core cid=` et `ldm set-mem mblock=`. Pour supprimer tous les coeurs et blocs de mémoire anonymes, exécutez respectivement les commandes `ldm set-core 0` et `ldm set-mem 0`.

  L'allocation de CPU et de mémoire au domaine de contrôle étant *obligatoire*, une erreur est renvoyée lorsque `cid=` ou `mblock=` est spécifié sur le domaine de contrôle.
- Supprimez une à une les ressources du domaine.

Gestion des ressources physiques sur le domaine de contrôle

Etant donné que le domaine de contrôle est toujours actif, il peut être en mode de reconfiguration retardée avant que vous ne procédiez à des affectations de ressources physiques. Lorsque vous assignez des ressources physiques de manière explicite, le domaine de contrôle est automatiquement placé en mode de reconfiguration retardée et la contrainte `physical-bindings` est définie.

Si `physical-bindings=core`, l'exécution de la commande `ldm set-core cid=`*core-ID* `primary` ou de la commande `ldm set-vcpu` *CPU-count* `primary` entraîne l'effacement de la contrainte `physical-bindings` lors de la réinitialisation suivante. Si la contrainte `physical-bindings` n'est pas définie sur `core`, exécutez la commande `ldm set-core cid=`*core-ID* `primary` pour définir `physical-bindings=core` lors de la réinitialisation suivante.

Si `physical-bindings=memory`, l'exécution de la commande `ldm set-mem` *size* `primary` entraîne l'effacement de la contrainte `physical-bindings` lors de la réinitialisation suivante. Si la contrainte `physical-bindings` n'est pas définie sur `memory`, exécutez la commande `ldm set-mem mblock=`*PA-start*`:`*size* `primary` pour définir la contrainte `physical-bindings` lors de la réinitialisation suivante.

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 individuelles dans un domaine où `physical-bindings=core` est vouée à l'échec.
- L'allocation et la vérification des ressources non liées 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 *ne pouvez pas* utiliser la commande `ldm add-core` ou `ldm set-core` pour affecter une ressource physique à un domaine.
- Si vous utilisez la commande `ldm add-mem` ou la commande `ldm set-mem` pour affecter plusieurs blocs de mémoire physique, les adresses et les tailles sont immédiatement vérifiées.
- 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 version Oracle VM Server for SPARC 2.0 introduit la reconfiguration dynamique de la mémoire (DR). Cette fonction repose sur la capacité et vous permet d'ajouter une quantité arbitraire de mémoire à un domaine logique actif ou d'en supprimer.

Voici les contraintes et les restrictions d'utilisation de la fonction de DR 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. Reportez-vous à la section "Alignement de la mémoire" à la page 230.
- 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 DR de mémoire. Reportez-vous à la section "Ajout de mémoire non alignée" à la page 231.

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.

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 opération ou annuler une demande de DR de mémoire en cours.

Suivi de la progression d'une demande de reconfiguration dynamique de mémoire

Vous pouvez suivre la progression d'une commande `ldm remove-memory` en exécutant la commande `ldm list -l` pour le domaine indiqué.

Annulation d'une demande de reconfiguration dynamique de mémoire

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

Une demande d'ajout de mémoire est rejetée s'il n'y a pas suffisamment de mémoire libre pour remplir toute la demande. Cependant, une demande d'ajout de mémoire peut être partiellement remplie si le domaine cible ne parvient pas à ajouter une partie de la mémoire demandée par Logical Domains Manager.

Une demande de suppression de mémoire est rejetée s'il n'y a pas suffisamment de mémoire dans le domaine pour remplir toute la demande. Cependant, une demande de suppression de mémoire peut être partiellement remplie si le domaine cible ne parvient pas à supprimer une partie de la mémoire demandée par Logical Domains Manager.

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

La fonction de reconfiguration dynamique (DR) de la mémoire peut être utilisée 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 un domaine actif n'est pas sûr d'ajouter ou plus généralement de donner 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. Redémarrez 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. Une demande de reconfiguration de la mémoire sur un domaine de contrôle qui ne prend pas en charge la reconfiguration dynamique de la mémoire 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. Cela signifie que 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. Cela signifie que 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 231 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. Cela signifie que la mémoire non alignée dans un domaine actif ne peut pas être supprimée à l'aide de la reconfiguration dynamique de la mémoire.

Cela signifie également que 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. Les commandes suivantes ajoutent les deux blocs de mémoire restants. La première commande lance une opération de reconfiguration retardée sur le domaine de contrôle. La seconde commande ajoute les blocs de mémoire de 128 Mo au domaine de contrôle. La cinquième commande ajoute l'autre bloc de mémoire de 128 Mo au domaine `ldom1`.

```
# 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.

# 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.
--------------------------------------------------------------------------------

# ldm list
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -ndcv-  SP      8     2688M    0.1%  23d 8h 8m

# 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

# ldm add-mem 128M ldom1
# 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 10–8 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`.

EXEMPLE 10–8 Opérations de reconfiguration dynamique de la mémoire sur des domaines actifs *(Suite)*

La sortie `ldm list` montre la mémoire pour chaque domaine dans la zone Memory. La première commande `ldm add-mem` existe avec 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.

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.

```
# 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

# ldm add-mem 200M ldom1
The size of memory must be a multiple of 256MB.

# 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.

# 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

# 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.

# 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 10–9 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`.

EXEMPLE 10–9 Opérations de reconfiguration dynamique de la mémoire sur des domaines liés *(Suite)*

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.

```
# 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

# ldm add-mem 100M ldom2

# 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

# 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.

# 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

# ldm rm-mem 100M ldom2
# 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

# 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.

# 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 10–10 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. La première commande `ldm set-mem` tente de définir la taille du domaine `primary` à 3 400 mégaoctets. 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.

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.

```
# 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

# 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.

# 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.

# ldm set-mem 690M ldom2
# 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

# 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.

# 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 la gestion de l'alimentation

Pour utiliser 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 :

- Contrôle de la consommation d'énergie du *Sun 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 :

- **Performance.** Le système est autorisé à utiliser la totalité de l'énergie disponible.
- **Elastique.** L'utilisation de l'énergie par le système est adaptée au niveau d'utilisation actuel. Par exemple, l'état d'alimentation des ressources est réduit lorsque l'utilisation diminue.

Voici les fonctions PM :

- **Auto-désactivation des coeurs de CPU.** La PM désactive automatiquement un coeur de CPU lorsque tous les threads (strands) de ce coeur ont été désactivés.
- **Saut du cycle d'horloge de la CPU.** A partir de la version Oracle VM Server for SPARC 2.0, la PM peut ajuster automatiquement le saut du cycle d'horloge de la CPU. Cette fonctionnalité est uniquement disponible pour les plates-formes SPARC T3 et SPARC T4. L'ajustement peut augmenter ou diminuer le nombre de cycles d'horloge qui sont ignorés pour maintenir tous les domaines dans les seuils d'utilisation d'énergie. Le mode PM détermine s'il faut faire des ajustements en fonction de l'utilisation de la CPU. Lorsque le système applique la stratégie performance, le nombre de cycles d'horloge ignorés est automatiquement ajusté à aucun.
- **Opérations de mémoire en mode veille profonde.** A partir de la version Oracle VM Server for SPARC 2.0, lorsque la stratégie élastique est en vigueur sur les plates-formes SPARC T3 et SPARC T4, celles-ci configurent automatiquement la mémoire sous-utilisée pour fonctionner en mode veille profonde pour économiser de l'énergie.
- **Limite de puissance.** Vous pouvez définir une *limite de puissance* sur les plates-formes SPARC T3 et SPARC T4 pour restreindre l'énergie consommée par un système. Si l'énergie consommée est supérieure à la limite de puissance, les techniques PM sont utilisées pour réduire la consommation. Vous pouvez utiliser le processeur de service (SP) d'ILOM pour définir la limite de puissance.

Consultez les documents suivants :

- *Sun 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 ressources en mode de gestion d'alimentation. 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.

Lorsque la stratégie élastique est en vigueur sur un système, certaines modifications de la configuration du domaine sont d'abord validées pour confirmer que la limite de puissance n'est pas dépassée. Si la limite de puissance est dépassée, seules certaines ressources peuvent être modifiées ou ajoutées comme demandé. Si la limite de puissance est augmentée ultérieurement, vous pouvez ensuite ajouter les ressources ayant été modifiées sans succès.

Si la charge du domaine fait que les ressources consomment plus d'énergie, seules les ressources pouvant maintenir la consommation inférieure à la limite de puissance sont alimentées avec succès.

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 au contrôle de la consommation d'énergie du *Sun Integrated Lights Out Manager (ILOM) 3.0 CLI Procedures Guide*.

Création d'une liste des threads de CPU et des CPU virtuelles avec gestion de l'alimentation

Cette section indique comment dresser la liste des threads et des CPU virtuelles avec gestion de l'alimentation.

▼ Procédure de création de la liste des threads de CPU avec gestion de l'alimentation

- **Répertoriez les threads avec gestion de l'alimentation à l'aide de l'une des commandes suivantes :**

  - **Utilisez la sous-commande `list -l`.**

```
# ldm list -l primary
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
primary          active     -n-cv-  UART    64    16G      1.0%  21h 33m

SOFTSTATE
Solaris running

UUID
    b9288150-327f-44f7-8c64-d4d57b92e524

MAC
    00:21:28:8f:8f:34

HOSTID
    0x858f8f34

CONTROL
    failure-policy=ignore

DEPENDENCY
    master=

CORE
    CID    CPUSET
    0      (0, 1, 2, 3, 4, 5, 6, 7)
    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)
    5      (40, 41, 42, 43, 44, 45, 46, 47)
    6      (48, 49, 50, 51, 52, 53, 54, 55)
    7      (56, 57, 58, 59, 60, 61, 62, 63)

VCPU
    VID    PID    CID    UTIL STRAND
    0      0      0      1.2%   100%
    1      1      0      1.5%   100%
    2      2      0      0.1%   100%
    3      3      0      0.2%   100%
.
.
.
```

La sortie suivante affiche des tirets (`---`) dans la colonne `UTIL` de la CPU, ce qui signifie que le thread est en mode de gestion d'alimentation. Les tirets s'affichent uniquement pour les domaines autres que le domaine `primary`.

```
# ldm list -l ldg1
NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
```

```
ldg1             active     -n--v-  5000    64    16G      1.1%  20h 55m

SOFTSTATE
Solaris running

UUID
    98d86371-24f6-4792-c631-eb14e81ad4a0

MAC
    00:14:4f:f9:02:f2

HOSTID
    0x84f902f2

CONTROL
    failure-policy=ignore

DEPENDENCY
    master=
CORE
    CID    CPUSET
    8      (64, 65, 66, 67, 68, 69, 70, 71)
    9      (72, 73, 74, 75, 76, 77, 78, 79)
    10     (80, 81, 82, 83, 84, 85, 86, 87)
    11     (88, 89, 90, 91, 92, 93, 94, 95)
    12     (96, 97, 98, 99, 100, 101, 102, 103)
    13     (104, 105, 106, 107, 108, 109, 110, 111)
    14     (112, 113, 114, 115, 116, 117, 118, 119)
    15     (120, 121, 122, 123, 124, 125, 126, 127)

VCPU
    VID    PID    CID    UTIL STRAND
    0      64     8      0.8%   100%
    1      65     8      2.0%   100%
    2      66     8      ----   100%
    3      67     8      ----   100%
    4      68     8      ----   100%
.
.
.
```

- **Utilisez l'option d'analyse (`-p`) avec la sous-commande `list -l`.**

  Dans la sortie, un espace après `util=` signifie que le thread (strand) est en mode de gestion d'alimentation.

  ```
  # ldm list -l -p

  VCPU
  |vid=0|pid=0|util=0.7%|strand=100
  |vid=1|pid=1|util=|strand=100
  |vid=2|pid=2|util=|strand=100
  |vid=3|pid=3|util=|strand=100
  |vid=4|pid=4|util=0.7%|strand=100
  |vid=5|pid=5|util=|strand=100
  |vid=6|pid=6|util=|strand=100
  |vid=7|pid=7|util=|strand=100
  ```

▼ Procédure de création de la liste des CPU avec gestion de l'alimentation

- **Répertoriez les CPU en mode de gestion d'alimentation à l'aide des commandes suivantes :**

  a. **Utilisez la commande `list-devices -a cpu`.**

     Dans la colonne `PM` de la sortie, un `yes` signifie que la CPU est en mode de gestion de l'alimentation, un `no` indique que la CPU est sous tension. On suppose que les CPU libres à 100 % sont en mode de gestion de l'alimentation par défaut, d'où les tirets (`- - -`) sous la colonne `PM` pour eux.

```
# ldm list-devices -a cpu
VCPU
   PID     %FREE      PM
   0       0          no
   1       0          yes
   2       0          yes
   3       0          yes
   4       100        ---
   5       100        ---
   6       100        ---
   7       100        ---
```

  b. **Utilisez l'option d'analyse `-p` sur la sous-commande `list-devices -a cpu`.**

     Dans la colonne `pm=` de la sortie, un `yes` signifie que la CPU est en mode de gestion de l'alimentation, un `no` indique que la CPU est sous tension. Il est supposé que les CPU 100 % libres sont en mode de gestion de l'alimentation par défaut, d'où l'espace dans cette zone pour eux.

```
# ldm list-devices -a -p cpu
VERSION 1.6
VCPU
|pid=0|free=0|pm=no
|pid=1|free=0|pm=yes
|pid=2|free=0|pm=yes
|pid=3|free=0|pm=yes
|pid=4|free=0|pm=no
|pid=5|free=0|pm=yes
|pid=6|free=0|pm=yes
|pid=7|free=0|pm=yes
|pid=8|free=100|pm=
|pid=9|free=100|pm=
|pid=10|free=100|pm=
```

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 :

- Lorsque la stratégie élastique de gestion de l'alimentation est activée, la DRM ne peut pas être activée.
- 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.
- Les stratégies DRM ne s'appliquent pas aux domaines configurés avec la contrainte `whole-core`.

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 ldom...
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 ldom...
ldm remove-policy [name=]policy-name... ldom
```

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` dans 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'appliquent *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 10–11 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

EXEMPLE 10–11 Ajout de stratégies de gestion des ressources *(Suite)*

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 minimum et maximum de CPU virtuelles sont 2 et 16 en définissant les propriétés `vcpu-min` et `vcpu-max` respectivement.
- Indique que le nombre maximum 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 maximum 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)`.

```
# 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 minimum et maximum de CPU virtuelles sont 2 et 16 en définissant les propriétés `vcpu-min` et `vcpu-max` respectivement.
- Indique que le nombre maximum 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 maximum de CPU virtuelles à supprimer au cours d'un cycle de contrôle des ressources est 1 en définissant la propriété `decay`.

EXEMPLE 10–11 Ajout de stratégies de gestion des ressources *(Suite)*

- 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)`.

```
# 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 que vous verrez réellement 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 une formulaire lisible par la machine de la sortie. Reportez-vous à la section "Procédure de génération d'une liste analysable lisible par la machine (`-p`)" à la page 246 pour plus d'informations.

▼ Procédure d'affichage de la syntaxe des sous-commandes `ldm`

- **Regardez l'utilisation de la syntaxe de toutes les sous-commandes `ldm`.**

  ```
  # 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 l'option longue analysable (`-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. Voici les valeurs pouvant apparaître dans chacune des six colonnes de gauche à droite.

Colonne 1

- `s` démarrage ou arrêt
- `-` paramètre substituable

Colonne 2

- `n` normal
- `t` transition

Colonne 3

- `d` reconfiguration retardée
- `r` reconfiguration dynamique (DR) de la mémoire
- `-` paramètre substituable

Colonne 4

- `c` domaine de contrôle
- `-` paramètre substituable

Colonne 5

- `v` domaine de service d'E/S virtuelle
- `-` paramètre substituable

Colonne 6

- `s` domaine source dans une migration
- `t` domaine cible dans une migration
- `e` erreur survenue au cours d'une migration
- `-` paramètre substituable

Définition des statistiques d'utilisation

Les statistiques d'utilisation des CPU virtuelles (`UTIL`) sont affichées sur 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. Un tiret (`---`) dans la colonne `UTIL` signifie que le thread est en mode de gestion de l'alimentation.

Affichage des différentes listes

▼ Procédure d'affichage des versions logicielles (-V)

- **Affiche la version logicielle actuellement installée.**

```
# ldm -V
```

▼ Procédure de génération d'une liste abrégée

- **Génère une liste abrégée de tous les domaines.**

```
# ldm list
```

▼ Procédure de génération d'une liste longue (-l)

- **Générez une liste longue pour tous les domaines.**

```
# ldm list -l
```

▼ Procédure de génération d'une liste étendue (-e)

- **Générez une liste étendue de tous les domaines.**

```
# ldm list -e
```

▼ Procédure de génération d'une liste analysable lisible par la machine (-p)

- **Génère une liste analysable lisible par la machine de tous les domaines.**

```
# ldm list -p
```

▼ Procédure de génération d'un sous-ensemble d'une liste longue (-o *format*)

- **Générez une sortie comme un 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.**

  `# ldm list -o` *resource*`[,`*resource*`...]` *ldom*

  - `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`)
- `domain` – La sortie contient des 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`), le client du canal de domaine logique (`vldcc`), le client du canal de plan de données virtuelles (`vdpcc`), le service du canal de plan de données virtuelles (`vdpcs`)
- `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 suivantes montrent différents sous-ensembles de sortie que vous pouvez définir :

- Liste des informations des CPU pour le domaine de contrôle

  ```
  # ldm list -o cpu primary
  ```
- Liste des informations de domaine pour un domaine invité

  ```
  # ldm list -o domain ldm2
  ```
- Liste des informations sur la mémoire et le réseau pour un domaine invité

  ```
  # ldm list -o network,memory ldm1
  ```
- Liste des informations sur la stratégie DRM pour un domaine invité

  ```
  # ldm list -o resmgmt,stats ldm1
  ```

▼ Procédure d'affichage de la valeur d'une variable

● **Affichez une variable et sa valeur pour un domaine.**

```
# ldm list-variable variable-name ldom
```

Par exemple, la commande suivante affiche la valeur de la variable `boot-device` sur le domaine `ldg1` :

```
# ldm list-variable boot-device ldg1
boot-device=/virtual-devices@100/channel-devices@200/disk@0:a
```

▼ Procédure de création de la liste des liaisons

- **Répertoriez les ressources liées à un domaine.**

  `# ldm list-bindings` *ldom*

▼ Procédure de création de la liste des configurations

- **Répertoriez les configurations de domaine logique ayant été stockées sur le SP.**

Exemple 10–12 Liste des configurations

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 Logical Domains" à la page 255. Pour plus d'exemples, reportez-vous à la page de manuel `ldm(1M)`.

```
# ldm list-config
factory-default
3guests
foo [next poweron]
primary
reconfig-primary
```

Informations supplémentaires Signification des libellés

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 à la prochaine remise sous tension.

▼ Procédure de création de la liste des périphériques

- **Répertoriez toutes les ressources du serveur, liées ou non.**

  `# ldm list-devices -a`

▼ Procédure de création de la liste de mémoire disponible

- **Répertoriez la quantité de mémoire disponible à allouer.**

```
# ldm list-devices mem
MEMORY
    PA                  SIZE
    0x14e000000         2848M
```

▼ Procédure de création de la liste des services

- **Répertoriez les services disponibles.**

```
# 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.

▼ Procédure de création de la liste des contraintes pour un domaine

- **Répertoriez les contraintes pour un domaine.**

```
# ldm list-constraints ldom
```

▼ Procédure de création de la liste des contraintes au format XML

- **Répertoriez les contraintes au format XML pour un domaine particulier.**

```
# ldm list-constraints -x ldom
```

▼ Procédure de création de la liste des contraintes dans un format lisible par machine

- **Répertoriez les contraintes pour tous les domaines dans un format analysable.**

```
# ldm list-constraints -p
```

◆ ◆ ◆ CHAPITRE 11

Gestion des configurations de domaine

Ce chapitre contient des informations sur la gestion des configurations de domaine.

Ce chapitre aborde les sujets suivants :

Enregistrement des configurations de domaine pour régénération ultérieure

Le processus basique consiste à enregistrer les informations de contraintes de ressource pour chaque domaine dans un fichier XML, qui peut être renvoyé au Logical Domains Manager, par exemple, après une panne matérielle pour régénérer la configuration souhaitée.

La procédure décrite à la section "Procédure de restauration d'une configuration de domaine à partir d'un fichier XML (`ldm add-domain`)" à la page 252 fonctionne pour les domaines invités, mais pas pour le domaine de contrôle (`primary`). Vous pouvez enregistrer les contraintes du domaine `primary` dans un fichier XML, mais vous ne pouvez pas réintégrer le fichier dans la commande `ldm add-domain -i`. Cependant, vous pouvez utiliser 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 ces derniers resteront inactifs lorsque la configuration sera terminée.

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. Cela signifie qu'à l'issue de cette procédure, 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.

▼ Procédure d'enregistrement des configurations de domaine

Cette procédure décrit comment enregistrer une configuration de domaine pour un domaine unique ou pour tous les domaines sur un système.

- **Enregistrez la configuration de domaine pour un ou plusieurs domaines.**

  - **Pour enregistrer la configuration pour un seul domaine, créez un fichier XML contenant les contraintes du domaine.**

    ```
    # ldm list-constraints -x ldom >ldom.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
    ```

▼ Procédure de restauration d'une configuration de domaine à partir d'un fichier XML (`ldm add-domain`)

Au lieu de cette procédure, vous pouvez utiliser la commande `ldm init-system` pour restaurer les configurations de domaine à partir d'un 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 253.

1. **Créez le domaine à l'aide du fichier XML créé en entrée.**

   ```
   # ldm add-domain -i ldom.xml
   ```

2. **Associez le domaine.**

   ```
   # ldm bind-domain [-fq] ldom
   ```

   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 ldom
   ```

Exemple 11–1 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 la configuration enregistrée précédemment. Le fichier XML décrit une ou plusieurs configurations de domaine. Le fichier XML peut être créé en exécutant la commande `ldm ls-constraints -x`. La commande `ldm init-system` doit être exécutée dans la configuration `factory-default`, mais elle ne peut pas restaurer de configuration à partir du fichier XML. Le domaine `primary` est reconfiguré comme indiqué dans le fichier, et tous les domaines autres que `primary` ayant des configurations dans le fichier XML sont reconfigurés mais laissés inactifs.

Au lieu de cette procédure, vous pouvez utiliser la commande `ldm add-domain` pour restaurer la configuration d'un seul domaine à partir d'un fichier XML. Reportez-vous à la section "Procédure de restauration d'une configuration de domaine à partir d'un fichier XML (`ldm add-domain`)" à la page 252.

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 43.

3. **Prenez le rôle d'administrateur, de superutilisateur ou un rôle équivalent.**

   Pour Oracle Solaris 10, reportez-vous à la section "Configuración de RBAC (mapa de tareas)" du manuel *Guía de administración del sistema: servicios de seguridad*. Pour Oracle Solaris 11, reportez-vous à la Partie III, "Roles, perfiles de derechos y privilegios" du manuel *Administración de Oracle Solaris: servicios de seguridad*.

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` redémarre 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 redémarrage.

L'option `-f` ignore la vérification 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.

Exemple 11–2 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 le redémarrage 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
  ```

Gestion des configurations de Logical Domains

Une *configuration* de Logical Domains 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.

Lorsque vous mettez un système sous tension, le SP démarre la configuration sélectionnée. En initialisant une configuration, le système exécute le même ensemble de domaines et utilise les mêmes allocations de virtualisation et de partitionnement que celles indiquées dans la configuration. La configuration par défaut est celle qui a été enregistrée le plus récemment.

Une copie de la configuration actuelle est automatiquement enregistrée sur le domaine de contrôle chaque fois que la configuration Logical Domains est modifiée.

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 réelle n'est pas effectuée tant que le domaine concerné n'a pas redémarré

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 lorsque le système a été remis sous tension. Dans ce cas, Logical Domains Manager peut restaurer cette configuration au redémarrage si elle est plus récente que la configuration sélectionnée pour le redémarrage suivant.

Remarque – Les événements de mise à jour de gestion de l'alimentation, FMA, ASR et PRI 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.

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 de démarrage, reportez-vous à la section "Utilisation de Logical Domains avec le processeur de service" à la page 264.

▼ Procédure de modification de la 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`. La propriété `autorecovery_policy` peut avoir 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`. L'utilisateur peut 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. L'utilisateur peut 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 à la prochaine remise sous tension. 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 à la prochaine remise sous tension. Un message est également consigné. Celui-ci indique qu'une configuration plus récente a été

enregistrée sur le SP et qu'elle sera initialisée à la prochaine remise sous tension du système. Ces messages sont consignés dans le fichier journal SMF `ldmd`.

1. **Connectez-vous au domaine de contrôle.**

2. **Prenez le rôle d'administrateur, de superutilisateur ou un rôle équivalent.**

   Pour Oracle Solaris 10, reportez-vous à la section "Configuración de RBAC (mapa de tareas)" du manuel *Guía de administración del sistema: servicios de seguridad*. Pour Oracle Solaris 11, reportez-vous à la Partie III, "Roles, perfiles de derechos y privilegios" du manuel *Administración de Oracle Solaris: servicios de seguridad*.

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 11–3 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
```

◆ ◆ ◆ CHAPITRE 12

Réalisation d'autres tâches d'administration

Ce chapitre contient des informations et des tâches sur l'utilisation du logiciel Oracle VM Server for SPARC qui ne sont pas décrites dans les chapitres précédents.

Ce chapitre aborde les sujets suivants :

Entrée de noms dans la CLI

Les sections suivantes décrivent les restrictions dans l'entrée des noms dans la CLI de Logical Domains Manager.

Noms de fichier (*file*) et noms de variable (*var-name*)

- 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.

backend du serveur de disque virtuel et noms de périphérique de commutateur virtuel

Les noms doivent contenir des lettres, des nombres ou des signes de ponctuation.

Nom de configuration (*config-name*)

Le nom de la configuration de domaine logique (*config-name*) que vous assignez à une configuration stockée sur le processeur de service (SP) doit comporter 64 caractères au maximum.

Tous les autres noms

Le reste des noms, notamment le nom du domaine logique (*ldom*), les noms de service (*vswitch-name*, *service-name*, *vdpcs-service-name*et *vcc-name*), le nom du réseau virtuel (*if-name*) et le nom du disque virtuel (*disk-name*) doivent présenter le format suivant :

- 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 : `-_+#.:;~()`.

Connexion à une console invitée sur un 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 host-name 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 fichier manifest d'utilitaire de gestion des services est un fichier XML qui décrit un service. Pour plus d'informations sur la création d'un fichier manifest 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 dans les paramètres linguistiques 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 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 à l'état inactif. Reportez-vous à la page de manuel `vntsd`(1M) du SE Oracle Solaris 10 pour plus d'informations sur la configuration et l'utilisation de SMF pour gérer `vntsd` et sur l'utilisation de groupes de consoles.

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 suivante 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
```

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 a 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 atteindre l'invite `ok` à partir du SE Oracle Solaris, vous devez arrêter le domaine. Pour ce faire, vous pouvez utiliser la commande `halt` du SE Oracle Solaris.

Cycle d'arrêt et de redémarrage d'un serveur

Lorsque vous effectuez de la maintenance sur un système exécutant le logiciel Oracle VM Server for SPARC nécessitant un cycle d'arrêt et de redémarrage du serveur, vous devez d'abord enregistrer les configurations actuelles des domaines logiques sur le SP.

▼ Procédure d'enregistrement des configurations actuelles des domaines sur le SP

- **Utilisez la commande suivante.**

  ```
  # ldm add-config config-name
  ```

N'utilisez pas la commande `psradm`(1M) sur les CPU actives dans un domaine avec gestion de l'alimentation.

N'essayez pas de modifier l'état opérationnel d'une CPU active dans un domaine avec gestion de l'alimentation à l'aide de la commande `psradm`.

Résultat des interruptions du SE Oracle Solaris

Le comportement décrit dans cette section est visible lorsque vous effectuez les opérations suivantes :

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`.

Après ces types d'interruptions, vous recevez 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 l'arrêt ou de la réinitialisation du domaine de contrôle

Le tableau suivant indique le comportement attendu suite à l'arrêt ou à la réinitialisation du domaine (`primary`) de contrôle.

TABLEAU 12–1 Comportement attendu suite à l'arrêt ou à la réinitialisation du domaine (primary) de contrôle

Commande	Autre domaine configuré ?	Comportement
`halt`	Non configuré	L'hôte est hors tension et le reste tant qu'il n'est pas remis sous tension au niveau du SP.
	Configuré	Le logiciel redémarre et s'initialise si la variable `auto-boot?=true`. Le logiciel se réinitialise et s'arrête à l'invite ok si la variable `auto-boot?=false`.
`reboot`	Non configuré	Redémarre l'hôte sans mise hors tension.
	Configuré	Redémarre l'hôte sans mise hors tension.
`shutdown -i 5`	Non configuré	L'hôte est hors tension et le reste tant qu'il n'est pas remis sous tension au niveau du SP.
	Configuré	Le logiciel se réinitialise et redémarre.

Pour plus d'informations sur les conséquences de la réinitialisation d'un domaine de contrôle ayant le rôle de domaine racine, reportez-vous à la section "Redémarrage du domaine `primary`" à la page 91.

Utilisation de Logical Domains avec le processeur de service

Cette section décrit les informations à connaître lors de 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, comme par exemple `http://www.oracle.com/technetwork/documentation/sparc-tseries-servers-252697.html`.

Une option supplémentaire est disponible dans la commande ILOM existante.

```
-> set /HOST/bootmode config=config-name
```

L'option `config=`*config-name* vous permet de définir la configuration à la prochaine remise sous tension sur une autre configuration, comprenant 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.

▼ Procédure de réinitialisation de la configuration du domaine à la configuration par défaut ou à une configuration différente

- **Réinitialisez la configuration du domaine logique à la prochaine mise sous tension à la configuration d'expédition par défaut en exécutant cette commande :**

  ```
  -> set /HOST/bootmode config=factory-default
  ```

  Vous pouvez également sélectionner d'autres configurations ayant été 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
  ```

  Vous devez maintenant remettre le système sous tension 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=apple,lemon,orange,peach pine
```

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, le premier domaine maître à être en panne déclenche la stratégie de panne qu'il a définie sur tous ses domaines esclaves.

Remarque – Si plusieurs domaines maîtres tombent en panne en même temps, seul une des stratégies de panne définies sera appliquée sur tous les domaines esclaves affectés. Par exemple, si le domaine maître en panne a des stratégies de panne `stop` et `panic`, tous les domaines esclaves s'arrêteront ou paniqueront.

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 lorsque le domaine maître est en panne.
- `panic` panique tous les domaines esclaves lorsque le domaine maître est en panne.
- `reset` réinitialise tous les domaines esclaves lorsque le domaine maître est en panne.
- `stop` arrête tous les domaines esclaves lorsque le domaine maître est en panne.

Dans cet exemple, les domaines maîtres définissent leur stratégie de panne comme suit :

```
# ldm set-domain failure-policy=ignore apple
# ldm set-domain failure-policy=panic lemon
# ldm set-domain failure-policy=reset orange
# ldm set-domain failure-policy=stop peach
```

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 268.

Pour obtenir des exemples XML de dépendance de domaine, reportez-vous à l'Exemple 17–6.

Exemples de dépendance de domaine

Les exemples suivants montrent comment configurer les 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=panic` pour indiquer que les domaines esclaves paniquent 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 ou panique. Le premier domaine maître à tomber en panne est celui responsable de la détermination du comportement des domaines esclaves.

  ```
  # ldm add-domain failure-policy=reset twizzle
  # ldm set-domain failure-policy=panic primary
  # ldm add-domain master=twizzle,primary chocktaw
  ```

- 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.

  ```
  # ldm set-domain master=primary orange
  # ldm set-domain master=orange,primary tangerine
  # 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=stop

  DEPENDENCY
  ```

```
    master=primary

------------------------------------------------------------------------------
NAME             STATE      FLAGS   UTIL
tangerine        bound      ------

HOSTID
    0x84f948e9

CONTROL
    failure-policy=ignore

DEPENDENCY
    master=orange,primary
```

- L'exemple suivant montre la liste d'exemples avec une sortie analysable :

```
# 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 suivante :

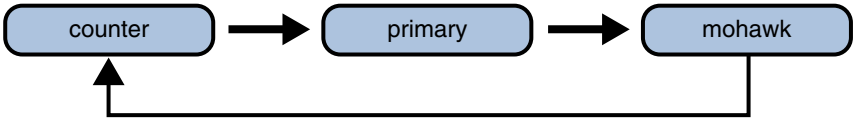
FIGURE 12–1 Dépendance d'un domaine unique

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 suivante :

FIGURE 12–2 Dépendance de plusieurs domaines

La troisième commande essaie de créer une dépendance entre les domaines `counter` et `mohawk`, ce qui produirait le cycle de dépendance suivant :

FIGURE 12–3 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 la CPU

Le numéro de domaine et de CPU virtuelle dans le domaine, qui correspond à un numéro de CPU physique donné, peut être déterminé avec les procédures suivantes.

▼ Procédure de détermination du numéro de la CPU

1. **Générez une longue liste analysable de tous les domaines.**

   ```
   primary# ldm list -l -p
   ```

2. **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

Le domaine et l'adresse de mémoire réelle dans le domaine, qui correspondent à une adresse de mémoire physique (PA) donnée, peuvent être déterminés comme suit.

▼ Procédure de détermination de l'adresse de la mémoire réelle

1. **Générez une longue liste analysable de tous les domaines.**

   ```
   primary# ldm list -l -p
   ```

2. **Recherchez la ligne dans les sections `MEMORY` de la liste où la PA se trouve dans la plage comprenant *pa* à (*pa* + *size* - 1), c'est-à-dire, *pa* ≤ PA ≤ (*pa* + *size* - 1).**

   Ici *pa* et *taille* concernent les valeurs dans les zones correspondantes 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.**

Exemples de mappage de CPU et de mémoire

Supposons que vous ayez une configuration de domaine logique identique à celle représentée dans l'Exemple 12–1, et que vous souhaitiez déterminer le domaine et la CPU virtuelle correspondant au numéro de CPU physique `5`, et 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.

EXEMPLE 12–1 Liste longue analysable des configurations de Logical Domains

```
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
...
```

Utilisation des identificateurs uniques universellement

A partir de la version Oracle VM Server for SPARC 2.0, 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, racine)
- 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).

PARTIE II

Logiciel Oracle VM Server for SPARC facultatif

Cette section présente les fonctionnalités et logiciels facultatifs que vous pouvez utiliser avec le logiciel Oracle VM Server for SPARC .2.2.

◆ ◆ ◆ CHAPITRE 13

Outil de conversion physique-à-virtuel Oracle VM Server for SPARC

Ce chapitre aborde les sujets suivants :

Présentation de l'outil P2V Oracle VM Server for SPARC

L'outil de conversion physique-à-virtuel (P2V) Oracle VM Server for SPARC convertit automatiquement un système physique existant en système virtuel exécutant le SE Oracle Solaris 10 dans un domaine logique sur un système chip multithreading (CMT). Vous pouvez exécuter la commande `ldmp2v` à partir d'un domaine de contrôle qui exécute le SE Oracle Solaris 10 ou le SE Oracle Solaris 11 pour convertir l'un des systèmes source suivants en un domaine logique :

- Un système `sun4u` SPARC qui exécute au moins le SE Solaris 8, Solaris 9 ou Oracle Solaris 10.
- Un système `sun4v` qui exécute le SE Oracle Solaris 10, mais ne s'exécute pas dans un domaine logique

---

Remarque – Vous *ne pouvez pas* utiliser l'outil P2V pour convertir un système physique Oracle Solaris 11 en un système virtuel.

La conversion d'un système physique en un système virtuel est effectué selon la procédure suivante :

- **Phase de collecte.** S'exécute sur le système source physique. Dans la phase de `collecte`, une image du système de fichier du système source est créée en fonction des informations de configuration collectées sur le système source.
- **Phase de préparation.** S'exécute sur le domaine de contrôle du système cible. Dans le phase de `préparation`, un domaine logique est créé sur le système cible en fonction des informations de configuration collectées au cours de la phase de `collecte`. L'image du système de fichiers est restaurée sur un ou plusieurs disques virtuels. Vous pouvez utiliser l'outil P2V pour créer les disques virtuels sur des fichiers bruts ou des volumes ZFS. Vous pouvez également créer des disques virtuels sur des disques physiques ou des LUN ou sur les volumes du gestionnaire de volumes que vous avez créé. L'image est modifiée afin de pouvoir s'exécuter en tant que domaine logique.
- **Phase de conversion.** S'exécute sur le domaine de contrôle du système cible. Lors de la phase `convert`, le domaine logique créé est converti en domaine logique exécutant le SE Oracle Solaris 10 à l'aide du processus de mise à niveau standard Oracle Solaris.

Pour plus d'informations sur l'outil P2V, reportez-vous à la page de manuel `ldmp2v(1M)`.

Les sections suivantes décrivent comme la conversion d'un système physique à un système virtuel est effectuée par phases.

Phase de collecte

La phase de collecte s'exécute sur le système à convertir. Pour créer un image du système de fichiers cohérente, vérifiez que le système est aussi silencieux que possible et que toutes les applications sont arrêtées. La commande `ldmp2v` créer une sauvegarde de tous les systèmes de fichiers UFS montés. Vérifiez donc que les systèmes de fichiers à déplacer vers un domaine logique sont montés. Vous pouvez exclure les systèmes de fichiers montés que vous ne souhaitez pas déplacer, notamment les systèmes de fichiers sur un stockage SAN ou les systèmes de fichiers qui seront déplacés par d'autres moyens. Utilisez l'option `-x` pour exclure de tels systèmes de fichiers. Les systèmes de fichiers exclus par l'option `-x` ne sont pas recréés sur le domaine invité. Vous pouvez utiliser l'option `-O` pour exclure des fichiers et des répertoires.

Aucune modification n'est nécessaire dans le système source. La seule contrainte est que le script `ldmp2v` qui a été installé sur le domaine de contrôle. Vérifiez que l'utilitaire `flarcreate` est présent sur le système source.

Phase de préparation

La phase de préparation utilise les données collectées au cours de la phase de collecte pour créer un domaine logique qui est comparable au système source.

Vous pouvez utiliser la commande `ldmp2v` de l'une des manières suivantes :

- **Mode automatique.** Ce mode crée automatiquement des disques virtuels et restaure les données du système de fichiers.
  - Crée le domaine logique et les disques virtuels requis de la même taille sur le système source.
  - Partitionne les disques et restaure les systèmes de fichiers.

    Si la taille combinée des systèmes de fichiers `/`, `/usr` et `/var` est inférieure à 10 Go, les tailles de ces systèmes de fichiers sont automatiquement ajustées pour s'adapter aux contraintes d'espace disque plus importantes du SE Oracle Solaris 10. Le redimensionnement automatique peut être désactivé à l'aide de l'option `-x no-auto-adjust-fs` ou à l'aide de l'option `-m` pour redimensionner manuellement un système de fichiers.
  - Modifie l'image du SE du domaine logique pour remplacer toutes les références au matériel physique avec les versions appropriées à un domaine logique. Cela vous permet de mettre à niveau le système au SE Oracle Solaris 10 à l'aide du processus de mise à niveau normal Oracle Solaris. Les modifications comprennent la mise à niveau du fichier `/etc/vfstab` pour prendre en compte les nouveaux noms de disque. Tous les disques de démarrage encapsulés Oracle Solaris Volume Manager ou Veritas Volume Manager (VxVM) sont automatiquement désencapsulés au cours de ce processus. Lorsqu'un disque est désencapsulé, il est converti en tranches de disque brutes. Si VxVM est installé sur le système source, le processus P2V désactive VxVM sur le domaine invité créé.
- **Mode non automatique.** Vous devez créer les disques virtuels et restaurer les données du système de fichiers manuellement. Ce mode vous permet de modifier la taille et le nombre de disques, le partitionnement et la disposition du système de fichiers. La phase de préparation de ce mode exécute uniquement la création du domaine logique et les étapes de modification de l'image du SE sur le système de fichiers.
- **Mode de nettoyage.** Supprime un domaine logique ainsi que tous les périphériques backend sous-jacents créés par `ldmp2v`.

Phase de conversion

Lors de la phase de conversion, le domaine logique utilise le processus de mise à niveau Oracle Solaris pour mettre à niveau le SE Oracle Solaris 10. L'opération de mise à niveau supprime tous les packages existants et installe les packages `sun4v` Oracle Solaris 10, qui exécutent automatiquement une conversion `sun4u` à `sun4v`. La phase `convert` peut utiliser une image iso du DVD Oracle Solaris ou une image d'installation réseau. Sur les systèmes Oracle Solaris 10, vous pouvez également utiliser la fonction JumpStart d'Oracle Solaris pour effectuer une opération de mise à niveau totalement automatisée.

Périphériques backend

Vous pouvez créer des disques virtuels pour un domaine invité sur un certain nombre de types de backends : fichiers (`file`), volumes ZFS (`zvol`), disques physiques ou LUN (`disk`) ou volumes de gestionnaire de volumes (`disk`). La commande `ldmp2v` crée automatiquement des fichiers ou des volumes ZFS de la taille appropriée si vous indiquez `file` ou `zvol` comme type de backend avec l'une des méthodes suivantes :

- A l'aide de l'option `-b`
- En indiquant la valeur du paramètre `BACKEND_TYPE` dans le fichier `/etc/ldmp2v.conf`

Le type de backend `disk` vous permet d'utiliser un disque physique, un LUN ou un volume de gestionnaire de volumes (Oracle Solaris Volume Manager et Veritas Volume Manager (VxVM)) en tant que périphérique backend pour les disques virtuels. Vous devez créer le disque ou le volume avec une taille appropriée avant de commencer la phase de `préparation`. Pour un disque physique ou un LUN, définissez le périphérique backend en tant que tranche 2 du bloc ou périphérique de caractère du disque, notamment `/dev/dsk/c0t3d0s2`. Pour un volume de gestionnaire de volumes, définissez le bloc ou le périphérique de caractère pour le volume, tel que `/dev/md/dsk/d100` pourOracle Solaris Volume Manager ou `/dev/vx/dsk/ldomdg/vol1` pour VxVM.

A moins de définir les noms de volume et de disque virtuel avec l'option `-B` *backend*:*volume*:*vdisk*, les volumes et les disques virtuels que vous créez pour l'invité sont les noms donnés par défaut.

- *backend* définit le nom du moteur de traitement à utiliser. Vous devez définir *backend* pour le type de backend `disk`. *backend* est facultatif pour les types de backends `file` et `zvol`, et peut être utilisé pour définir un nom non par défaut pour le fichier ou le volume ZFS que `ldmp2v` crée. Le nom par défaut est `$BACKEND_PREFIX`/*guest-name*/`disk`*N*.
- *volume* est facultatif pour tous les types de backends et définit le nom du volume du serveur de disques virtuels pour créer le domaine invité. Si cette option n'est pas indiquée, *volume* est *guest-name*`-vol`*N*.
- *vdisk* est facultatif pour tous les types de backends et définit le nom du volume dans le domaine invité. Si cette option n'est pas indiquée, *vdisk* est `disk`*N*.

Remarque – Au cours du processus de conversion, le disque virtuel est temporairement nommé *guest-name*`-disk`*N* pour garantir que le nom dans le domaine de contrôle est unique.

Pour indiquer une valeur vide pour *backend*, *volume* ou*vdisk*, incluez uniquement le séparateur deux-points. Par exemple, `-B ::vdisk001` définit le nom du disque virtuel sur `vdisk001` et utilise les noms par défaut pour le moteur de traitement et le volume. Si vous ne définissez pas *vdisk*, vous pouvez omettre le séparateur deux-points de fin. Par exemple, `-B`

`/ldoms/ldom1/vol001:vol001` définit le nom du fichier backend comme `/ldoms/ldom1/vol001` et le nom du volume comme `vol001`. Le nom du disque virtuel par défaut est `disk0`.

Installation de l'outil P2V Oracle VM Server for SPARC

Le package de l'outil P2V Oracle VM Server for SPARC doit être installé et configuré *uniquement* sur le domaine de contrôle du système cible. Il est inutile d'installer le package sur le système source. Au lieu de cela, il vous suffit de copier le script `/usr/sbin/ldmp2v` du système cible sur le système source.

Remarque – Sur un système Oracle Solaris 10, `ldmp2v` est installé à partir du package `SUNWldmp2v`, tandis que sur un système Oracle Solaris 11, `ldmp2v` est installé par défaut, et ce à partir du package `ldomsmanager`.

Conditions requises

Les conditions suivantes doivent être remplies pour vous permettre d'exécuter l'outil P2V Oracle VM Server for SPARC :

- Les patchs de l'archive Flash suivants sont installés sur le système source :
  - **Pour le SE Solaris 8 :** au moins le patch ID 109318-34
  - **Pour le SE Solaris 9 :** au moins le patch ID 113434-06
- Le système cible exécute au moins Logical Domains 1.1 sur :
  - SE Solaris 10 10/08
  - SE Solaris 10 5/08 avec les patchs Logical Domains 1.1 appropriés
- Domaines invités exécutant au moins le SE Solaris 10 5/08
- Système source exécutant au moins le SE Solaris 8

En plus de ces conditions requises, configurez un système de fichiers NFS à partager par les systèmes source et cible. Le système de fichiers doit être accessible en écriture par `root`. Cependant, si un système de fichiers partagé n'est pas disponible, utilisez un système de fichiers local de taille suffisante pour contenir un vidage du système de fichiers du système source sur les système source et cible.

Restrictions

L'outil Oracle VM Server for SPARC P2V présente les restrictions suivantes :

- Seuls les systèmes de fichiers UFS sont pris en charge.
- Seuls les disques bruts (`//dev/dsk/c0t0d0s0`), les métapériphériques Oracle Solaris Volume Manager (`/dev/md/dsk/d`*NNN*) et les disques de démarrage encapsulés VxVM sont pris en charge sur le système source.
- Au cours du processus P2V, chaque domaine invité ne peut avoir qu'un seul commutateur virtuel et un seul serveur de disque virtuel. Vous pouvez ajouter plusieurs commutateurs virtuels et serveurs de disque virtuel au domaine après la conversion P2V.
- La prise en charge VxVM est limitée aux volumes suivants sur le disque d'initialisation encapsulé : `rootvol`, `swapvol`, `usr`, `var`, `opt` et `home`. Les tranches d'origine de ces volumes doivent toujours être présentes sur le disque d'initialisation. L'outil P2V prend en charge Veritas Volume Manager 5.x sur le SE Oracle Solaris 10. Cependant, vous pouvez également utiliser l'outil P2V pour convertir les systèmes d'exploitation Solaris 8 et Solaris 9 utilisant VxVM.
- Les systèmes Oracle Solaris 10 comportant des zones peuvent être convertis si les zones sont détachées à l'aide de la commande `zoneadm detach` avant l'exécution de l'opération `ldmp2v collect`. A l'issue de la conversion P2V, exécutez la commande `zoneadm attach` pour associer à nouveau les zones créées sur le domaine invité. Pour plus d'informations sur l'exécution de ces étapes sur un domaine invité, reportez-vous à la section "Migración de una zona no global a un equipo distinto" du manuel *Administración de Oracle Solaris: zonas de Oracle Solaris, zonas de Oracle Solaris 10 y gestión de recursos*.

Remarque – L'outil P2V ne met *pas* à jour les paramètres de configuration des zones telles que le chemin ou l'interface réseau des zones, pas plus qu'il ne déplace ou configure le stockage pour le chemin d'accès des zones. Vous devez mettre à jour manuellement la configuration des zones et déplacer le chemin d'accès des zones sur le domaine invité. Reportez-vous à la section "Migración de una zona no global a un equipo distinto" du manuel *Administración de Oracle Solaris: zonas de Oracle Solaris, zonas de Oracle Solaris 10 y gestión de recursos*.

▼ Procédure d'installation de l'outil P2V Oracle VM Server for SPARC

Cette procédure décrit l'installation de la commande `ldmp2v` à l'aide du package `SUNWldmp2v` sur un système Oracle Solaris 10.

Si vous souhaitez installer la commande `ldmp2v` sur un système Oracle Solaris 11, la commande est installée par défaut lorsque vous installez le package `ldomsmanager`.

1. **Accédez à la page de téléchargement d'Oracle VM Server for SPARC à l'adresse suivante : `http://www.oracle.com/virtualization/index.html`.**

2. **Téléchargez le package logiciel P2V, `SUNWldmp2v`.**

   Le package `SUNWldmp2v` est inclus dans le fichier zip Oracle VM Server for SPARC.

3. **Prenez le rôle d'administrateur, de superutilisateur ou un rôle équivalent.**

   Pour Oracle Solaris 10, reportez-vous à la section "Configuración de RBAC (mapa de tareas)" du manuel *Guía de administración del sistema: servicios de seguridad*. Pour Oracle Solaris 11, reportez-vous à la Partie III, "Roles, perfiles de derechos y privilegios" du manuel *Administración de Oracle Solaris: servicios de seguridad*.

4. **Utilisez la commande `pkgadd` pour installer le package `SUNWldmp2v`.**

   ```
   # pkgadd -d . SUNWldmp2v
   ```

5. **Créez le fichier `/etc/ldmp2v.conf` et configurez les propriétés par défaut suivantes :**

   - `VDS` – Nom du service de disque virtuel, par exemple `VDS="primary-vds0"`
   - `VSW` – Nom du commutateur virtuel, par exemple `VSW="primary-vsw0"`
   - `VCC` – Nom du concentrateur de console virtuelle, par exemple `VCC="primary-vcc0"`
   - `BACKEND_TYPE` – Type de backend de `zvol`, `file` ou `disk`
   - `BACKEND_SPARSE` – Créer des périphériques backend en tant que volumes ou fichiers d'analyse syntaxique `BACKEND_SPARSE="yes"` ou en tant que fichiers ou volumes sans analyse syntaxique `BACKEND_SPARSE="no"`
   - `BACKEND_PREFIX` – Emplacement pour créer les périphériques backend du disque virtuel

     Si `BACKEND_TYPE="zvol"`, définissez la valeur `BACKEND_PREFIX` en tant que nom d'ensemble de données ZFS. Si `BACKEND_TYPE="files"`, la valeur `BACKEND_PREFIX` est interprétée comme un nom de chemin d'un répertoire relatif à `/`.

     Par exemple, `BACKEND_PREFIX="tank/ldoms"` créerait des ZVOL dans l'ensemble de données `tank/ldoms/`*domain-name* et des fichiers dans le sous-répertoire `/tank/ldoms/`*domain-name*.

     La propriété `BACKEND_PREFIX` n'est pas applicable au backend `disk`.
   - `BOOT_TIMEOUT` – Délai d'attente du démarrage du SE Oracle Solaris en secondes

   Pour plus d'informations, consultez le fichier de configuration `ldmp2v.conf.sample` qui fait partie du bundle téléchargeable.

Utilisation de la commande ldmp2v

Cette section inclut des exemples pour ces trois phrases.

EXEMPLE 13–1 Exemples de phase de collecte

Les exemples suivants montrent comment vous pouvez utiliser la commande `ldmp2v collect`.

- **Partage d'un système de fichiers monté NFS.** L'exemple suivant illustre la méthode la plus simple pour effectuer l'étape `collect` lorsque les systèmes source et cible partagent un système de fichiers monté NFS.

  En tant que superutilisateur, vérifiez que tous les systèmes de fichiers UFS requis sont montés.

  ```
  volumia# df -k
  Filesystem            kbytes    used   avail capacity  Mounted on
  /dev/dsk/c1t1d0s0    16516485  463289 15888032     3%    /
  /proc                      0       0       0     0%    /proc
  fd                         0       0       0     0%    /dev/fd
  mnttab                     0       0       0     0%    /etc/mnttab
  /dev/dsk/c1t1d0s3    8258597    4304 8171708     1%    /var
  swap                 4487448      16 4487432     1%    /var/run
  swap                 4487448      16 4487432     1%    /tmp
  /dev/dsk/c1t0d0s0    1016122       9  955146     1%    /u01
  vandikhout:/u1/home/dana
                       6230996752 1051158977 5179837775    17%    /home/dana
  ```

  L'exemple suivant montre comme exécuter l'outil de collecte lorsque les systèmes source et cible partagent un système de fichiers monté NFS :

  ```
  volumia# ldmp2v collect -d home/dana/volumia
  Collecting system configuration ...
  Archiving file systems ...
  Determining which filesystems will be included in the archive...
  Creating the archive...
  895080 blocks
  Archive creation complete.
  ```

- **Sans partage d'un système de fichiers monté NFS.** Lorsque les systèmes source et cible ne partagent pas un système de fichiers monté NFS, l'image du système de fichiers peut être écrite sur le stockage local, puis copiée ultérieurement sur le domaine de contrôle. L'utilitaire Flash exclut automatiquement l'archive qu'il crée.

  ```
  volumia# ldmp2v collect -d /var/tmp/volumia
  Collecting system configuration ...
  Archiving file systems ...
  Determining which filesystems will be included in the archive...
  Creating the archive...
  895080 blocks
  Archive creation complete.
  ```

  Copiez l'archive Flash et le fichier `manifest` du répertoire`/var/tmp/volumia` sur le système cible.

EXEMPLE 13–1 Exemples de phase de collecte *(Suite)*

Astuce – Dans certains cas, `ldmp2v` peut afficher les erreurs de la commande `cpio`. En général, ces erreurs déclenchent des messages de type `File size of etc/mnttab has increased by 435`. Vous pouvez ignorer les messages relatifs aux fichiers journaux ou aux fichiers qui rapportent l'état du système. Assurez-vous de vérifier avec soin tous les messages d'erreur.

- **Omission de l'étape de sauvegarde du système de fichiers.** Si des sauvegardes du système sont déjà disponibles grâce à un outil de sauvegarde tiers tels que NetBackup, vous pouvez passer l'étape de sauvegarde du système de fichiers à l'aide de la méthode d'archivage `none`. Lorsque vous utilisez cette option, seul le manifeste de la configuration du système est créé.

  ```
  volumia# ldmp2v collect -d /home/dana/p2v/volumia -a none
  Collecting system configuration ...
  The following file system(s) must be archived manually: / /u01 /var
  ```

  Notez que, si la répertoire indiqué par `-d` n'est pas partagé par les systèmes source et cible, vous devez copier le contenu de ce répertoire sur le domaine de contrôle. Le contenu du répertoire doit être copié sur le domaine de contrôle avant la phase de préparation.

EXEMPLE 13–2 Exemples de phase de préparation

Les exemples suivants montrent comment vous pouvez utiliser la commande `ldmp2v prepare`.

- L'exemple suivant crée un domaine logique appelé `volumia` à l'aide des valeurs par défaut configurées dans `/etc/ldmp2v.conf` tout en conservant les adresses MAC du système physique :

  ```
  # ldmp2v prepare -d /home/dana/p2v/volumia -o keep-mac volumia
  Creating vdisks ...
  Creating file systems ...
  Populating file systems ...
  Modifying guest domain OS image ...
  Removing SVM configuration ...
  Unmounting guest file systems ...
  Creating domain volumia ...
  Attaching vdisks to domain volumia ...
  ```

- La commande suivante montre des informations sur le domaine logique `volumia` :

  ```
  # ldm list -l volumia
  NAME             STATE      FLAGS   CONS    VCPU  MEMORY   UTIL  UPTIME
  volumia          inactive   ------          2     4G

  NETWORK
      NAME    SERVICE                 DEVICE     MAC               MODE   PVID VID
      vnet0   primary-vsw0                       00:03:ba:1d:7a:5a        1

  DISK
      NAME    DEVICE  TOUT MPGROUP        VOLUME                      SERVER
      disk0                                       volumia-vol0@primary-vds0
      disk1                                       volumia-vol1@primary-vds0
  ```

EXEMPLE 13–2 Exemples de phase de préparation *(Suite)*

- L'exemple suivant montre que vous pouvez totalement supprimer un domaine et ses périphériques backend à l'aide de l'option `-C` :

```
# ldmp2v prepare -C volumia
Cleaning up domain volumia ...
Removing vdisk disk0 ...
Removing vdisk disk1 ...
Removing domain volumia ...
Removing volume volumia-vol0@primary-vds0 ...
Removing ZFS volume tank/ldoms/volumia/disk0 ...
Removing volume volumia-vol1@primary-vds0 ...
Removing ZFS volume tank/ldoms/volumia/disk1 ...
```

- L'exemple suivant montre que vous pouvez redimensionner un ou plusieurs systèmes de fichiers au cours de P2V en indiquant le point de montage et la nouvelle taille avec l'option `-m`.

```
# ldmp2v prepare -d /home/dana/p2v/volumia -m /:8g volumia
Resizing file systems ...
Creating vdisks ...
Creating file systems ...
Populating file systems ...
Modifying guest domain OS image ...
Removing SVM configuration ...
Modifying file systems on SVM devices ...
Unmounting guest file systems ...
Creating domain volumia ...
Attaching vdisks to domain volumia ...
```

EXEMPLE 13–3 Exemples de phase de conversion

Les exemples suivants montrent comment vous pouvez utiliser la commande `ldmp2v convert`.

- **Utilisation d'un serveur d'installation réseau.** La commande `ldmp2v convert` initialise le domaine via le réseau à l'aide de l'interface de réseau virtuel spécifiée. Vous devez exécuter les scripts `setup_install_server` et `add_install_client` sur le serveur d'installation.

  Sur les systèmes Oracle Solaris 10, vous pouvez utiliser la fonction JumpStart d'Oracle Solaris pour effectuer une conversion totalement automatisée. Cette fonction nécessite que vous créiez et configuriez les fichiers `sysidcfg` et de profil appropriés pour le client du serveur JumpStart. Le profil doit être composé des lignes suivantes :

```
install_type    upgrade
root_device     c0d0s0
```

  Le fichier `sysidcfg` est uniquement utilisé pour l'opération de mise à niveau. Par conséquent, une configuration telle que la suivante devrait être suffisante :

```
name_service=NONE
root_password=uQkoXlMLCsZhI
system_locale=C
timeserver=localhost
timezone=Europe/Amsterdam
terminal=vt100
```

EXEMPLE 13–3 Exemples de phase de conversion *(Suite)*

```
security_policy=NONE
nfs4_domain=dynamic
auto_reg=disable
network_interface=PRIMARY {netmask=255.255.255.192
        default_route=none protocol_ipv6=no}
```

Pour plus d'informations sur l'utilisation de JumpStart, reportez-vous au manuel *Guía de instalación de Oracle Solaris 10 8/11: instalaciones JumpStart personalizadas y avanzadas*.

Remarque – Le fichier `sysidcfg` d'exemple inclut le mot-clé `auto_reg`, qui a été introduit dans la version Oracle Solaris 10 9/10. Ce mot-clé est *uniquement* requis si vous exécutez au moins la version Oracle Solaris 10 9/10.

```
# ldmp2v convert -j -n vnet0 -d /p2v/volumia volumia
LDom volumia started
Waiting for Solaris to come up ...
Using Custom JumpStart
Trying 0.0.0.0...
Connected to 0.
Escape character is '^]'.

Connecting to console "volumia" in group "volumia" ....
Press ~? for control options ..
SunOS Release 5.10 Version Generic_137137-09 64-bit
Copyright (c) 1983-2010, Oracle and/or its affiliates.  All rights reserved.
Configuring devices.
Using RPC Bootparams for network configuration information.
Attempting to configure interface vnet0...
Configured interface vnet0
Reading ZFS config: done.
Setting up Java. Please wait...
Serial console, reverting to text install
Beginning system identification...
Searching for configuration file(s)...
Using sysid configuration file
  129.159.206.54:/opt/SUNWjet/Clients/volumia/sysidcfg
Search complete.
Discovering additional network configuration...
Completing system identification...
Starting remote procedure call (RPC) services: done.
System identification complete.
Starting Solaris installation program...
Searching for JumpStart directory...
Using rules.ok from 129.159.206.54:/opt/SUNWjet.
Checking rules.ok file...
Using begin script: Clients/volumia/begin
Using profile: Clients/volumia/profile
Using finish script: Clients/volumia/finish
Executing JumpStart preinstall phase...
Executing begin script "Clients/volumia/begin"...
Begin script Clients/volumia/begin execution completed.
Searching for SolStart directory...
Checking rules.ok file...
```

EXEMPLE 13–3 Exemples de phase de conversion *(Suite)*

```
Using begin script: install_begin
Using finish script: patch_finish
Executing SolStart preinstall phase...
Executing begin script "install_begin"...
Begin script install_begin execution completed.
WARNING: Backup media not specified.  A backup media (backup_media)
  keyword must be specified if an upgrade with disk space reallocation
  is required

Processing profile

Loading local environment and services

Generating upgrade actions
Checking file system space: 100% completed
Space check complete.

Building upgrade script

Preparing system for Solaris upgrade

Upgrading Solaris: 10% completed
[...]
```

- **Utilisation d'une image ISO.** La commande `ldmp2v convert` associe l'image ISO du DVD Oracle Solaris au domaine logique et effectue une initialisation à partir de celui-ci. Pour effectuer la mise à niveau, répondez à toutes les invites `sysid` et sélectionnez Upgrade (Mise à niveau).

Attention – Un contrôle de sécurité est effectué avant la conversion du domaine invité. Il garantit qu'aucune adresse IP du système original n'est active afin d'éviter la présence d'adresses IP actives dupliquées sur le réseau. Vous pouvez utiliser l'option `-x skip-ping-test` pour ignorer ce contrôle de sécurité. Cela accélère le processus de conversion. Utilisez cette option *uniquement* si vous êtes certain qu'aucune adresse IP n'existe, par exemple si l'hôte original est inactif.

Remarque – Les réponses aux questions `sysid` questions sont *uniquement* utilisées pour la durée du processus de mise à niveau. Ces données ne sont pas appliquées à l'image du SE existant sur le disque. La méthode la plus simple et la plus rapide pour exécuter la conversion consiste à sélectionner Non-networked (Non mis en réseau). Le mot de passe `root` que vous indiquez ne doit pas nécessairement correspondre au mot de passe `root` du système source. L'identité d'origine du système est préservée par la mise à niveau et prend effet après le redémarrage post mise à niveau. Le temps requis pour effectuer la mise à niveau dépend du cluster Oracle Solaris installé sur le système d'origine.

```
# ldmp2v convert -i /tank/iso/s10s_u5.iso -d /home/dana/p2v/volumia volumia
Testing original system status ...
LDom volumia started
```

EXEMPLE 13–3 Exemples de phase de conversion *(Suite)*

```
Waiting for Solaris to come up ...

        Select 'Upgrade' (F2) when prompted for the installation type.
        Disconnect from the console after the Upgrade has finished.

Trying 0.0.0.0...
Connected to 0.
Escape character is '^]'.

Connecting to console "volumia" in group "volumia" ....
Press ~? for control options ..
Configuring devices.
Using RPC Bootparams for network configuration information.
Attempting to configure interface vnet0...
Extracting windowing system. Please wait...
Beginning system identification...
Searching for configuration file(s)...
Search complete.
Discovering additional network configuration...
Configured interface vnet0
Setting up Java. Please wait...

Select a Language

   0. English
   1. French
   2. German
   3. Italian
   4. Japanese
   5. Korean
   6. Simplified Chinese
   7. Spanish
   8. Swedish
   9. Traditional Chinese

Please make a choice (0 - 9), or press h or ? for help:
[...]
- Solaris Interactive Installation ----------------------------------------------

  This system is upgradable, so there are two ways to install the Solaris
  software.

  The Upgrade option updates the Solaris software to the new release, saving
  as many modifications to the previous version of Solaris software as
  possible.  Back up the system before using the Upgrade option.

  The Initial option overwrites the system disks with the new version of
  Solaris software.  This option allows you to preserve any existing file
  systems.  Back up any modifications made to the previous version of Solaris
  software before starting the Initial option.

  After you select an option and complete the tasks that follow, a summary of
  your actions will be displayed.

--------------------------------------------------------------------------------
     F2_Upgrade    F3_Go Back    F4_Initial    F5_Exit    F6_Help
```

◆ ◆ ◆ CHAPITRE 14

Assistant de configuration d'Oracle VM Server for SPARC (Oracle Solaris 10)

L'assistant de configuration d'Oracle VM Server for SPARC (la commande `ldmconfig`) vous guide tout au long de la configuration d'un domaine logique en paramétrant les propriétés de base. Il s'exécute sur des systèmes basés sur chip multithreading (CMT).

Après avoir rassemblé les données de configuration, l'assistant de configuration crée une configuration adaptée à l'initialisation en tant que domaine logique. Vous pouvez également utiliser les valeurs par défaut sélectionnées par l'assistant de configuration afin de créer une configuration système utilisable.

Remarque – La commande `ldmconfig` est uniquement prise en charge sur les systèmes Oracle Solaris 10.

Pour des informations supplémentaires, reportez-vous à la page de manuel `ldmconfig(1M)`.

Utilisation de l'assistant de configuration (`ldmconfig`)

La commande `ldmconfig` s'exécute au moyen d'une série d'opérations correspondant à des écrans de l'interface graphique. Le résultat final est la création d'une configuration que vous pouvez déployer sur un domaine logique.

Les sections suivantes décrivent la procédure d'installation de la commande `ldmconfig` ainsi que certaines fonctions de l'outil Assistant de configuration.

Installation de l'assistant de configuration

L'assistant de configuration est livré avec le package `SUNWldm`.

Après avoir installé le package SUNWldm, vous pouvez rechercher la commande ldmconfig dans le répertoire /usr/bin. La commande est également installée dans le répertoire /opt/SUNWldm/bin à des fins de récupération.

Conditions requises

Avant d'installer et d'exécuter l'assistant de configuration, vérifiez que les conditions suivantes sont remplies :

- Le système cible doit exécuter au moins le logiciel Logical Domains 1.2.
- Votre fenêtre de terminal doit présenter une largeur d'au moins 80 caractères sur une hauteur de 24 lignes.

Limitations et problèmes recensés

L'assistant de configuration présente les limitations suivantes :

- Le redimensionnement du terminal pendant l'utilisation de ldmconfig peut provoquer une sortie confuse
- Prise en charge des fichiers de disque UFS en tant que disques virtuels uniquement
- Fonctionne uniquement avec des systèmes où aucune configuration de domaines logiques existante n'est présente
- Les ports du concentrateur de la console virtuelle vont de 5000 à 5100
- Les noms par défaut utilisés pour les domaines, les services et les périphériques invités ne peuvent pas être modifiés

Fonctions de ldmconfig

La commande ldmconfig s'exécute au moyen d'une série d'opérations correspondant à des écrans de l'interface graphique. Vous pouvez naviguer en arrière (précédent) et en avant (suivant) dans ces écrans jusqu'à ce que vous atteigniez l'étape finale. L'étape finale produit la configuration. Vous pouvez quitter à tout moment l'assistant de configuration ou réinitialiser la configuration sur les valeurs par défaut. A partir de l'écran final, vous pouvez déployer la configuration sur un domaine logique.

L'assistant de configuration inspecte tout d'abord le système pour déterminer les valeurs de propriété par défaut les plus adaptées en fonction des pratiques recommandées, puis affiche les propriétés requises pour contrôler le déploiement. Notez qu'il ne s'agit pas d'une liste exhaustive. Vous pouvez définir d'autres propriétés pour personnaliser encore la configuration.

Pour plus d'informations sur l'utilisation de l'outil ldmconfig, reportez-vous à la page de manuel ldmconfig(1M).

Vous pouvez ajuster les propriétés suivantes :

- **Nombre de domaines invités.** Indiquez le nombre de domaines invités pour l'application à créer. Le minimum est un domaine invité. La valeur maximale est déterminée par la disponibilité des ressources de la VCPU. Par exemple, vous pouvez créer jusqu'à 60 domaines invités avec un seul thread, chacun sur un système CMT 64 threads et quatre threads étant réservés pour le domaine de contrôle. Si les pratiques recommandées sont sélectionnées, le nombre minimum de ressources VCPU par domaine invité est un serveur de base unique. Par conséquent, sur un système 8 serveurs de base, 8 threads par serveur de base pour lequel les pratiques recommandées sont sélectionnées, vous pouvez créer jusqu'à sept domaines invités avec chacun un serveur de base. Par conséquent, un seul serveur de base est assigné au domaine de contrôle.

  L'assistant de configuration présente le nombre maximum de domaines pouvant être configurés pour ce système.

  L'assistant de configuration effectue les tâches suivantes pour créer les domaines :

  - **Pour tous les domaines.**
    - Crée un service de terminal virtuel sur les ports de 5000 à 5100
    - Crée un service de disque virtuel
    - Crée un commutateur de réseau virtuel sur chaque adaptateur réseau indiqué
    - Active le démon du serveur de terminal virtuel
  - **Pour chaque domaine.**
    - Crée le domaine logique
    - Configure les VCPU assignées au domaine
    - Configure la mémoire assignée au domaine
    - Crée un fichier de disque UFS à utiliser en tant que disque virtuel
    - Crée un périphérique de serveur de disque virtuel `vdsdev` pour le fichier de disque
    - Assigne le fichier de disque comme disque virtuel `vdisk0` pour le domaine
    - Ajoute un adaptateur réseau virtuel au commutateur virtuel sur l'adaptateur réseau indiqué
    - Définit la propriété OBP `auto-boot?=true`
    - Définit la propriété OBP `boot-device=vdisk0`
    - Associe le domaine
    - Démarre le domaine
- **Réseau par défaut.** Spécifiez l'adaptateur réseau que les nouveaux domaines utiliseront pour la mise en réseau virtuelle. L'adaptateur doit être présent dans le système. L'assistant de configuration met en évidence les adaptateurs qui sont utilisés par le système en tant qu'adaptateurs par défaut ainsi que ceux ayant un état de liaison active (adaptateurs de câble).

- **Taille du disque virtuel.** Créez des disques virtuels pour chacun des nouveaux domaines. Ces disques virtuels sont créés en fonction des fichiers de disque situés dans les systèmes de fichiers locaux. Cette propriété contrôle la taille de chaque disque virtuel en Go. La taille minimale, 8 Go, est basée sur la taille approximative requise pour contenir un SE Oracle Solaris 10, et la taille maximale est 100 Go.

  Si l'assistant de configuration ne peut pas trouver les systèmes de fichiers contenant un espace adapté pour contenir les fichiers du disque pour tous les domaines, un écran d'erreur s'affiche. Dans ce cas, vous devrez suivre la procédure suivante avant de réexécuter l'application :

  - Réduisez la taille des disques virtuels
  - Réduisez le nombre de domaines
  - Ajoutez des systèmes de fichiers avec une capacité plus importante
- **Répertoire du disque virtuel.** Spécifiez un système de fichiers disposant d'une capacité suffisante sur lequel stocker les fichiers à créer en tant que disques virtuels pour les nouveaux domaines. Le répertoire est basé sur le nombre de domaines sélectionnés et la taille des disques virtuels. La valeur doit être recalculée et les répertoires de destination sélectionnés à chaque fois que ces valeurs de propriété sont modifiées. L'assistant de configuration vous présente une liste des systèmes de fichiers ayant suffisamment d'espace. Après avoir indiqué le nom du système de fichiers, l'assistant de configuration crée un répertoire dans ce système de fichiers appelé `/ldoms/disks` dans lequel créer les images de disque.
- **Pratique recommandée.** Indiquez s'il faut utiliser les pratiques recommandées pour les valeurs de propriétés.
  - Lorsque la valeur est `yes`, l'assistant de configuration utilise la pratique recommandée pour plusieurs valeurs de propriété de configuration. Il force le minimum d'un serveur de base par domaine, y compris pour les domaines du système. En conséquence, cette opération limite le nombre maximum de domaines invités au nombre total de serveurs de base présents dans le système moins un serveur de base pour les domaines du système. Par exemple, dans le cas d'un T5140 SPARC Enterprise deux sockets avec huit serveurs de base chacun, le nombre maximum de domaines invités est 15 plus le domaine du système.
  - Si la valeur est `no`, l'assistant de configuration autorise la création de domaines ayant un thread au minimum, mais conserve au moins quatre threads pour le domaine du système.

Ensuite, l'assistant de configuration récapitule la configuration de déploiement à créer, qui comprend les informations suivantes :

- Nombre de domaines
- CPU assignée à chaque domaine invité
- Mémoire assignée à chaque domaine invité
- Taille et emplacement des disques virtuels
- Adaptateur réseau à utiliser pour les services de réseau virtuel pour les domaines invités

- Quantité de CPU et de mémoire à utiliser par le système pour les services
- Si un DVD du SE Oracle Solaris valide a été identifié, il sera utilisé pour créer un périphérique CD-ROM virtuel partagé pour permettre aux domaines invités d'installer le SE Oracle Solaris.

Enfin, l'assistant de configuration configure le système pour créer le déploiement de Logical Domains spécifié. Il décrit également les actions à effectuer et montre les commandes à exécuter pour configurer le système. Ces informations peuvent vous aider à apprendre comment utiliser les commandes `ldm` nécessaires pour configurer le système.

Attention – N'interagissez *pas* avec cette étape de configuration et n'interrompez *pas* ce processus, car le système risquerait d'être partiellement configuré.

Une fois les commandes exécutées avec succès, redémarrez le système pour appliquer les modifications.

◆ ◆ ◆ CHAPITRE 15

Utilisation du logiciel 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 contrôler à distance des domaines, et de démarrer et d'arrêter des domaines logiques (domaines) via le protocole SNMP (Simple Network Management Protocol).

Vous pouvez exécuter une seule instance du logiciel Oracle VM Server for SPARC MIB sur le domaine de contrôle. Le domaine de contrôle doit exécuter le SE Solaris 10 11/06 et le logiciel Oracle VM Server for SPARC 2.2 minimum.

Remarque – Le logiciel Oracle VM Server for SPARC MIB peut uniquement être utilisé sur un système Oracle Solaris 10.

Ce chapitre aborde les sujets suivants :

Pour utiliser le logiciel Oracle VM Server for SPARC MIB correctement, vous devez comprendre la procédure d'utilisation des produits logiciels et fonctionnalités suivants :

- SE Oracle Solaris
- Logiciel Oracle VM Server for SPARC
- SNMP (Simple Network Management Protocol)
- MIB (Management Information Base) SNMP
- Agent de gestion du système (SMA, System Management Agent)

- Protocoles SNMP version 1 (SNMPv1), SNMP version 2 (SNMPv2c) et SNMP version 3 (SNMPv3)
- Structure of Management Information (SMI) version 1 et version 2
- Structure Management Information Base (MIB)
- Abstract Syntax Notation (ASN.1)

Présentation du logiciel MIB Oracle VM Server for SPARC

Cette section couvre les sujets suivants :

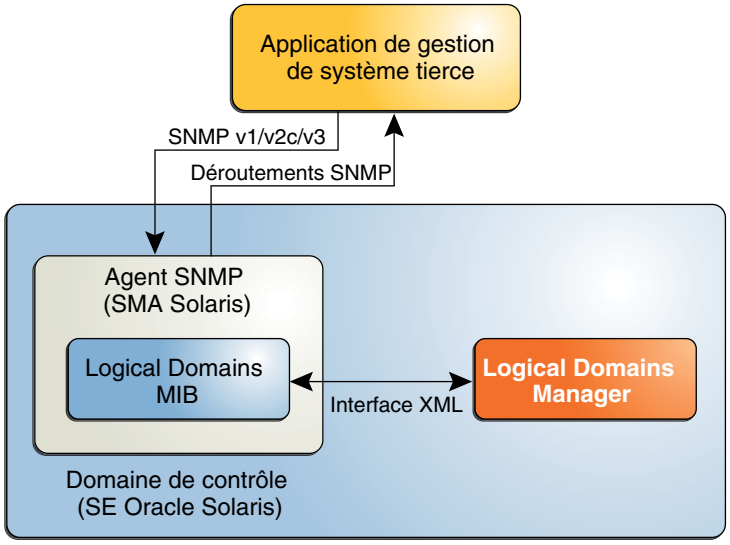
Composants logiciels

Le package Oracle VM Server for SPARC MIB, `SUNWldmib.v`, contient les composants logiciels suivants :

- `SUN-LDOM-MIB.mib` est une MIB SNMP sous forme de fichier texte. Ce fichier définit les objets du logiciel Oracle VM Server for SPARC MIB.
- `ldomMIB.so` est un module d'extension SMA sous forme de bibliothèque partagée. Ce module permet au SMA de répondre aux requêtes d'informations spécifiées dans le logiciel Oracle VM Server for SPARC MIB et de générer des déroutements.

La figure suivante présente l'interaction entre le logiciel Oracle VM Server for SPARC MIB, le SMA, Logical Domains Manager et une application de gestion de système tierce. L'interaction illustrée dans cette figure est décrite dans les sections "Agent de gestion du système" à la page 299 et "Logical Domains Manager et Oracle VM Server for SPARC MIB" à la page 300.

FIGURE 15–1 Interaction du logiciel Oracle VM Server for SPARC MIB avec le SMA, Logical Domains Manager et les applications de gestion de système tierces



Agent de gestion du système

L'agent (SMA) SNMP Solaris exécute les fonctions suivantes :

- Ecoute des requêtes issues d'une application de gestion de système tierce afin d'obtenir ou de définir des données proposées par le logiciel Oracle VM Server for SPARC MIB. L'agent écoute sur le port SNMP standard, `161`.
- Emission de déroutements vers l'application de gestion de système configurée à l'aide du port standard pour les notifications SNMP, `162`.

Le logiciel Oracle VM Server for SPARC MIB est exporté par le SMA par défaut du SE Oracle Solaris sur le domaine de contrôle.

Le SMA prend en charge les fonctions get, set et trap des versions v1, v2 et v3 de SNMP. La plupart des objets Oracle VM Server for SPARC MIB sont en lecture seule à des fins de contrôle. Cependant, pour démarrer ou arrêter un domaine, vous devez écrire une valeur dans la propriété `ldomAdminState` de la table `ldomTable`. Reportez-vous au Tableau 15–1.

Logical Domains Manager et Oracle VM Server for SPARC MIB

Un *domaine* est un conteneur formé d'un ensemble de ressources virtuelles pour un système d'exploitation invité. Logical Domains Manager fournit l'interface de ligne de commande nécessaire à la création, la configuration et la gestion des domaines. Logical Domains Manager et le logiciel Oracle VM Server for SPARC MIB prennent en charge les ressources virtuelles suivantes :

- CPU
- Mémoire
- Disque, réseau et E/S de console
- Unité cryptographique

Analyse syntaxique de l'interface de contrôle XML

Logical Domains Manager exporte une interface de contrôle XML vers Oracle VM Server for SPARC MIB. Oracle VM Server for SPARC MIB effectue une analyse syntaxique de l'interface XML et remplit la MIB. Oracle VM Server for SPARC MIB prend uniquement en charge le domaine de contrôle.

Emission de déroutements SNMP

Oracle VM Server for SPARC MIB interroge régulièrement Logical Domains Manager à la recherche de mises à jour ou de modifications de l'état, puis émet des déroutements SNMP vers les applications de gestion du système.

Indication d'informations sur les erreurs et la récupération

Si Oracle VM Server for SPARC MIB ne peut plus affecter une ressource nécessaire, MIB renvoie une erreur générale à l'application de gestion du système, via l'agent SNMP. Le mécanisme d'émission de déroutements SNMP ne confirme pas l'erreur. Aucun état ni aucun point de contrôle particulier n'est implémenté dans Oracle VM Server for SPARC MIB. Le SMA contenant Oracle VM Server for SPARC MIB est lancé et contrôlé par le processus `init` et l'utilitaire de gestion des services (SMF, Service Management Facility). Si le SMA échoue et se ferme, le SMF redémarre automatiquement le processus, puis le nouveau processus redémarre dynamiquement le module Oracle VM Server for SPARC MIB.

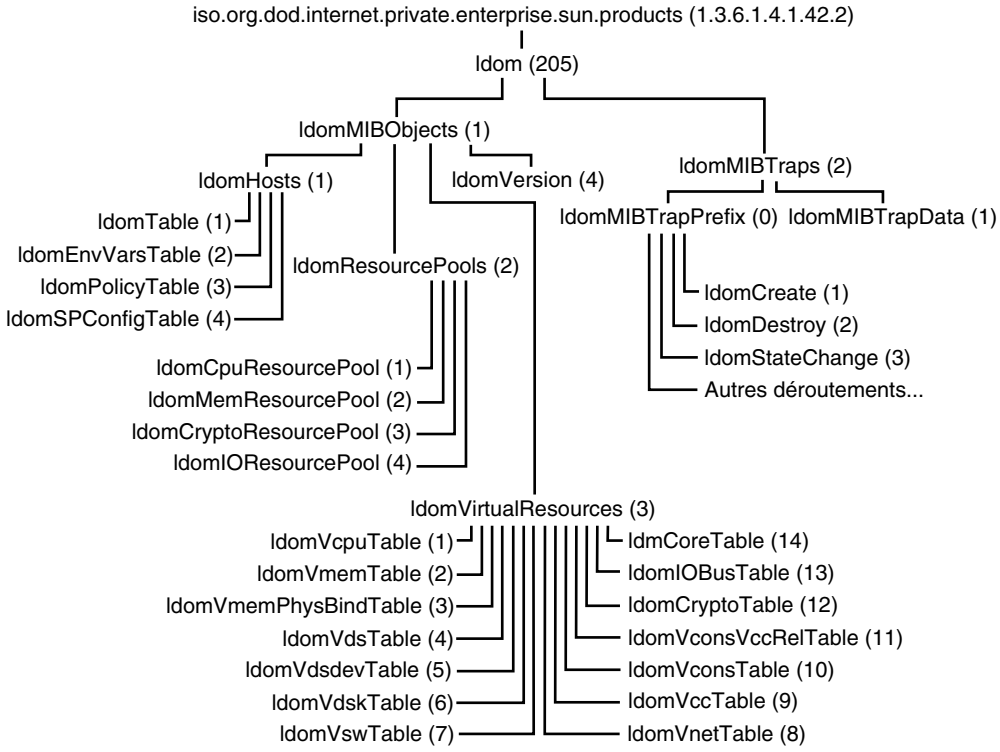
Arborescence d'objets Oracle VM Server for SPARC MIB

Les objets gérés par SNMP sont organisés sous forme d'arborescence. Un identificateur d'objet (OID) consiste en une série de nombres entiers basés sur les noeuds de l'arborescence, séparés par des points. Chaque objet géré possède un OID numérique et un nom textuel associé. Oracle VM Server for SPARC MIB est enregistré en tant que branche `ldom (205)` dans cette partie de l'arborescence d'objets :

```
iso(1).org(3).dod(6).internet(1).private(4).enterprises(1).sun(42).products(2)
```

La figure suivante illustre les principales sous-arborescences de Oracle VM Server for SPARC MIB.

FIGURE 15–2 Arborescence Oracle VM Server for SPARC MIB



Installation et configuration du logiciel Oracle VM Server for SPARC MIB

Cette section présente l'installation et la configuration du logiciel Oracle VM Server for SPARC MIB. Pour plus d'informations sur la gestion SNMP, reportez-vous à la page de manuel `snmpd.conf`(4).

Installation et configuration du logiciel Oracle VM Server for SPARC MIB (liste des tâches)

Le tableau suivant propose des liens vers les tâches permettant d'installer et de configurer le logiciel Oracle VM Server for SPARC MIB.

Tâche	Description	Voir
Installation du package logiciel Oracle VM Server for SPARC MIB sur le domaine `primary`.	Utilisez la commande `pkgadd` pour installer le package `SUNWldmib.v`.	"Procédure d'installation du package logiciel Oracle VM Server for SPARC MIB" à la page 302
Chargement du module Oracle VM Server for SPARC MIB dans le SMA.	Chargez le module `ldomMIB.so` dans le SMA du SE Oracle Solaris afin d'interroger Oracle VM Server for SPARC MIB.	"Procédure de chargement du module Oracle VM Server for SPARC MIB dans le SMA" à la page 303
Suppression du package logiciel Oracle VM Server for SPARC MIB du domaine `primary`.	Utilisez la commande `pkgrm` pour supprimer le package `SUNWldmib.v`.	"Procédure de suppression du package logiciel Oracle VM Server for SPARC MIB" à la page 303

▼ Procédure d'installation du package logiciel Oracle VM Server for SPARC MIB

Cette procédure décrit la procédure d'installation du package logiciel Oracle VM Server for SPARC MIB, `SUNWldmib.v`, qui est inclus dans le logiciel Oracle VM Server for SPARC .2.2.

Une fois ce package installé, vous pouvez configurer votre système afin de charger dynamiquement le module Oracle VM Server for SPARC MIB. Reportez-vous à la section "Procédure de chargement du module Oracle VM Server for SPARC MIB dans le SMA" à la page 303.

Avant de commencer Téléchargez et installez le logiciel Oracle VM Server for SPARC .2.2. Reportez-vous au Chapitre 2, "Installation et activation du logiciel".

- **Ajoutez le package `SUNWldmib.v` au domaine `primary`.**

```
# pkgadd -d . SUNWldmib.v
```

Cette commande installe les fichiers suivants :

- `/opt/SUNWldmib/lib/mibs/SUN-LDOM-MIB.mib`
- `/opt/SUNWldmib/lib/ldomMIB.so`

Procédure de chargement du module Oracle VM Server for SPARC MIB dans le SMA

Le module Oracle VM Server for SPARC MIB, `ldomMIB.so`, doit être chargé dans le SMA du SE Oracle Solaris afin d'interroger Oracle VM Server for SPARC MIB. Le module Oracle VM Server for SPARC MIB est chargé dynamiquement, de sorte que le module est inclus dans l'agent SNMP sans nécessité de recompilation ou de reliaison de l'agent binaire.

Cette procédure décrit la configuration de votre système afin qu'il charge dynamiquement le module Oracle VM Server for SPARC MIB. Les instructions pour charger dynamiquement le module sans redémarrer le SMA sont disponibles dans le *Solaris System Management Agent Developer's Guide*. Pour plus d'informations sur le SMA, reportez-vous au *Solaris System Management Administration Guide*.

1. **Mettez à jour le fichier de configuration SNMP du SMA.**

   Ajoutez la ligne suivante au fichier de configuration `/etc/sma/snmp/snmpd.conf` :

   ```
   dlmod ldomMIB /opt/SUNWldmib/lib/ldomMIB.so
   ```

2. **Redémarrez le SMA.**

   ```
   # svcadm restart svc:/application/management/sma:default
   ```

Procédure de suppression du package logiciel Oracle VM Server for SPARC MIB

Cette procédure décrit la suppression du package logiciel Oracle VM Server for SPARC MIB, `SUNWldmib.v`, et le déchargement du module Oracle VM Server for SPARC MIB du SMA.

1. **Arrêtez le SMA.**

   ```
   # svcadm disable svc:/application/management/sma:default
   ```

2. **Suppression du package logiciel Oracle VM Server for SPARC MIB du domaine `primary`.**

   ```
   # pkgrm SUNWldmib
   ```

3. **Mettez à jour le fichier de configuration SNMP du SMA.**

   Supprimez la ligne que vous avez ajouté au fichier `/etc/sma/snmp/snmpd.conf` au cours de l'installation.

   ```
   dlmod ldomMIB /opt/SUNWldmib/lib/ldomMIB.so
   ```

4. **Redémarrez le SMA.**

   ```
   # svcadm restart svc:/application/management/sma:default
   ```

Gestion de la sécurité

Cette section décrit la procédure de création d'utilisateurs SNMP (Simple Network Management Protocol) version 3 (v3) afin de fournir un accès sécurisé au SMA (System Management Agent). Pour SNMP version 1 (v1) et version 2 (v2c), le mécanisme de contrôle d'accès est la *community string*, qui définit la relation entre un serveur SNMP et ses clients. Cette chaîne contrôle l'accès client au serveur, tout comme un mot de passe contrôle l'accès utilisateur à un système. Reportez-vous au *Solaris System Management Agent Administration Guide*.

Remarque – La création d'utilisateurs `snmpv3` vous permet d'utiliser le SMA dans SNMP avec Oracle VM Server for SPARC MIB. Ce type d'utilisateurs n'interagit en aucun cas ni n'entre en conflit avec les utilisateurs éventuellement configurés via la fonction RBAC (Role-Based Access Control) d'Oracle Solaris pour Logical Domains Manager.

▼ Procédure de création de l'utilisateur snmpv3 initial

Cette procédure décrit la création de l'utilisateur `snmpv3` initial.

Vous pouvez créer des utilisateurs supplémentaires en clonant cet utilisateur initial. Le clonage permet aux utilisateurs créés par la suite d'hériter des types d'authentification et de sécurité de l'utilisateur initial. Ces types peuvent être modifiés ultérieurement.

Lorsque vous clonez l'utilisateur initial, vous définissez des données de clé secrète pour le nouvel utilisateur. Vous devez connaître les mots de passe de l'utilisateur initial et des utilisateurs que vous configurez ultérieurement. Vous pouvez uniquement cloner un utilisateur à la fois à partir de l'utilisateur initial. Pour connaître votre version du SE Solaris, reportez-vous à la section Création d'utilisateurs SNMPv3 supplémentaires en toute sécurité du *Solaris System Management Agent Administration Guide*.

1 Arrêtez le SMA.

```
# svcadm disable -t svc:/application/management/sma:default
```

2 Créez l'utilisateur initial.

```
# /usr/sfw/bin/net-snmp-config --create-snmpv3-user -a my-password initial-user
```

Cette commande crée l'utilisateur *initial-user* avec un mot de passe de votre choix, *my-password*, puis ajoute une entrée au fichier `/etc/sma/snmp/snmpd.conf`. Cette entrée donne à l'utilisateur initial un accès en lecture et en écriture à l'agent.

Remarque – Les mots de passes doivent contenir au moins huit caractères.

---

3 Démarrez le SMA.

```
# svcadm enable svc:/application/management/sma:default
```

4. **Vérifiez que l'utilisateur initial a été créé.**

```
# snmpget -v 3 -u initial-user -l authNoPriv -a MD5 -A my-password localhost sysUpTime.0
```

Contrôle des domaines

Cette section décrit la procédure de contrôle des domaines logiques (domaines) en interrogeant Oracle VM Server for SPARC MIB. Elle fournit également des descriptions des différents types de sorties de la MIB.

Cette section couvre les sujets suivants :

Définition des variables d'environnement

▼ Procédure de définition des variables d'environnement

Avant de pouvoir interroger Oracle VM Server for SPARC MIB, vous devez définir les variables d'environnement pour le shell que vous utilisez. Cette procédure décrit la configuration de ces variables pour les shells C, Bourne et Korn.

- **Définissez les variables d'environnement `PATH`, `MIBDIRS` et `MIBS`.**

  - **Utilisateurs du shell C :**

```
% setenv PATH /usr/sfw/bin:$PATH
% setenv MIBDIRS /opt/SUNWldmib/lib/mibs:/etc/sma/snmp/mibs
% setenv MIBS +SUN-LDOM-MIB
```

  - **Utilisateurs des shells Bourne et Korn :**

```
$ PATH=/usr/sfw/bin:$PATH; export PATH
$ MIBDIRS=/opt/SUNWldmib/lib/mibs:/etc/sma/snmp/mibs; export MIBDIRS
$ MIBS=+SUN-LDOM-MIB; export MIBS
```

Interrogation de Oracle VM Server for SPARC MIB

▼ Procédure de récupération d'objets de Oracle VM Server for SPARC MIB

Lorsqu'un système contient de nombreux domaines, l'agent SNMP risque d'expirer avant de pouvoir répondre à une requête SNMP. Pour augmenter la valeur de temporisation, utilisez l'option `-t` pour spécifier une valeur de temporisation supérieure. Par exemple, la commande `snmpwalk` suivante définit la valeur de temporisation sur 20 secondes :

```
# snmpwalk -t 20 -v1 -c public localhost SUN-LDOM-MIB::ldomTable
```

Vous pouvez également utiliser l'option `-t` pour spécifier la valeur de temporisation pour les commandes `snmpget` et `snmptable`.

- **Récupérez un ou plusieurs objets MIB.**

  - **Récupérez un seul objet MIB.**

    # **snmpget -v** *version* **-c** *community-string host MIB-object*

  - **Récupérez une gamme d'objets MIB.**

    Utilisez la commande `snmpwalk` ou `snmptable`.

    # **snmpwalk -v** *version* **-c** *community-string host MIB-object*
    # **snmptable -v** *version* **-c** *community-string host MIB-object*

Exemple 15–1 Récupération d'un seul objet Oracle VM Server for SPARC MIB (`snmpget`)

La commande `snmpget` suivante demande la valeur de l'objet `ldomVersionMajor`. La commande spécifie `snmpv1` (`-v1`) et une chaîne de communauté (`-c public`) pour l'hôte `localhost`.

```
# snmpget -v1 -c public localhost SUN-LDOM-MIB::ldomVersionMajor.0
SUN-LDOM-MIB::ldomVersionMajor.0 = INTEGER: 1
```

Exemple 15–2 Récupération des valeurs d'un objet à partir de ldomTable (`snmpwalk`)

L'exemple suivant illustre l'utilisation de la commande `snmpwalk` pour récupérer les valeurs d'un objet à partir de `ldomTable`.

- La commande `snmpwalk -v1` suivante renvoie les valeurs de tous les objets de la table `ldomTable` :

```
# snmpwalk -v1 -c public localhost SUN-LDOM-MIB::ldomTable
SUN-LDOM-MIB::ldomName.1 = STRING: primary
SUN-LDOM-MIB::ldomName.2 = STRING: LdomMibTest_1
SUN-LDOM-MIB::ldomAdminState.1 = INTEGER: 0
SUN-LDOM-MIB::ldomAdminState.2 = INTEGER: 0
SUN-LDOM-MIB::ldomOperState.1 = INTEGER: active(1)
SUN-LDOM-MIB::ldomOperState.2 = INTEGER: bound(6)
SUN-LDOM-MIB::ldomNumVCpu.1 = INTEGER: 32
SUN-LDOM-MIB::ldomNumVCpu.2 = INTEGER: 2
SUN-LDOM-MIB::ldomMemSize.1 = INTEGER: 3968
SUN-LDOM-MIB::ldomMemSize.2 = INTEGER: 256
SUN-LDOM-MIB::ldomMemUnit.1 = INTEGER: megabytes(2)
SUN-LDOM-MIB::ldomMemUnit.2 = INTEGER: megabytes(2)
SUN-LDOM-MIB::ldomNumCrypto.1 = INTEGER: 8
SUN-LDOM-MIB::ldomNumCrypto.2 = INTEGER: 0
SUN-LDOM-MIB::ldomNumIOBus.1 = INTEGER: 2
SUN-LDOM-MIB::ldomNumIOBus.2 = INTEGER: 0
SUN-LDOM-MIB::ldomUUID.1 = STRING: c2c3d93b-a3f9-60f6-a45e-f35d55c05fb6
```

```
SUN-LDOM-MIB::ldomUUID.2 = STRING: af0b05f0-d262-e633-af32-a6c4e81fb81c
SUN-LDOM-MIB::ldomMacAddress.1 = STRING: 00:14:4f:86:63:2a
SUN-LDOM-MIB::ldomMacAddress.2 = STRING: 00:14:4f:fa:78:b9
SUN-LDOM-MIB::ldomHostID.1 = STRING: 0x8486632a
SUN-LDOM-MIB::ldomHostID.2 = STRING: 0x84fa78b9
SUN-LDOM-MIB::ldomFailurePolicy.1 = STRING: ignore
SUN-LDOM-MIB::ldomFailurePolicy.2 = STRING: ignore
SUN-LDOM-MIB::ldomMaster.1 = STRING:
SUN-LDOM-MIB::ldomMaster.2 = STRING:
```

- La commande `snmpwalk` suivante utilise `snmpv2c` et `snmpv3` pour récupérer le contenu de `ldomTable` :

```
# snmpwalk -v2c -c public localhost SUN-LDOM-MIB::ldomTable
# snmpwalk -v 3 -u test -l authNoPriv -a MD5 -A testpassword localhost \
SUN-LDOMMIB::ldomTable
```

Exemple 15–3 Récupération des valeurs d'un objet à partir de ldomTable sous forme de tableau (`snmptable`)

L'exemple suivant illustre l'utilisation de la commande `snmptable` pour récupérer les valeurs d'un objet à partir de `ldomTable` sous de tableau.

- La commande `snmptable -v1` suivante affiche le contenu de `ldomTable` sous forme de tableau :

```
# snmptable -v1 -c public localhost SUN-LDOM-MIB::ldomTable
```

- La commande `snmptable` suivante affiche le contenu de `ldomTable` sous forme de tableau à l'aide de `snmpv2c`.

  Notez que pour la commande `v2c` ou `v3` `snmptable`, vous utilisez l'option `-CB` pour spécifier uniquement les requêtes `GETNEXT`, et non pas `GETBULK`, pour récupérer des données.

```
# snmptable -v2c -CB -c public localhost SUN-LDOM-MIB::ldomTable
```

Récupération d'informations Oracle VM Server for SPARC MIB

Cette section décrit les informations que vous pouvez récupérer dans Oracle VM Server for SPARC MIB sous forme de tableaux ou d'objets scalaires.

Table de domaine (ldomTable)

`ldomTable` permet de représenter tous les domaines du système. Les informations incluent les contraintes de ressource pour les CPU virtuelles, la mémoire, les unités cryptographiques et les bus d'E/S. Le tableau inclut également d'autres informations de domaine, telles que l'identifiant universel unique (UUID), l'adresse MAC, l'ID hôte, la stratégie de panne et le domaine maître.

TABLEAU 15–1 Table de domaine (ldomTable)

Nom	Type de données	Accès	Description
`ldomIndex`	Nombre entier	Non accessible	Nombre entier utilisé en tant qu'index de cette table
`ldomName`	Chaîne d'affichage	Lecture seule	Nom du domaine
`ldomAdminState`	Nombre entier	Lecture/écriture	Démarre ou arrête le domaine pour une gestion active : ■ La valeur 1 démarre le domaine ■ La valeur 2 arrête le domaine
`ldomOperState`	Nombre entier	Lecture seule	Etat actuel du domaine, qui peut prendre l'une des valeurs suivantes : ■ 1 : Actif ■ 2 : Arrêt ■ 3 : Inactif ■ 4 : Association ■ 5 : Dissociation ■ 6 : Lien ■ 7 : Démarrage
`ldomNumVCPU`	Nombre entier	Lecture seule	Nombre de CPU virtuelles utilisées. Si l'état du domaine est Inactif, cette valeur correspond au nombre demandé de CPU virtuelles.
`ldomMemSize`	Nombre entier	Lecture seule	Quantité de mémoire virtuelle utilisée. Si l'état du domaine est Inactif, cette valeur correspond à la taille de mémoire virtuelle demandée.
`ldomMemUnit`	Nombre entier	Lecture seule	L'une des unités de mémoire suivantes : ■ 1 = Ko ■ 2 = Mo ■ 3 = Go ■ 4 = octets Si cela n'est pas spécifié, la valeur de l'unité est octets.
`ldomNumCrypto`	Nombre entier	Lecture seule	Nombre d'unités cryptographiques utilisées. Si l'état du domaine est Inactif, cette valeur correspond au nombre demandé d'unités cryptographiques.
`ldomNumIOBus`	Nombre entier	Lecture seule	Nombre de périphériques d'E/S physiques utilisés.

TABLEAU 15–1 Table de domaine (ldomTable) *(Suite)*

Nom	Type de données	Accès	Description
`ldomUUID`	Chaîne d'affichage	Lecture seule	UUID du domaine.
`ldomMacAddress`	Chaîne d'affichage	Lecture seule	Adresse MAC du domaine.
`ldomHostID`	Chaîne d'affichage	Lecture seule	ID hôte du domaine.
`ldomFailurePolicy`	Chaîne d'affichage	Lecture seule	Stratégie de panne du domaine maître, qui peut adopter l'une des valeurs suivantes : `ignore`, `panic`, `reset` ou `stop`.
`ldomMaster`	Chaîne d'affichage	Lecture seule	Désignation d'un maximum de quatre domaines maîtres pour un domaine esclave.

Table des variables d'environnement (ldomEnvVarsTable)

`ldomEnvVarsTable` décrit la variable d'environnement OpenBoot PROM que tous les domaines utilisent.

TABLEAU 15–2 Table des variables d'environnement (ldomEnvVarsTable)

Nom	Type de données	Accès	Description
`ldomEnvVarsLdomIndex`	Nombre entier	Lecture seule	Nombre entier utilisé comme index dans `ldomTable` et représentant le domaine contenant les variables d'environnement OpenBoot PROM.
`ldomEnvVarsIndex`	Nombre entier	Lecture seule	Nombre entier utilisé pour indexer les variables d'environnement OpenBoot PROM dans cette table.
`ldomEnvVarsName`	Chaîne d'affichage	Lecture seule	Nom de la variable OpenBoot PROM.
`ldomEnvVarsValue`	Chaîne d'affichage	Lecture seule	Valeur de la variable OpenBoot PROM.

Table de stratégie de domaine (ldomPolicyTable)

`ldomPolicyTable` décrit les stratégies DRM (Dynamic Resource Management, gestion dynamique des ressources) qui s'appliquent à tous les domaines.

TABLEAU 15–3 Table de stratégie de domaine (ldomPolicyTable)

Nom	Type de données	Accès	Description
`ldomPolicyLdomIndex`	Nombre entier	Lecture seule	Nombre entier utilisé comme index dans `ldomTable` et représente le domaine contenant la stratégie DRM.
`ldomPolicyIndex`	Nombre entier	Non accessible	Nombre entier utilisé pour indexer la stratégie DRM dans cette table.
`ldomPolicyName`	Chaîne d'affichage	Lecture seule	Nom de la stratégie.
`ldomPolicyStatus`	Chaîne d'affichage	Lecture seule	Etat de la stratégie.
`ldomPolicyPriority`	Nombre entier	Lecture seule	Priorité appliquée pour déterminer la stratégie DRM à sélectionner en cas de chevauchement.
`ldomPolicyVcpuMin`	Nombre entier	Lecture seule	Nombre minimal de CPU virtuelles pour un domaine.
`ldomPolicyVcpuMax`	Nombre entier	Lecture seule	Nombre maximal de CPU virtuelles pour un domaine. La valeur `unlimited` utilise la valeur maximale de 2 147 483 647.
`ldomPolicyUtilLower`	Nombre entier	Lecture seule	Niveau d'utilisation inférieur auquel l'analyse de stratégie est lancée.
`ldomPolicyUtilUpper`	Nombre entier	Lecture seule	Niveau d'utilisation supérieur auquel l'analyse de stratégie est lancée.
`ldomPolicyTodBegin`	Chaîne d'affichage	Lecture seule	Heure de démarrage effectif d'une stratégie au format *hh*:*mm*:[*ss*]
`ldomPolicyTodEnd`	Chaîne d'affichage	Lecture seule	Heure de fin effective d'une stratégie au format *hh*:*mm*:[*ss*]
`ldomPolicySampleRate`	Nombre entier	Lecture seule	Temps du cycle de ressources en secondes.
`ldomPolicyElasticMargin`	Nombre entier	Lecture seule	Quantité de tampon entre la propriété `util-lower` (`ldomPolicyUtilLower`) et le nombre de CPU virtuelles libres afin d'éviter les fluctuations lorsqu'il y a peu de CPU virtuelles.
`ldomPolicyAttack`	Nombre entier	Lecture seule	Quantité maximale d'une ressource qui sera ajoutée au cours d'un cycle de contrôle de ressources. La valeur `unlimited` utilise la valeur maximale de 2 147 483 647.

TABLEAU 15–3 Table de stratégie de domaine (ldomPolicyTable) *(Suite)*

Nom	Type de données	Accès	Description
`ldomPolicyDecay`	Nombre entier	Lecture seule	Quantité maximale d'une ressource qui sera supprimée au cours d'un cycle de contrôle de ressources.

Table de configuration du processeur de service (ldomSPConfigTable)

`ldomSPConfigTable` décrit les configurations du processeur de service (SP) pour tous les domaines.

TABLEAU 15–4 Table de configuration du processeur de service (ldomSPConfigTable)

Nom	Type de données	Accès	Description
`ldomSPConfigIndex`	Nombre entier	Non accessible	Nombre entier utilisé pour indexer une configuration SP dans cette table.
`ldomSPConfigName`	Chaîne d'affichage	Lecture seule	Nom de la configuration SP.
`ldomSPConfigStatus`	Chaîne d'affichage	Lecture seule	Etat de la configuration SP.

Pool de ressources de domaine et variables scalaires

Les ressources suivantes peuvent être affectées aux domaines :

- CPU virtuelle (`vcpu`)
- Mémoire (`mem`)
- Unité cryptographique (`mau`)
- Commutateur virtuel (`vsw`)
- Réseau virtuel (`vnet`)
- Serveur de disque virtuel (`vds`)
- Périphérique de serveur de disque virtuel (`vdsdev`)
- Disque virtuel (`vdisk`)
- Concentrateur de console virtuelle (`vcc`)
- Console virtuelle (`vcons`)
- Périphérique d'E/S physique (`io`)

Les variables MIB scalaires suivantes sont utilisées pour représenter les pools de ressources et leurs propriétés.

TABLEAU 15–5 Variables scalaires pour un pool de ressources CPU

Nom	Type de données	Accès	Description
`ldomCpuRpCapacity`	Nombre entier	Lecture seule	Réservation maximale autorisées par le pool de ressources dans `ldomCpuRpCapacityUnit`.
`ldomCpuRpReserved`	Nombre entier	Lecture seule	Fréquence d'horloge accumulée du processeur de la CPU, en MHz, actuellement réservée dans le pool de ressources.
`ldomCpuRpCapacityUnit` et `ldomCpuRpReservedUnit`	Nombre entier	Lecture seule	L'une des unités d'allocation de CPU suivantes : ■ 1 = MHz ■ 2 = GHz La valeur par défaut est MHz.

TABLEAU 15–6 Variables scalaires pour un pool de ressources de mémoire

Nom	Type de données	Accès	Description
`ldomMemRpCapacity`	Nombre entier	Lecture seule	Réservation maximale autorisées par le pool de ressources dans `MemRpCapacityUnit`.
`ldomMemRpReserved`	Nombre entier	Lecture seule	Quantité de mémoire, en `MemRpReservedUnit`, actuellement réservée dans le pool de ressources.
`ldomMemRpCapacityUnit` et `ldomMemRpReservedUnit`	Nombre entier	Lecture seule	L'une des unités d'allocation de mémoire suivantes : ■ 1 = Ko ■ 2 = Mo ■ 3 = Go ■ 4 = octets Si cela n'est pas spécifié, la valeur de l'unité est octets.

TABLEAU 15–7 Variables scalaires pour un pool de ressources cryptographiques

Nom	Type de données	Accès	Description
`ldomCryptoRpCapacity`	Nombre entier	Lecture seule	Réservation maximale autorisée par le pool de ressources.

TABLEAU 15–7 Variables scalaires pour un pool de ressources cryptographiques *(Suite)*

Nom	Type de données	Accès	Description
`ldomCryptoRpReserved`	Nombre entier	Lecture seule	Nombre d'unités cryptographiques actuellement réservées dans le pool de ressources.

TABLEAU 15–8 Variables scalaires pour un pool de ressources de bus d'E/S

Nom	Type de données	Accès	Description
`ldomIOBusRpCapacity`	Nombre entier	Lecture seule	Réservation maximale autorisée par le pool.
`ldomIOBusRpReserved`	Nombre entier	Lecture seule	Nombre de bus d'E/S actuellement réservés dans le pool de ressources.

Table de CPU virtuelle (ldomVcpuTable)

`ldomVcpuTable` décrit les CPU virtuelles que tous les domaines utilisent.

TABLEAU 15–9 Table de CPU virtuelle (ldomVcpuTable)

Nom	Type de données	Accès	Description
`ldomVcpuLdomIndex`	Nombre entier	Lecture seule	Nombre entier utilisé comme index dans `ldomTable` qui représente le domaine contenant la CPU virtuelle.
`ldomVcpuIndex`	Nombre entier	Non accessible	Nombre entier utilisé pour indexer la CPU virtuelle dans cette table.
`ldomVcpuDeviceID`	Chaîne d'affichage	Lecture seule	Identificateur de la CPU virtuelle (VID).

TABLEAU 15–9 Table de CPU virtuelle (ldomVcpuTable) *(Suite)*

Nom	Type de données	Accès	Description
`ldomVcpuOperationalStatus`	Nombre entier	Lecture seule	L'un des états de CPU suivants : 1 = Inconnu 2 = Autre 3 = OK 4 = Endommagé 5 = Chargé 6 = Panne prédictive 7 = Erreur 8 = Erreur irrécupérable 9 = Démarrage 10 = Arrêt 11 = Arrêté 12 = En service 13 = Aucun contact 14 = Communication perdue 15 = Abandon 16 = Inexploité 17 = Entité de soutien en erreur 18 = Terminé 19 = Mode d'alimentation La valeur par défaut est 1 (Inconnue) car Logical Domains Manager ne fournit pas l'état de la CPU.
`ldomVcpuPhysBind`	Chaîne d'affichage	Lecture seule	Liaison physique (PID). Contient l'identificateur du thread matériel (strand) assigné à cette CPU virtuelle. Cet ID identifie de manière unique le coeur et la puce.

TABLEAU 15–9 Table de CPU virtuelle (ldomVcpuTable) *(Suite)*

Nom	Type de données	Accès	Description
`ldomVcpuPhysBindUsage`	Nombre entier	Lecture seule	Indique la quantité (en MHz) de la capacité totale du thread utilisée par cette CPU virtuelle. Par exemple, si un thread peut s'exécuter à 1 GHz au maximum, Si une moitié seulement de cette capacité est allouée à cette CPU virtuelle (50 % du thread), la valeur de la propriété est de 500.
`ldomVcpuCoreID`	Chaîne d'affichage	Lecture seule	Identificateur du coeur (ID coeur)
`ldomVcpuUtilPercent`	Chaîne d'affichage	Lecture seule	Indique le pourcentage d'utilisation de la CPU virtuelle

Tables de mémoire virtuelle

L'espace mémoire d'un domaine est appelé *mémoire réelle*, ou, en d'autres termes, *mémoire virtuelle*. L'espace mémoire de la plate-forme hôte détecté par l'hyperviseur est appelé *mémoire physique*. L'hyperviseur mappe les blocs de mémoire physique afin de former un bloc de mémoire réelle qui sera utilisé par un domaine.

L'exemple suivant indique que la taille de la mémoire requise peut être divisée entre deux blocs mémoire plutôt que d'être assignée à un seul grand bloc de mémoire. Si un domaine demande 521 Mo de mémoire réelle, deux blocs de 256 Mo peuvent être assignés à la mémoire sur le système hôte en tant que mémoire physique à l'aide du format {*physical-address*, *real-address*, *size*}.

```
{0x1000000, 0x1000000, 256}, {0x2000000, 0x2000000,256}
```

Un domaine peut avoir jusqu'à 64 tranches de mémoire physique assignées à un domaine invité. Ainsi, une table auxiliaire, plutôt qu'une chaîne d'affichage, est utilisée pour contenir chaque tranche de mémoire. Une chaîne d'affichage dispose d'une limite de 255 caractères.

Table de mémoire virtuelle (ldomVmemTable)

`ldomVmemTable` décrit les propriétés de la mémoire virtuelle utilisées par les domaines.

TABLEAU 15–10 Table de mémoire virtuelle (ldomVmemTable)

Nom	Type de données	Accès	Description
`ldomVmemLdomIndex`	Nombre entier	Lecture seule	Nombre entier utilisé comme index dans `ldomTable` qui représente le domaine contenant la mémoire virtuelle.

TABLEAU 15–10 Table de mémoire virtuelle (ldomVmemTable) *(Suite)*

Nom	Type de données	Accès	Description
`ldomVmemIndex`	Nombre entier	Non accessible	Nombre entier utilisé pour indexer la mémoire virtuelle dans cette table.
`ldomVmemNumberofBlocks`	Nombre entier	Lecture seule	Nombre de blocs de la mémoire virtuelle.

Table de liaison physique de la mémoire virtuelle (ldomVmemPhysBindTable)

`ldomVmemPhysBindTable` est une table auxiliaire qui contient des tranches de mémoire physique pour tous les domaines.

TABLEAU 15–11 Table de liaison physique de la mémoire virtuelle (ldomVmemPhysBindTable)

Nom	Type de données	Accès	Description
`ldomVmemPhysBindLdomIndex`	Nombre entier	Lecture seule	Nombre entier utilisé comme index dans `ldomTable` qui représente le domaine contenant les tranches de mémoire physique.
`ldomVmemPhysBind`	Chaîne d'affichage	Lecture seule	Liste de la mémoire physique mappée à ce bloc de mémoire virtuelle au format suivant : {*physical-address*, *real-address*, *size*}

Table de disques virtuels

Un service de disque virtuel (`vds`) et le périphérique physique auquel il est mappé (`vdsdev`) fournissent la capacité de disque virtuel à la technologie Oracle VM Server for SPARC. Un service de disque virtuel exporte un certain nombre de volumes locaux (disques physiques ou systèmes de fichiers). Lorsqu'un service de disque virtuel est spécifié, les éléments suivants sont inclus :

- Chemin d'accès `/dev` complet au périphérique de sauvegarde (`vdsdev`)
- Nom unique (nom de volume) du périphérique ajouté au service

Un ou plusieurs disques, systèmes de fichiers et une ou plusieurs tranches de disque peuvent être liés à un seul service de disque. Chaque disque possède un nom unique et un nom de volume. Le nom de volume est utilisé lorsque le disque est lié au service. Logical Domains Manager crée des clients de disques virtuels (`vdisk`) à partir du service de disque virtuel et de ses volumes logiques.

Table des services de disque virtuel (ldomVdsTable)

`ldomVdsTable` décrit les services de disques virtuels pour tous les domaines.

TABLEAU 15–12 Table des services de disque virtuel (ldomVdsTable)

Nom	Type de données	Accès	Description
`ldomVdsLdomIndex`	Nombre entier	Lecture seule	Nombre entier utilisé comme index dans `ldomTable` qui représente le domaine contenant le service de disque virtuel.
`ldomVdsIndex`	Nombre entier	Non accessible	Nombre entier utilisé pour indexer le service de disque virtuel dans cette table.
`ldomVdsServiceName`	Chaîne d'affichage	Lecture seule	Nom du service du service de disque virtuel. La valeur de propriété est le *service-name* spécifié par la commande `ldm add-vds`.
`ldomVdsNumofAvailVolume`	Nombre entier	Lecture seule	Nombre de volumes logiques exportés par ce service de disque virtuel.
`ldomVdsNumofUsedVolume`	Nombre entier	Lecture seule	Nombre de volumes logiques utilisés (liés) à ce service de disque virtuel.

Table des périphériques de services de disques virtuels (ldomVdsdevTable)

`ldomVdsdevTable` décrit les périphériques de services de disques virtuels utilisés par tous les services de disques virtuels.

TABLEAU 15–13 Table des périphériques de services de disques virtuels (ldomVdsdevTable)

Nom	Type de données	Accès	Description
`ldomVdsdevVdsIndex`	Nombre entier	Lecture seule	Nombre entier utilisé comme index dans `ldomVdsTable` qui représente le service de disque virtuel contenant le périphérique de disque virtuel.
`ldomVdsdevIndex`	Nombre entier	Non accessible	Nombre entier utilisé pour indexer le périphérique de service de disque virtuel dans cette table.
`ldomVdsdevVolumeName`	Chaîne d'affichage	Lecture seule	Nom de volume du périphérique de service de disque virtuel. Cette propriété spécifie un nom unique pour le périphérique ajouté au service de disque virtuel. Ce nom est exporté par le service de disque virtuel vers les clients afin d'ajouter ce périphérique. La valeur de propriété est le *volume-name* spécifié par la commande `ldm add-vdsdev`.

TABLEAU 15–13 Table des périphériques de services de disques virtuels (ldomVdsdevTable) *(Suite)*

Nom	Type de données	Accès	Description
`ldomVdsdevDevPath`	Chaîne d'affichage	Lecture seule	Nom du chemin du périphérique de disque physique. La valeur de propriété est le *backend* spécifié par la commande `ldm add-vdsdev`.
`ldomVdsdevOptions`	Chaîne d'affichage	Lecture seule	Une ou plusieurs des options du périphérique de disque : `ro`, `slice` ou `excl`
`ldomVdsdevMPGroup`	Chaîne d'affichage	Lecture seule	Nom de groupe multivoie du périphérique de disque.

Table de disques virtuels (ldomVdiskTable)

`ldomVdiskTable` décrit les disques virtuels pour tous les domaines.

TABLEAU 15–14 Table de disques virtuels (ldomVdiskTable)

Nom	Type de données	Accès	Description
`ldomVdiskLdomIndex`	Nombre entier	Lecture seule	Nombre entier utilisé comme index dans `ldomTable` qui représente le domaine contenant le périphérique de disque virtuel.
`ldomVdiskVdsDevIndex`	Nombre entier	Lecture seule	Nombre entier utilisé comme index dans `ldomVdsdevTable` qui représente le périphérique de services de disques virtuels.
`ldomVdiskIndex`	Nombre entier	Non accessible	Nombre entier utilisé pour indexer le disque virtuel dans cette table.
`ldomVdiskName`	Chaîne d'affichage	Lecture seule	Nom du disque virtuel. La valeur de propriété est le *disk-name* spécifié par la commande `ldm add-vdisk`.
`ldomVdiskTimeout`	Nombre entier	Lecture seule	Délai d'attente, en secondes, pour l'établissement d'une connexion entre un client de disque virtuel et un serveur de disque virtuel.
`ldomVdiskID`	Chaîne d'affichage	Lecture seule	Identificateur du disque virtuel.

La figure suivante présente la manière dont les index sont utilisés pour définir les relations entre les tables des disques virtuels et la table de domaines. Les index sont utilisés comme suit :

- `ldomIndex` dans `ldomVdsTable` et `ldomVdiskTable` pointe vers `ldomTable`.
- `VdsIndex` dans `ldomVdsdevTable` pointe vers `ldomVdsTable`.
- `VdsDevIndex` dans `ldomVdiskTable` pointe vers `ldomVdsdevTable`.

FIGURE 15–3 Relations entre les tables de disques virtuels et la table de domaines

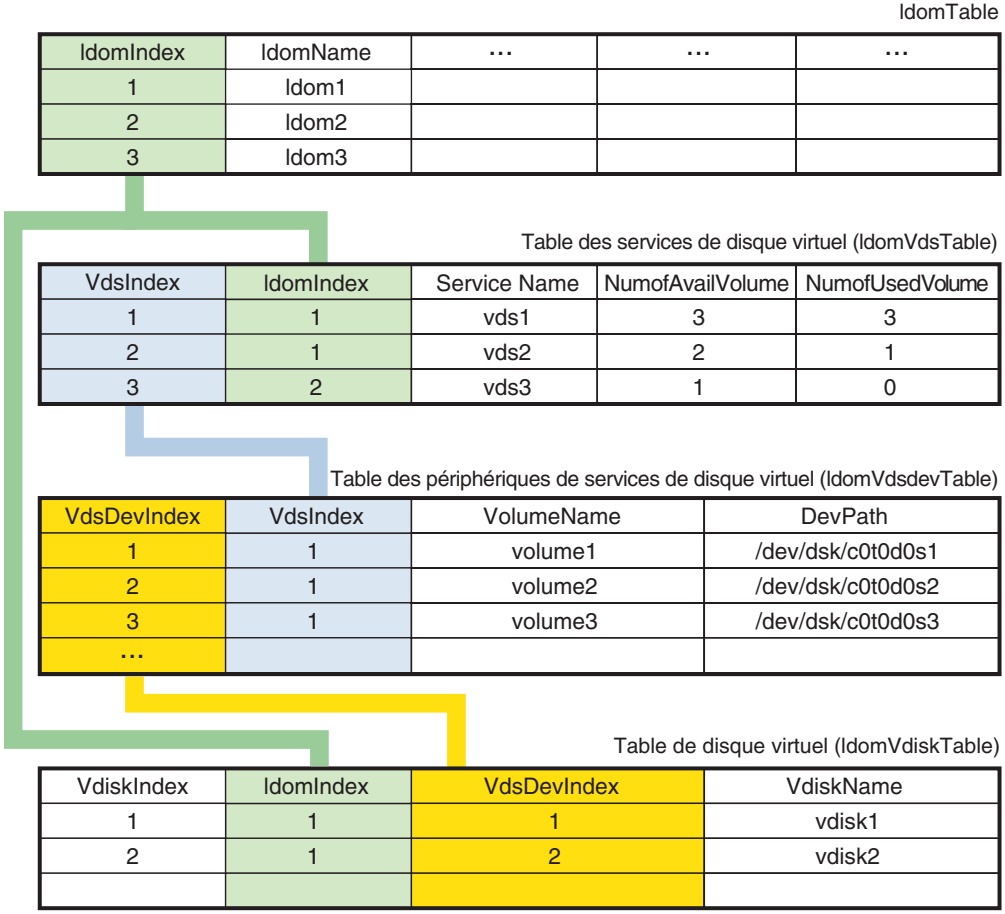


Table de réseaux virtuels

La prise en charge du réseau virtuel Oracle VM Server for SPARC permet aux domaines invités de communiquer entre eux, de même qu'avec des hôtes externes, via un périphérique Ethernet physique. Le réseau virtuel contient les composants principaux suivants :

- Commutateur virtuel (`vsw`)
- Périphérique réseau virtuel (`vnet`)

Après avoir créé un commutateur virtuel sur un domaine de service, vous pouvez lier un périphérique réseau physique au commutateur virtuel. Ensuite, vous pouvez créer un périphérique réseau virtuel pour un domaine qui utilise le service de commutateurs virtuels pour la communication. Le service de commutateurs virtuels communique avec d'autres domaines en se connectant au même commutateur virtuel. Le service de commutateurs virtuels communique avec des hôtes externes si un périphérique physique est lié au commutateur virtuel.

Table des services de commutateurs virtuels (ldomVswTable)

`ldomVswTable` décrit les services de commutateurs virtuels pour tous les domaines.

TABLEAU 15–15 Table des services de commutateurs virtuels (ldomVswTable)

Nom	Type de données	Accès	Description
`ldomVswLdomIndex`	Nombre entier	Lecture seule	Nombre entier utilisé comme index dans `ldomTable` qui représente le domaine contenant le service de commutateur virtuel.
`ldomVswIndex`	Nombre entier	Non accessible	Nombre entier utilisé pour indexer le périphérique de commutateur virtuel dans cette table.
`ldomVswServiceName`	Chaîne d'affichage	Lecture seule	Nom du service de commutateur virtuel.
`ldomVswMacAddress`	Chaîne d'affichage	Lecture seule	Adresse MAC utilisée par le commutateur virtuel.
`ldomVswPhysDevPath`	Chaîne d'affichage	Lecture seule	Chemin d'accès au périphérique physique du commutateur réseau virtuel. La valeur de propriété est nulle si aucun périphérique physique n'est lié au commutateur virtuel.
`ldomVswMode`	Chaîne d'affichage	Lecture seule	La valeur est `mode=sc` pour les noeuds de cluster en cours d'exécution.
`ldomVswDefaultVlanID`	Chaîne d'affichage	Lecture seule	ID VLAN par défaut du commutateur virtuel.

TABLEAU 15–15 Table des services de commutateurs virtuels (ldomVswTable) *(Suite)*

Nom	Type de données	Accès	Description
`ldomVswPortVlanID`	Chaîne d'affichage	Lecture seule	ID VLAN de port du commutateur virtuel.
`ldomVswVlanID`	Chaîne d'affichage	Lecture seule	ID VLAN du commutateur virtuel.
`ldomVswLinkprop`	Chaîne d'affichage	Lecture seule	La valeur `linkprop=phys-state` permet de rapporter l'état du lien en fonction du périphérique réseau physique.
`ldomVswMtu`	Nombre entier	Lecture seule	Unité de transmission maximale (MTU) d'un périphérique de commutateur virtuel.
`ldomVswID`	Chaîne d'affichage	Lecture seule	Identificateur du périphérique de commutateur virtuel.
`ldomVswInterVnetLink`	Chaîne d'affichage	Lecture seule	Etat de l'assignation de canal LDC pour les communications inter-vnet. La valeur est `on` ou `off`.

Table des périphériques réseau virtuels (ldomVnetTable)

`ldomVnetTable` décrit les périphériques réseau virtuels pour tous les domaines.

TABLEAU 15–16 Table des périphériques réseau virtuels (ldomVnetTable)

Nom	Type de données	Accès	Description
`ldomVnetLdomIndex`	Nombre entier	Lecture seule	Nombre entier utilisé comme index dans `ldomTable` qui représente le domaine contenant le périphérique réseau virtuel.
`ldomVnetVswIndex`	Nombre entier	Lecture seule	Nombre entier utilisé comme index dans la table du service de commutateurs virtuels.
`ldomVnetIndex`	Nombre entier	Non accessible	Nombre entier utilisé pour indexer le périphérique réseau virtuel dans cette table.
`ldomVnetDevName`	Chaîne d'affichage	Lecture seule	Nom du périphérique réseau virtuel. La valeur de propriété est la propriété `net-dev` spécifiée par la commande `ldm add-vnet`.

TABLEAU 15–16 Table des périphériques réseau virtuels (ldomVnetTable) *(Suite)*

Nom	Type de données	Accès	Description
`ldomVnetDevMacAddress`	Chaîne d'affichage	Lecture seule	Adresse MAC du périphérique réseau. La valeur de propriété est la propriété `mac-addr` spécifiée par la commande `ldm add-vnet`.
`ldomVnetMode`	Chaîne d'affichage	Lecture seule	La valeur `mode=hybrid` permet d'utiliser l'E/S hybride NIU sur le périphérique réseau physique.
`ldomVnetPortVlanID`	Chaîne d'affichage	Lecture seule	ID VLAN de port du périphérique réseau virtuel
`ldomVnetVlanID`	Chaîne d'affichage	Lecture seule	ID VLAN du périphérique réseau virtuel
`ldomVnetLinkprop`	Chaîne d'affichage	Lecture seule	La valeur `linkprop=phys-state` permet de rapporter l'état du lien en fonction du périphérique réseau physique.
`ldomVnetMtu`	Nombre entier	Lecture seule	MTU d'un périphérique réseau virtuel.
`ldomVnetID`	Chaîne d'affichage	Lecture seule	Identificateur du périphérique réseau virtuel.

Table de consoles virtuelles

Le domaine de service Oracle VM Server for SPARC fournit un service de terminal réseau virtuel (`vNTS`). `vNTS` fournit un service de consoles virtuelles, appelé concentrateur de consoles virtuelles (`vcc`), avec tout une gamme de numéros de ports. Chaque concentrateur de consoles virtuelles possède plusieurs groupes de consoles (`vcons`), et un numéro de port est assigné à chacun de ces groupes. Chaque groupe peut contenir plusieurs domaines.

Table des concentrateurs de consoles virtuelles (ldomVccTable)

`ldomVccTable` décrit les concentrateurs de consoles virtuelles pour tous les domaines.

TABLEAU 15–17 Table des concentrateurs de consoles virtuelles (ldomVccTable)

Nom	Type de données	Accès	Description
`ldomVccLdomIndex`	Nombre entier	Lecture seule	Nombre entier utilisé comme index dans `ldomTable` qui représente le domaine contenant le service de consoles virtuelles.
`ldomVccIndex`	Nombre entier	Non accessible	Nombre entier utilisé pour indexer le concentrateur de console virtuelle dans cette table.

TABLEAU 15–17 Table des concentrateurs de consoles virtuelles (ldomVccTable) *(Suite)*

Nom	Type de données	Accès	Description
`ldomVccName`	Chaîne d'affichage	Lecture seule	Nom du concentrateur de console virtuelle. La valeur de propriété est le *vcc-name* spécifié par la commande `ldm add-vcc`.
`ldomVccPortRangeLow`	Nombre entier	Lecture seule	Limite inférieure de la plage de ports TCP qui sera utilisée par le concentrateur de consoles virtuelles. La valeur de propriété est la portion *x* de la `port-range` spécifiée par la commande `ldm add-vcc`.
`ldomVccPortRangeHigh`	Nombre entier	Lecture seule	Limite supérieure de la plage de ports TCP qui sera utilisée par le concentrateur de consoles virtuelles. La valeur de propriété est la portion *y* de la `port-range` spécifiée par la commande `ldm add-vcc`.

Table des groupes de consoles virtuelles (ldomVconsTable)

`ldomVconsTable` écrit les groupes de consoles virtuelles pour tous les services de consoles virtuelles.

TABLEAU 15–18 Table des groupes de consoles virtuelles (ldomVconsTable)

Nom	Type de données	Accès	Description
`ldomVconsIndex`	Nombre entier	Non accessible	Nombre entier utilisé pour indexer un groupe virtuel dans cette table.
`ldomVconsGroupName`	Chaîne d'affichage	Lecture seule	Nom du groupe auquel lier la console virtuelle. La valeur de propriété est le `group` spécifié par la commande `ldm set-vcons`.
`ldomVconsPortNumber`	Nombre entier	Lecture seule	Numéro de port assigné à ce groupe. La valeur de propriété est le `port` spécifié par la commande `ldm set-vcons`.

Table des relations de consoles virtuelles (ldomVconsVccRelTable)

`ldomVconsVccRelTable` contient des valeurs d'index permettant d'afficher les relations entre les tables d'un domaine, d'un concentrateur de consoles virtuelles et de groupes de consoles.

TABLEAU 15–19 Table des relations de consoles virtuelles (ldomVconsVccRelTable)

Nom	Type de données	Accès	Description
`ldomVconsVccRelVconsIndex`	Nombre entier	Lecture seule	Valeur de `ldomVconsIndex` dans `ldomVconsTable`.
`ldomVconsVccRelLdomIndex`	Nombre entier	Lecture seule	Valeur de `ldomIndex` dans `ldomTable`.
`ldomVconsVccRelVccIndex`	Nombre entier	Lecture seule	Valeur de `ldomVccIndex` dans `ldomVccTable`

La figure suivante présente la manière dont les index sont utilisés pour définir les relations entre les tables de consoles virtuelles et la table de domaines. Les index sont utilisés comme suit :

- `ldomIndex` dans `ldomVccTable` et `ldomVconsVccRelTable` pointe vers `ldomTable`.
- `VccIndex` dans `ldomVconsVccRelTable` pointe vers `ldomVccTable`.
- `VconsIndex` dans `ldomVconsVccRelTable` pointe vers `ldomVconsTable`.

FIGURE 15–4 Relations entre les tables de consoles virtuelles et la table de domaines

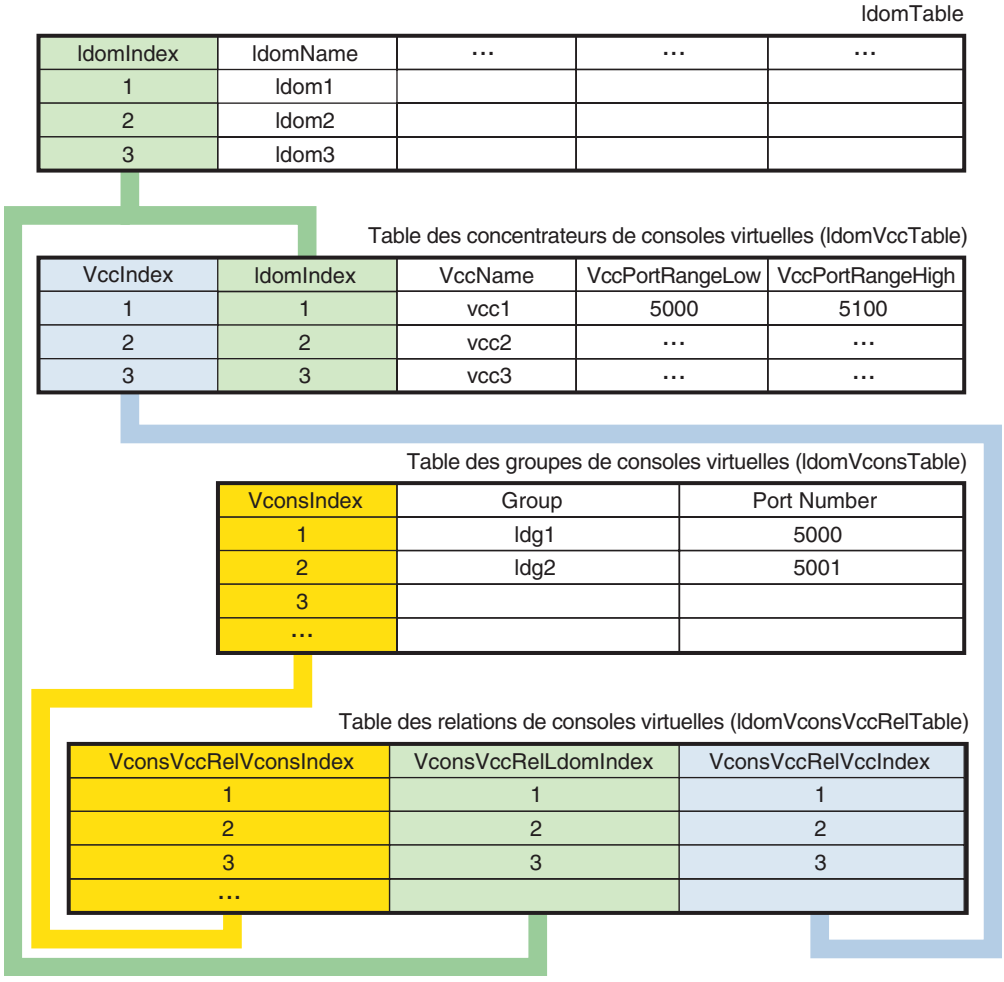


Table des unités cryptographiques (ldomCryptoTable)

`ldomCryptoTable` décrit les unités cryptographiques utilisées par tous les domaines. Une unité cryptographique est parfois appelée unité arithmétique modulaire (MAU, Modular Arithmetic Unit).

TABLEAU 15–20 Table des unités cryptographiques (ldomCryptoTable)

Nom	Type de données	Accès	Description
`ldomCryptoLdomIndex`	Nombre entier	Lecture seule	Nombre entier utilisé comme index dans `ldomTable` qui représente le domaine contenant l'unité cryptographique.
`ldomCryptoIndex`	Nombre entier	Non accessible	Nombre entier utilisé pour indexer l'unité cryptographique dans cette table.
`ldomCryptoCpuSet`	Chaîne d'affichage	Lecture seule	Liste des CPU mappées à `MAU-unit cpuset`. Par exemple, `{0, 1, 2, 3}`.

Table des E/S de bus (ldomIOBusTable)

`ldomIOBusTable` décrit les périphériques d'E/S physique et les bus PCI utilisés par tous les domaines.

TABLEAU 15–21 Table des E/S de bus (ldomIOBusTable)

Nom	Type de données	Accès	Description
`ldomIOBusLdomIndex`	Nombre entier	Lecture seule	Nombre entier utilisé comme index dans `ldomTable` qui représente le domaine contenant le bus d'E/S.
`ldomIOBusIndex`	Nombre entier	Non accessible	Nombre entier utilisé pour indexer le bus d'E/S dans cette table.
`ldomIOBusName`	Chaîne d'affichage	Lecture seule	Nom du périphérique d'E/S physique.
`ldomIOBusPath`	Chaîne d'affichage	Lecture seule	Chemin d'accès au périphérique d'E/S physique.
`ldomIOBusOptions`	Chaîne d'affichage	Lecture seule	Options du périphérique d'E/S physique.

Table des coeurs (ldomCoreTable)

`ldomCoreTable` décrit les informations relatives au coeur, telles que `core-id` et `cpuset`, pour tous les domaines.

TABLEAU 15–22 Table des coeurs (ldomCoreTable)

Nom	Type de données	Accès	Description
`ldomCoreLdomIndex`	Nombre entier	Lecture seule	Nombre entier utilisé comme index dans `ldomTable` qui représente le domaine contenant le coeur.

TABLEAU 15–22 Table des coeurs (ldomCoreTable) *(Suite)*

Nom	Type de données	Accès	Description
`ldomCoreIndex`	Nombre entier	Non accessible	Nombre entier utilisé pour indexer un coeur dans cette table.
`ldomCoreID`	Chaîne d'affichage	Lecture seule	Identificateur d'un coeur (ID coeur).
`ldomCoreCpuSet`	Chaîne d'affichage	Lecture seule	Liste des CPU mappées à `MAU-unit cpuset` du coeur.

Variables scalaires des informations de version de Logical Domains

Le protocole Logical Domains Manager prend en charge les versions de Logical Domains, qui sont composées d'un numéro majeur et d'un numéro mineur. Oracle VM Server for SPARC MIB dispose de variables scalaires pour décrire les informations de version de Logical Domains.

TABLEAU 15–23 Variables scalaires des informations de version de Logical Domains

Nom	Type de données	Accès	Description
`ldomVersionMajor`	Nombre entier	Lecture seule	Numéro de version majeure.
`ldomVersionMinor`	Nombre entier	Lecture seule	Numéro de version mineure.

Les valeurs pour `ldomVersionMajor` et `ldomVersionMinor` sont équivalentes à la version affichée par la commande `ldm list -p`. Par exemple :

```
$ ldm ls -p
VERSION 1.6
...

$ snmpget -v1 -c public localhost SUN-LDOM-MIB::ldomVersionMajor.0
SUN-LDOM-MIB::ldomVersionMajor.0 = INTEGER: 1

$ snmpget -v1 -c public localhost SUN-LDOM-MIB::ldomVersionMinor.0
SUN-LDOM-MIB::ldomVersionMinor.0 = INTEGER: 5
```

Utilisation des déroutements SNMP

Cette section décrit la configuration de votre système afin d'envoyer et de recevoir des déroutements. Elle décrit également les déroutements que vous pouvez utiliser pour recevoir des notifications de modifications pour les domaines logiques (domaines), ainsi que d'autres déroutements que vous êtes susceptible d'utiliser.

Utilisation des déroutements du module Oracle VM Server for SPARC MIB

Envoi et réception de déroutements

▼ Procédure d'envoi de déroutements

- **Configurez le déroutement.**

  Editez le fichier `/etc/sma/snmp/snmpd.conf` afin d'ajouter les directives qui définissent le déroutement, la version et la destination.

  ```
  trapcommunity string --> define community string to be used when sending traps
  trapsink host[community [port]] --> to send v1 traps
  trap2sink host[community [port]] --> to send v2c traps
  informsink host[community [port]] --> to send informs
  ```

  Pour plus d'informations, reportez-vous à la page de manuel `snmpd.conf`(4).

Exemple 15–4 Envoi de déroutements SNMP v1 et v2c

Cet exemple envoie des déroutements `v1` et `v2c` au démon de déroutement SNMP qui s'exécute sur le même hôte. Le fichier `/etc/sma/snmp/snmpd.conf` est mis à jour avec les directives suivantes :

```
trapcommunity public
trapsink localhost
trap2sink localhost
```

▼ Procédure de réception de déroutements

- **Lancez l'utilitaire du démon de déroutement SNMP.**

  Pour plus d'informations sur les options du format de sortie, reportez-vous à la page de manuel `snmptrapd`(1M).

  L'utilitaire `snmptrapd` est une application SNMP qui reçoit et consigne les messages `TRAP` SNMP. Par exemple, la commande `snmptrapd` suivante indique qu'un nouveau domaine a été créé (`ldomTrapDesc = Ldom Created`) avec le nom `ldg2` (`ldomName = ldg2`).

```
# /usr/sfw/sbin/snmptrapd -P -F \
"TRAP from %B on %m/%l/%y at %h:%j:%k Enterprise=%N Type=%w SubType=%q \nwith Varbinds: %v \nSecurity info:%P\n\n" \
localhost:162
TRAP from localhost on 5/18/2007 at 16:30:10 Enterprise=. Type=0 SubType=0
with Varbinds: DISMAN-EVENT-MIB::sysUpTimeInstance = Timeticks: (47105)
0:07:51.05 SNMPv2-MIB::snmpTrapOID.0 = OID: SUN-LDOM-MIB::ldomCreate
SUN-LDOM-MIB::ldomIndexNotif = INTEGER: 3 SUN-LDOM-MIB::ldomName = STRING: ldg2
SUN-LDOM-MIB::ldomTrapDesc = STRING: Ldom Created
Security info:TRAP2, SNMP v2c, community public
```

Description des déroutements Oracle VM Server for SPARC MIB

Cette section décrit les déroutements Oracle VM Server for SPARC MIB que vous pouvez utiliser.

Création de domaine (ldomCreate)

Ce déroutement vous notifie lorsque des domaines sont créés.

TABLEAU 15–24 Déroutement de création de domaine (ldomCreate)

Nom	Type de données	Description
`ldomIndexNotif`	Nombre entier	Index dans `ldomTable`
`ldomName`	Chaîne d'affichage	Nom du domaine
`ldomTrapDesc`	Chaîne d'affichage	Description du déroutement.

Destruction de domaine (ldomDestroy)

Ce déroutement vous notifie lorsque des domaines sont détruits.

TABLEAU 15–25 Déroutement de destruction de domaine (ldomDestroy)

Nom	Type de données	Description
`ldomIndexNotif`	Nombre entier	Index dans `ldomTable`
`ldomName`	Chaîne d'affichage	Nom du domaine
`ldomTrapDesc`	Chaîne d'affichage	Description du déroutement.

Modification de l'état du domaine (ldomStateChange)

Ce déroutement vous notifie en cas de modification de l'état de fonctionnement d'un domaine.

TABLEAU 15–26 Déroutement de modification de l'état du domaine (ldomStateChange)

Nom	Type de données	Description
`ldomIndexNotif`	Nombre entier	Index dans `ldomTable`
`ldomName`	Chaîne d'affichage	Nom du domaine
`ldomOperState`	Nombre entier	Nouvel état du domaine.
`ldomStatePrev`	Nombre entier	Ancien état du domaine.
`ldomTrapDesc`	Chaîne d'affichage	Description du déroutement.

Modification de la CPU virtuelle (ldomVCpuChange)

Ce déroutement vous notifie lorsque le nombre de CPU virtuelles d'un domaine change.

TABLEAU 15–27 Déroutement de modification de la CPU virtuelle (ldomVCpuChange)

Nom	Type de données	Description
`ldomIndexNotif`	Nombre entier	Index dans `ldomTable`
`ldomName`	Chaîne d'affichage	Nom du domaine contenant la CPU virtuelle.
`ldomNumVCPU`	Nombre entier	Nouveau nombre de CPU virtuelles pour le domaine.
`ldomNumVCPUPrev`	Nombre entier	Ancien nombre de CPU virtuelles pour le domaine.
`ldomTrapDesc`	Chaîne d'affichage	Description du déroutement.

Modification de la mémoire virtuelle (ldomVMemChange)

Ce déroutement vous notifie lorsque la quantité de mémoire virtuelle d'un domaine change.

TABLEAU 15–28 Déroutement de modification de la mémoire virtuelle (ldomVMemChange)

Nom	Type de données	Description
`ldomIndexNotif`	Nombre entier	Index dans `ldomTable`
`ldomName`	Chaîne d'affichage	Nom du domaine contenant la mémoire virtuelle.
`ldomMemSize`	Nombre entier	Quantité de mémoire virtuelle dans le domaine.
`ldomMemSizePrev`	Nombre entier	Quantité antérieure de mémoire virtuelle dans le domaine.
`ldomMemUnit`	Nombre entier	Unité de mémoire pour la mémoire virtuelle, parmi les suivantes : ■ 1 = Ko ■ 2 = Mo ■ 3 = Go ■ 4 = octets Si cela n'est pas spécifié, la valeur de l'unité est octets.

TABLEAU 15–28 Déroutement de modification de la mémoire virtuelle (ldomVMemChange) *(Suite)*

Nom	Type de données	Description
`ldomMemUnitPrev`	Nombre entier	Unité de mémoire pour la mémoire virtuelle antérieure, parmi les suivantes : ■ 1 = Ko ■ 2 = Mo ■ 3 = Go ■ 4 = octets Si cela n'est pas spécifié, la valeur de l'unité est octets.
`ldomTrapDesc`	Chaîne d'affichage	Description du déroutement.

Modification du service de disques virtuels (ldomVdsChange)

Ce déroutement vous notifie en cas de modification du service de disques virtuels.

TABLEAU 15–29 Déroutement de modification du service de disques virtuels d'un domaine (ldomVdsChange)

Nom	Type de données	Description
`ldomIndexNotif`	Nombre entier	Index dans `ldomTable`
`ldomName`	Chaîne d'affichage	Nom du domaine contenant le service de disques virtuels.
`ldomVdsServiceName`	Chaîne d'affichage	Nom du service de disques virtuels modifié.
`ldomChangeFlag`	Nombre entier	Indique que l'une des modifications suivantes s'est produite sur le service de disques virtuels : ■ 1 = Ajout ■ 2 = Modification ■ 3 = Suppression
`ldomTrapDesc`	Chaîne d'affichage	Description du déroutement.

Modification du disque virtuel (ldomVdiskChange)

Ce déroutement vous notifie en cas de modification du disque virtuel d'un domaine.

TABLEAU 15–30 Déroutement de modification du disque virtuel (ldomVdiskChange)

Nom	Type de données	Description
`ldomIndexNotif`	Nombre entier	Index dans `ldomTable`

TABLEAU 15–30 Déroutement de modification du disque virtuel (ldomVdiskChange) *(Suite)*

Nom	Type de données	Description
`ldomName`	Chaîne d'affichage	Nom du domaine contenant le périphérique de disques virtuels.
`ldomVdiskName`	Chaîne d'affichage	Nom du périphérique de disques virtuels modifié.
`ldomChangeFlag`	Nombre entier	Indique que l'une des modifications suivantes s'est produite sur le service de disques virtuels : ■ 1 = Ajout ■ 2 = Modification ■ 3 = Suppression
`ldomTrapDesc`	Chaîne d'affichage	Description du déroutement.

Modification du commutateur virtuel (ldomVswChange)

Ce déroutement vous notifie en cas de modification du commutateur virtuel d'un domaine.

TABLEAU 15–31 Déroutement de modification du commutateur virtuel (ldomVswChange)

Nom	Type de données	Description
`ldomIndexNotif`	Nombre entier	Index dans `ldomTable`
`ldomName`	Chaîne d'affichage	Nom du domaine contenant le service de commutateurs virtuels.
`ldomVswServiceName`	Chaîne d'affichage	Nom du service de commutateurs virtuels modifié.
`ldomChangeFlag`	Nombre entier	Indique que l'une des modifications suivantes s'est produite sur le service de commutateurs virtuels : ■ 1 = Ajout ■ 2 = Modification ■ 3 = Suppression
`ldomTrapDesc`	Chaîne d'affichage	Description du déroutement.

Modification de réseau virtuel (ldomVnetChange)

Ce déroutement vous notifie en cas de modification du réseau virtuel d'un domaine.

TABLEAU 15–32 Déroutement de modification de réseau virtuel (ldomVnetChange)

Nom	Type de données	Description
`ldomIndexNotif`	Nombre entier	Index dans `ldomTable`
`ldomName`	Chaîne d'affichage	Nom du domaine contenant le périphérique réseau virtuel.
`ldomVnetDevName`	Chaîne d'affichage	Nom du périphérique réseau virtuel du domaine.
`ldomChangeFlag`	Nombre entier	Indique que l'une des modifications suivantes s'est produite sur le service de disques virtuels : ■ 1 = Ajout ■ 2 = Modification ■ 3 = Suppression
`ldomTrapDesc`	Chaîne d'affichage	Description du déroutement.

Modification du concentrateur de consoles virtuelles (ldomVccChange)

Ce déroutement vous notifie en cas de modification du concentrateur de consoles virtuelles d'un domaine.

TABLEAU 15–33 Déroutement de modification du concentrateur de consoles virtuelles (ldomVccChange)

Nom	Type de données	Description
`ldomIndexNotif`	Nombre entier	Index dans `ldomTable`
`ldomName`	Chaîne d'affichage	Nom du domaine contenant le concentrateur de consoles virtuelles.
`ldomVccName`	Chaîne d'affichage	Nom du service de concentrateurs de consoles virtuelles modifié.
`ldomChangeFlag`	Nombre entier	Indique que l'une des modifications suivantes s'est produite sur le concentrateur de consoles virtuelles : ■ 1 = Ajout ■ 2 = Modification ■ 3 = Suppression
`ldomTrapDesc`	Chaîne d'affichage	Description du déroutement.

Modification du groupe de consoles virtuelles (ldomVconsChange)

Ce déroutement vous notifie en cas de modification du groupe de consoles virtuelles d'un domaine.

TABLEAU 15–34 Déroutement de modification du groupe de consoles virtuelles (ldomVconsChange)

Nom	Type de données	Description
`ldomIndexNotif`	Nombre entier	Index dans `ldomTable`
`ldomName`	Chaîne d'affichage	Nom du domaine contenant le groupe de consoles virtuelles.
`ldomVconsGroupName`	Chaîne d'affichage	Nom du groupe de consoles virtuelles modifié.
`ldomChangeFlag`	Nombre entier	Indique que l'une des modifications suivantes s'est produite sur le groupe de consoles virtuelles : ■ 1 = Ajout ■ 2 = Modification ■ 3 = Suppression
`ldomTrapDesc`	Chaîne d'affichage	Description du déroutement.

Démarrage et arrêt des domaines

Cette section décrit les opérations de gestion active utilisées pour arrêter et démarrer des domaines. Vous pouvez les utiliser en définissant la valeur de la propriété `ldomAdminState` dans la table de domaines, `ldomTable`. Reportez-vous au Tableau 15–1.

Démarrage et arrêt d'un domaine

▼ Procédure de démarrage d'un domaine

Cette procédure décrit la procédure de démarrage d'un domaine lié. Si un domaine portant le nom du domaine spécifié n'existe pas ou est déjà lié, cette opération échouera.

1 Vérifiez que le domaine *domain-name* existe et est lié.

```
# ldm list domain-name
```

2 Identifiez *domain-name* dans `ldomTable`.

```
# snmpwalk -v1 -c public localhost SUN-LDOM-MIB::ldomTable
SUN-LDOM-MIB::ldomName.1 = STRING: primary
SUN-LDOM-MIB::ldomName.2 = STRING: LdomMibTest_1
SUN-LDOM-MIB::ldomAdminState.1 = INTEGER: 0
SUN-LDOM-MIB::ldomAdminState.2 = INTEGER: 0
SUN-LDOM-MIB::ldomOperState.1 = INTEGER: active(1)
SUN-LDOM-MIB::ldomOperState.2 = INTEGER: bound(6)
SUN-LDOM-MIB::ldomNumVCpu.1 = INTEGER: 32
SUN-LDOM-MIB::ldomNumVCpu.2 = INTEGER: 2
```

```
SUN-LDOM-MIB::ldomMemSize.1 = INTEGER: 3968
SUN-LDOM-MIB::ldomMemSize.2 = INTEGER: 256
SUN-LDOM-MIB::ldomMemUnit.1 = INTEGER: megabytes(2)
SUN-LDOM-MIB::ldomMemUnit.2 = INTEGER: megabytes(2)
SUN-LDOM-MIB::ldomNumCrypto.1 = INTEGER: 8
SUN-LDOM-MIB::ldomNumCrypto.2 = INTEGER: 0
SUN-LDOM-MIB::ldomNumIOBus.1 = INTEGER: 2
SUN-LDOM-MIB::ldomNumIOBus.2 = INTEGER: 0
SUN-LDOM-MIB::ldomUUID.1 = STRING: c2c3d93b-a3f9-60f6-a45e-f35d55c05fb6
SUN-LDOM-MIB::ldomUUID.2 = STRING: af0b05f0-d262-e633-af32-a6c4e81fb81c
SUN-LDOM-MIB::ldomMacAddress.1 = STRING: 00:14:4f:86:63:2a
SUN-LDOM-MIB::ldomMacAddress.2 = STRING: 00:14:4f:fa:78:b9
SUN-LDOM-MIB::ldomHostID.1 = STRING: 0x8486632a
SUN-LDOM-MIB::ldomHostID.2 = STRING: 0x84fa78b9
SUN-LDOM-MIB::ldomFailurePolicy.1 = STRING: ignore
SUN-LDOM-MIB::ldomFailurePolicy.2 = STRING: ignore
SUN-LDOM-MIB::ldomMaster.1 = STRING:
SUN-LDOM-MIB::ldomMaster.2 = STRING:
```

3 **Démarrez le domaine** *domain-name***.**

Utilisez la commande `snmpset` pour démarrer le domaine en définissant la valeur de la propriété `ldomAdminState` sur `1`. *n* spécifie le domain à démarrer.

```
# snmpset -v version -c community-string hostname \
SUN-LDOM-MIB::ldomTable.1.ldomAdminState.n = 1
```

4 **Vérifiez que le domaine** *domain-name* **est actif.**

- **Utilisez la commande `ldm list`.**

  ```
  # ldm list domain-name
  ```

- **Utilisez la commande `snmpget`.**

  ```
  # snmpget -v version -c community-string hostname SUN-LDOM-MIB::ldomOperState.n
  ```

Exemple 15–5 Démarrage d'un domaine invité

Cet exemple vérifie que le domaine `LdomMibTest_1` existe et est lié avant de définir la propriété `ldomAdminState` sur `1`. Enfin, la commande `ldm list LdomMibTest_1` vérifie que le domaine `LdomMibTest_1` est actif.

```
# ldm list LdomMibTest_1
# snmpset -v1 -c private localhost SUN-LDOM-MIB::ldomTable.1.ldomAdminState.2 = 1
# ldm list LdomMibTest_1
```

Au lieu d'utiliser la commande `ldm list` pour récupérer l'état du domaine `LdomMibTest_1`, vous pouvez utiliser la commande `snmpget`.

```
# snmpget -v1 -c public localhost SUN-LDOM-MIB::ldomOperState.2
```

Notez que si le domaine est inactif lorsque vous utilisez la commande `snmpset` pour démarrer le domaine, celui-ci est d'abord lié, puis démarré.

▼ Procédure d'arrêt d'un domaine

Cette procédure décrit la procédure d'arrêt d'un domaine démarré. Toutes les instances du système d'exploitation hébergées par le domaine seront stoppées.

1 Identifiez *domain-name* dans `ldomTable`.

```
# snmpwalk -v1 -c public localhost SUN-LDOM-MIB::ldomTable
SUN-LDOM-MIB::ldomName.1 = STRING: primary
SUN-LDOM-MIB::ldomName.2 = STRING: LdomMibTest_1
SUN-LDOM-MIB::ldomAdminState.1 = INTEGER: 0
SUN-LDOM-MIB::ldomAdminState.2 = INTEGER: 0
SUN-LDOM-MIB::ldomOperState.1 = INTEGER: active(1)
SUN-LDOM-MIB::ldomOperState.2 = INTEGER: bound(6)
SUN-LDOM-MIB::ldomNumVCpu.1 = INTEGER: 32
SUN-LDOM-MIB::ldomNumVCpu.2 = INTEGER: 2
SUN-LDOM-MIB::ldomMemSize.1 = INTEGER: 3968
SUN-LDOM-MIB::ldomMemSize.2 = INTEGER: 256
SUN-LDOM-MIB::ldomMemUnit.1 = INTEGER: megabytes(2)
SUN-LDOM-MIB::ldomMemUnit.2 = INTEGER: megabytes(2)
SUN-LDOM-MIB::ldomNumCrypto.1 = INTEGER: 8
SUN-LDOM-MIB::ldomNumCrypto.2 = INTEGER: 0
SUN-LDOM-MIB::ldomNumIOBus.1 = INTEGER: 2
SUN-LDOM-MIB::ldomNumIOBus.2 = INTEGER: 0
SUN-LDOM-MIB::ldomUUID.1 = STRING: c2c3d93b-a3f9-60f6-a45e-f35d55c05fb6
SUN-LDOM-MIB::ldomUUID.2 = STRING: af0b05f0-d262-e633-af32-a6c4e81fb81c
SUN-LDOM-MIB::ldomMacAddress.1 = STRING: 00:14:4f:86:63:2a
SUN-LDOM-MIB::ldomMacAddress.2 = STRING: 00:14:4f:fa:78:b9
SUN-LDOM-MIB::ldomHostID.1 = STRING: 0x8486632a
SUN-LDOM-MIB::ldomHostID.2 = STRING: 0x84fa78b9
SUN-LDOM-MIB::ldomFailurePolicy.1 = STRING: ignore
SUN-LDOM-MIB::ldomFailurePolicy.2 = STRING: ignore
SUN-LDOM-MIB::ldomMaster.1 = STRING:
SUN-LDOM-MIB::ldomMaster.2 = STRING:
```

2 Arrêtez le domaine *domain-name*.

Utilisez la commande `snmpset` pour arrêter le domaine en définissant la valeur de la propriété `ldomAdminState` sur 2. *n* spécifie le domaine à arrêter.

```
# snmpset -v version -c community-string hostname \
SUN-LDOM-MIB::ldomTable.1.ldomAdminState.n = 2
```

3 Vérifiez que le domaine *domain-name* est lié.

- **Utilisez la commande `ldm list`.**

  ```
  # ldm list domain-name
  ```

- **Utilisez la commande `snmpget`.**

  ```
  # snmpget -v version -c community-string hostname SUN-LDOM-MIB::ldomOperState.n
  ```

Exemple 15–6 Arrêt d'un domaine invité

Cet exemple définit la propriété `ldomAdminState` sur 2 afin d'arrêter le domaine invité, puis utilise la commande `ldm list LdomMibTest_1` pour vérifier si le domaine `LdomMibTest_1` est lié.

```
# snmpset -v1 -c private localhost SUN-LDOM-MIB::ldomTable.1.ldomAdminState.2 = 2
# ldm list LdomMibTest_1
```

◆ ◆ ◆ CHAPITRE 16

16 Recherche de Logical Domains Manager

Logical Domains Manager peut être recherché sur un sous-réseau à l'aide des messages de multidiffusion. Le démon `ldmd` peut écouter sur un réseau pour recherche un paquet de multidiffusion spécifique. Si ce message de multidiffusion est d'un certain type, `ldmd` répond au programme appelant. Cela permet à `ldmd` d'être recherchés sur les systèmes exécutant Oracle VM Server for SPARC.

Ce chapitre contient des informations sur la détection de Logical Domains Manager exécuté sur des systèmes d'un sous-réseau.

Recherche des systèmes exécutant Logical Domains Manager

Communication en multidiffusion

Ce mécanisme de découverte utilise le même réseau de multidiffusion que celui utilisée par le démon `ldmd` pour détecter les collisions lors de l'adressage automatique des adresses MAC. Pour configurer le socket de multidiffusion, vous devez fournir les informations suivantes :

```
#define    MAC_MULTI_PORT        64535
#define    MAC_MULTI_GROUP        "239.129.9.27"
```

Par défaut, *seuls* les paquets de multidiffusion peuvent être envoyés sur le sous-réseau auquel la machine est connectée. Vous pouvez modifier ce comportement en définissant la propriété SMF `ldmd/hops` pour le démon `ldmd`.

Format du message

Les messages de recherche doivent être clairement identifiés afin de ne pas être confondus avec d'autres messages. Le format de message de multidiffusion suivant garantit que les messages de recherche peuvent être distingués par le processus d'écoute de recherche :

```
#include <netdb.h> /* Used for MAXHOSTNAMELEN definition */
#define    MAC_MULTI_MAGIC_NO    92792004
#define    MAC_MULTI_VERSION    1

enum {
    SEND_MSG = 0,
    RESPONSE_MSG,
    LDMD_DISC_SEND,
    LDMD_DISC_RESP,
};

typedef struct {
    uint32_t    version_no;
    uint32_t    magic_no;
    uint32_t    msg_type;
    uint32_t    resv;
    union {
        mac_lookup_t        Mac_lookup;
        ldmd_discovery_t    Ldmd_discovery;
    } payload;
#define    lookup        payload.Mac_lookup
#define    discovery    payload.Ldmd_discovery
} multicast_msg_t;

#define    LDMD_VERSION_LEN    32

typedef struct {
    uint64_t mac_addr;
    char        source_ip[INET_ADDRSTRLEN];
} mac_lookup_t;

typedef struct {
    char        ldmd_version[LDMD_VERSION_LEN];
    char        hostname[MAXHOSTNAMELEN];
    struct in_addr    ip_address;
    int        port_no;
} ldmd_discovery_t;
```

▼ Procédure de découverte d'instances de Logical Domains Manager s'exécutant sur votre sous-réseau

1 Ouvrez un socket de multidiffusion.

Vérifiez que vous utilisez le port et les informations de groupe indiquées dans "Communication en multidiffusion" à la page 339.

2 Envoyez un message `multicast_msg_t` sur le socket.

Le message doit inclure les éléments suivants :

- Une valeur valide pour `version_no`, qui est `1` comme défini par `MAC_MULTI_VERSION`
- Une valeur valide pour `magic_no`, qui est `92792004` comme défini par `MAC_MULTI_MAGIC_NO`
- `msg_type` de `LDMD_DISC_SEND`

3 Ecoutez sur le socket de multidiffusion les réponses de Logical Domains Manager.

Les réponses doivent être un message `multicast_msg_t` avec les éléments suivants :

- Valeur valide pour `version_no`
- Valeur valide pour `magic_no`
- `msg_type` défini sur `LDMD_DISC_RESP`
- Charge utile composée d'une structure `ldmd_discovery_t`, qui contient les informations suivantes :
  - `ldmd_version` - Version de Logical Domains Manager s'exécutant sur le système
  - `hostname` – Nom d'hôte du système
  - `ip_address` – Adresse IP du système
  - `port_no` – Numéro de port utilisé par Logical Domains Manager pour les communications, qui doit être le port XMPP `6482`

Lors de l'écoute d'une réponse de Logical Domains Manager, vérifiez que tous les paquets de détection-collision MAC d'allocation automatique sont ignorés.

◆ ◆ ◆ CHAPITRE 17

Utilisation de l'interface XML avec Logical Domains Manager

Ce chapitre explique le mécanisme de communication Extensible Markup Language (XML) par lequel les programmes des utilisateurs externes peuvent communiquer avec le logiciel Oracle VM Server for SPARC. Les sujets de base suivants sont abordés :

Transport XML

Les programmes externes peuvent utiliser le Extensible Messaging and Presence Protocol (XMPP - RFC 3920) pour communiquer avec Logical Domains Manager. Le protocole XMPP est pris en charge par les connexions locales et distantes et est activé par défaut. Pour interrompre une connexion à distance, définissez la propriété SMF `ldmd/xmpp_enabled` sur `false` et redémarrez Logical Domains Manager.

```
# svccfg -s ldom/ldmd setprop ldmd/xmpp_enabled=false
# svcadm refresh ldmd
# svcadm restart ldmd
```

Remarque – La désactivation du serveur XMPP empêche également la migration du domaine et la reconfiguration dynamique de la mémoire.

Serveur XMPP

Logical Domains Manager implémente un serveur XMPP qui peut communiquer avec de nombreuses applications et bibliothèques client XMPP disponibles. Logical Domains Manager utilise les mécanismes de sécurité suivants :

- TLS (Transport Layer Security - Sécurité de la couche de transport) pour sécuriser le canal de communication entre le client et lui-même.
- SASL (Simple Authentification and Security Layer - Authentification simple et couche de sécurité) pour l'authentification. `PLAIN` est le seul mécanisme SASL pris en charge. Vous devez envoyer un nom d'utilisateur et un mot de passe au serveur afin qu'il puisse vous accorder les droits avant d'autoriser des opérations de surveillance ou de gestion.

Connexions locales

Logical Domains Manager détecte si les clients utilisateur sont en cours d'exécution sur le même domaine que lui-même et, le cas échéant, effectue une communication XMPP minimale avec ce client. Plus précisément, l'étape d'authentification SASL après la configuration d'un canal sécurisé par TLS est ignorée. L'authentification et l'autorisation sont effectuées en fonction des données d'identification du processus implémentant l'interface client.

Les clients peuvent choisir d'implémenter un client totalement XMPP ou d'exécuter simplement un analyseur syntaxique XML de transmission en continu, tel que l'analyseur syntaxique d'API simple `libxml2` pour XML (SAX). Quelle que soit la méthode, le client doit traiter une communication XMPP au moment de la communication TLS. Reportez-vous à la spécification XMPP pour la séquence nécessaire.

Protocole XML

A la fin de l'initialisation de la communication, les messages XML définis par Logical Domains sont envoyés. Il a deux types généraux de messages XML :

- Les messages de demande et de réponse utilisent la balise `<LDM_interface>`. Ce type de message XML est utilisé pour communiquer les commandes et obtenir les résultats de Logical Domains Manager, comme lors de l'exécution des commandes utilisant l'interface de ligne de commande (CLI). Cette balise est également utilisée pour l'enregistrement et l'annulation de l'enregistrement des événements.
- Les messages d'événement utilisent la balise `<LDM_event>`. Ce type de message XML est utilisé pour signaler de manière asynchrone les événements postés par Logical Domains Manager.

Messages de demande et de réponse

L'interface XML de Logical Domains a deux formats différents :

- Un format pour envoyer des commandes dans Logical Domains Manager
- Un autre format pour que Logical Domains Manager réponde sur l'état du message entrant et les actions demandées dans le message.

Ces deux formats partagent de nombreuses structures XML communes, mais sont séparés dans cette présentation pour une meilleure compréhension de leurs différences.

Messages de demande

Une demande XML entrante vers Logical Domains Manager à son niveau le plus basique comprend une description d'une seule commande, s'appliquant à un seul objet. Les demandes plus complexes peuvent traiter plusieurs commandes et plusieurs objets par commande. Vous trouverez ci-dessous la structure d'une commande XML basique.

EXEMPLE 17–1 Format d'une commande unique s'appliquant sur un seul objet

```
<LDM_interface version="1.3">
  <cmd>
    <action>Place command here</action>
    <option>Place options for certain commands here</option>
    <data version="3.0">
      <Envelope>
        <References/>
        <!-- Note a <Section> section can be here instead of <Content> -->
        <Content xsi:type="ovf:VirtualSystem_Type" id="Domain name">
          <Section xsi:type="ovf:ResourceAllocationSection_type">
            <Item>
              <rasd:OtherResourceType>LDom Resource Type</rasd:OtherResourceType>
              <gprop:GenericProperty
              key="Property name">Property Value</gprop:GenericProperty>
            </Item>
          </Section>
          <!-- Note: More Sections sections can be placed here -->
        </Content>
      </Envelope>
    </data>
    <!-- Note: More Data sections can be placed here -->
  </cmd>
  <!-- Note: More Commands sections can be placed here -->
</LDM_interface>
```

Balise <LDM_interface>

Toutes les commandes envoyées au Logical Domains Manager doivent commencer par la balise `<LDM_interface>`. Tout document envoyé au Logical Domains Manager doit comporter une balise unique `<LDM_interface>` contenue dedans. La balise `<LDM_interface>` doit comprendre un attribut de version comme indiqué dans l'Exemple 17–1.

Balise <cmd>

Dans la balise <LDM_interface>, le document doit inclure au moins une balise <cmd>. Chaque section <cmd> doit comporter uniquement une seule balise <action>. Utilisez la balise <action> pour décrire la commande à exécuter. Chaque balise <cmd> doit inclure au moins une balise <data> pour décrire les objets sur lesquels la commande doit être appliquée.

La balise <cmd> peut également avoir une balise <option>, qui est utilisée pour les options et les indicateurs associés à certaines commandes. Les commandes suivantes utilisent des options :

- La commande `remove-domain` peut utiliser l'option `-a`.
- La commande `stop-domain` peut utiliser l'option `-f`.
- La commande `cancel-operation` peut utiliser l'option `migration` ou `reconf`.
- La commande `add-spconfig` peut utiliser l'option `-r` *autosave-name*.
- La commande `remove-spconfig` peut utiliser l'option `-r`.
- La commande `list-spconfig` peut utiliser l'option `-r` [*autosave-name*].

Balise <data>

Chaque section <data> contient une description d'un objet correspondant à la commande indiquée. Le format de la section de données repose sur la portion de schéma XML de la spécification brouillon OVF (Open Virtualization Format). Le schéma définit une section <Envelope> qui contient une balise <References> (non utilisée par Logical Domains) et des sections <Content> et <Section>.

Pour Logical Domains, la section <Content> est utilisée pour identifier et décrire un domaine particulier. Le nom de domaine dans l'attribut id= du noeud <content> identifie le domaine. Dans la section <Content>, il ya une ou plusieurs sections <Section> décrivant les ressources du domaine comme requis par la commande spécifique.

Si vous devez uniquement identifier un nom de domaine, il est alors inutile d'utiliser des balises <Section>. Au contraire de cela, si aucun identifiant de domaine n'est nécessaire pour la commande, vous devez alors fournir une section <Section>, décrivant les ressources nécessaires à la commande, en dehors de la section <Content>, mais toujours dans la section <Envelope>.

Une section <data> ne doit pas contenir une balise <Envelope> si les informations de l'objet peuvent être induites. Cette situation s'applique principalement aux demandes de surveillance de tous les objets applicables à une action, à l'enregistrement des événements et aux demandes d'annulation d'enregistrement.

Pour permettre l'utilisation du schéma de spécification OVF afin de définir correctement tous les types d'objets, deux types OVF supplémentaires ont été définis :

- Balise <gprop:GenericProperty>
- Balise <Binding>

La balise <gprop:GenericProperty> a été définie pour traiter la propriété de n'importe quel objet pour lequel la spécification OVF n'a pas de définition. Le nom de la propriété est défini dans l'attribut key= du noeud et la valeur de la propriété est le contenu du noeud. La balise <binding> est utilisée dans la sous-commande list-bindings pour définir les ressources liées à d'autres ressources.

Messages de réponse

Une réponse XML sortante correspond étroitement à la structure de la demande entrante en termes de commandes et d'objets inclus, en ajoutant une section <Response> pour chaque objet et commande indiqué, ainsi qu'une section globale <Response> pour la demande. Les sections <Response> fournissent des informations d'état et des messages d'informations, comme décrit dans l'Exemple 17–2. Vous trouverez ci-dessous la structure d'une réponse à une demande XML basique.

EXEMPLE 17–2 Format d'une réponse à une commande unique s'appliquant sur un seul objet

```
<LDM_interface version="1.3">
  <cmd>
    <action>Place command here</action>
    <data version="3.0">
      <Envelope>
        <References/>
        <!-- Note a <Section> section can be here instead of <Content> -->
        <Content xsi:type="ovf:VirtualSystem_Type" id="Domain name">
          <Section xsi:type="ovf:ResourceAllocationSection_type">
            <Item>
              <rasd:OtherResourceType>
                LDom Resource Type
              </rasd:OtherResourceType>
              <gprop:GenericProperty
              key="Property name">
                Property Value
            </gprop:GenericProperty>
            </Item>
          </Section>
          <!-- Note: More <Section> sections can be placed here -->
        </Content>
      </Envelope>
      <response>
        <status>success or failure</status>
        <resp_msg>Reason for failure</resp_msg>
      </response>
    </data>
    <!-- Note: More Data sections can be placed here -->
    <response>
      <status>success or failure</status>
      <resp_msg>Reason for failure</resp_msg>
    </response>
  </cmd>
  <!-- Note: More Command sections can be placed here -->
  <response>
    <status>success or failure</status>
    <resp_msg>Reason for failure</resp_msg>
```

EXEMPLE 17–2 Format d'une réponse à une commande unique s'appliquant sur un seul objet *(Suite)*

```
  </response>
</LDM_interface>
```

Réponse globale

Cette section <response>, qui est l'enfant direct de la section <LDM_interface>, indique le succès ou l'échec global de toute la demande. A moins que le document XML entrant ne sont pas correctement formé, la section <response> comprend uniquement une balise <status>. Si l'état de la réponse indique un succès, toutes les commandes sur tous les objets ont abouti. Si cette réponse est un échec et qu'il n'y a pas de balise <resp_msg>, une des commandes inclue dans la demande d'origine a échoué. La balise <resp_msg> est utilisée uniquement pour décrire certains problèmes avec le document XML lui-même.

Réponse de la commande

La section <response> sous la section <cmd> alerte l'utilisateur du succès ou de l'échec de cette commande en particulier. La balise <status> indique si cette commande a abouti ou échoué. Tout comme pour la réponse globale, si la commande échoue, la section <response> comprend uniquement une balise <resp_msg> si le contenu de la section <cmd> de la demande n'est pas correctement formé. Sinon, l'état d'échec signifie qu'un des objets sur lesquels la commande a été exécutée a provoqué une erreur.

Réponse de l'objet

Enfin, la section <data> d'une section <cmd> comporte également une section <response>. Cela indique si la commande en cours d'exécution sur cet objet en particulier aboutit ou échoue. Si l'état de la réponse est SUCCESS, il n'y a pas de balise <resp_msg> dans la section <response>. Si l'état est FAILURE, il y a une ou plusieurs balises <resp_msg> dans la zone <response>, en fonction des erreurs rencontrées lors de l'exécution de la commande sur cet objet. Les erreurs des objets peuvent provenir de problèmes détectés lors de l'exécution de la commande ou d'un objet mal formé ou inconnu.

En plus de la section <response>, la section <data> peut contenir d'autres informations. Ces informations sont dans le même format qu'une zone <data> entrante, décrivant l'objet qui a provoqué l'erreur. Reportez-vous à la section "Balise <data>" à la page 346. Ces informations supplémentaires sont surtout utiles dans les cas suivants :

- Lorsqu'une commande échoue sur une section <data> spécifique mais aboutit pour d'autres sections <data>
- Lorsqu'une section <data> est transmise dans une commande et échoue pour certains domaines, mais aboutit pour d'autres

Messages d'événements

Au lieu d'interroger, vous pouvez vous abonner pour recevoir des notifications sur les événements lorsque certains changements d'état surviennent. Il y a trois types d'événements auxquels vous pouvez vous abonner, individuellement ou collectivement. Reportez-vous à la section "Types d'événements" à la page 350 pour obtenir des détails complets.

Enregistrement et annulation de l'enregistrement

Utilisez un message `<LDM_interface>` pour enregistrer les événements. Reportez-vous à la section "Balise `<LDM_interface>`" à la page 345. Les détails de la balise d'action définissent le type d'événement pour lequel s'enregistrer ou annuler l'enregistrement et la section `<data>` est laissée vide.

EXEMPLE 17–3 Exemple de message de demande d'enregistrement d'événement

```
<LDM_interface version="1.3">
  <cmd>
    <action>reg-domain-events</action>
    <data version="3.0"/>
  </cmd>
</LDM_interface>
```

Logical Domains Manager répond avec un message de réponse `<LDM_interface>` indiquant si l'enregistrement ou l'annulation de l'enregistrement a abouti.

EXEMPLE 17–4 Exemple de message de réponse d'enregistrement d'événement

```
<LDM_interface version="1.3">
  <cmd>
    <action>reg-domain-events</action>
    <data version="3.0"/>
      <response>
        <status>success</status>
      </response>
    </data>
    <response>
      <status>success</status>
    </response>
  </cmd>
  <response>
    <status>success</status>
  </response>
</LDM_interface>
```

La chaîne d'action pour chaque type d'événement est répertoriée dans la sous-section des événements.

Messages <LDM_event>

Les messages d'événements ont le même format qu'un message <LDM_interface> entrant à l'exception que la balise de début pour le message est <LDM_event>. La balise d'action du message est l'action qui a été effectuée pour déclencher l'événement. La section de données du message décrit l'objet associé à l'événement : les détails dépendent du type d'événement survenu.

EXEMPLE 17–5 Exemple de notification <LDM_event>

```
<LDM_event version='1.1'>
  <cmd>
    <action>Event command here</action>
    <data version='3.0'>
      <Envelope
        <References/>
        <Content xsi:type='ovf:VirtualSystem_Type' ovf:id='ldg1'/>
          <Section xsi:type="ovf:ResourceAllocationSection_type">
            <Item>
              <rasd:OtherResourceType>LDom Resource Type</rasd:OtherResourceType>
              <gprop:GenericProperty
              key="Property name">Property Value</gprop:GenericProperty>
            </Item>
          </Section>
      </Envelope>
    </data>
  </cmd>
</LDM_event>
```

Types d'événements

Ci-dessous se trouvent les types d'événements auxquels vous pouvez vous abonner :

- Evénements du domaine
- Evénements matériels
- Evénements de progression
- Evénements de ressource

Tous les événements correspondent aux sous-commandes `ldm`.

Evénements du domaine

Les événements du domaine décrivent quelles actions peuvent être effectuées directement sur un domaine. La liste suivante répertorie les événements de domaine qui peuvent être spécifiés dans la balise <action> du message <LDM_event>.

- `add-domain`
- `bind-domain`
- `domain-reset`
- `migrate-domain`

- `panic-domain`
- `remove-domain`
- `start-domain`
- `stop-domain`
- `unbind-domain`

Ces événements contiennent toujours *uniquement* une balise `<Content>` dans la section des données OVF qui décrit sur quel domaine l'événement est survenu. Pour vous abonner aux événements de domaine, envoyez un message `<LDM_interface>` avec la balise `<action>` définie sur **`reg-domain-events`**. Le désabonnement à ces événements nécessite un message `<LDM_interface>` avec la balise d'action définie sur **`unreg-domain-events`**.

Evénements matériels

Les événements matériels concernent les modifications du matériel du système physique. Dans le cas du logiciel Oracle VM Server for SPARC, les seules modifications matérielles pouvant être effectuées sont celles sur le processeur de service (SP) lorsqu'un utilisateur ajoute, supprime ou définit une configuration de SP. Actuellement, les trois seuls événements de ce type sont :

- `add-spconfig`
- `set-spconfig`
- `remove-spconfig`

Les événements matériels contiennent toujours *uniquement* une balise `<Section>` dans la section des données OVF qui décrit la configuration SP sur laquelle l'événement se produit. Pour vous abonner à ces événements, envoyez un message `<LDM_interface>` avec la balise `<action>` définie sur **`reg-hardware-events`**. Le désabonnement à ces événements nécessite un message `<LDM_interface>` avec la balise `<action>` définie sur **`unreg-hardware-events`**.

Evénements de progression

Les événements de progression sont émis pour les commandes à exécution longue, notamment la migration de domaine. Ces événements signalent la quantité de progression ayant été effectuée au cours du cycle de vie de la commande. Actuellement, seul l'événement `migration-process` est signalé.

Les événements de progression contiennent toujours *uniquement* une balise `<Section>`dans la section des données OVF qui décrit la configuration SP affectée par l'événement. Pour vous abonner à ces événements, envoyez un message `<LDM_interface>` avec la balise `<action>` définie sur `reg-hardware-events`. Le désabonnement à ces événements nécessite un message `<LDM_interface>` avec la balise `<action>` définie sur `unreg-hardware-events`.

La section <data> d'un événement de progression est composée d'une section <content> qui décrit le domaine concerné. Cette section <content> utilise une balise ldom_info <Section> pour mettre à jour la progression. Les propriétés génériques suivantes sont affichées dans la section ldom_info :

- --progress - Pourcentage de progression effectué par la commande
- --status – Etat de la commande qui peut être ongoing, failed ou done
- --source – Machine qui signale la progression

Evénements de ressource

Les événements de ressource surviennent lorsque les ressources sont ajoutées, supprimées ou modifiées dans un domaine. La section de données de certains de ces événements contient la balise <Content> avec une balise <Section> donnant un nom de service dans la section des données OVF.

La liste suivante répertorie les événements qui peuvent être spécifiés dans la balise <action> du message <LDM_event>.

- add-vdiskserverdevice
- remove-vdiskserverdevice
- set-vdiskserverdevice
- remove-vdiskserver
- set-vconscon
- remove-vconscon
- set-vswitch
- remove-vswitch
- remove-vdpcs

Les événements de ressource suivants contiennent toujours *uniquement* une balise <Content> dans la section des données OVF qui décrit sur quel domaine l'événement est survenu.

- add-vcpu
- add-crypto
- add-memory
- add-io
- add-variable
- add-vconscon
- add-vdisk
- add-vdiskserver
- add-vnet
- add-vswitch
- add-vdpcs
- add-vdpcc
- set-vcpu
- set-crypto
- set-memory

- `set-variable`
- `set-vnet`
- `set-vconsole`
- `set-vdisk`
- `remove-vcpu`
- `remove-crypto`
- `remove-memory`
- `remove-io`
- `remove-variable`
- `remove-vdisk`
- `remove-vnet`
- `remove-vdpcc`

Pour vous abonner aux événements de ressource, envoyez un message `<LDM_interface>` avec la balise `<action>` définie sur **`reg-resource-events`**. Le désabonnement à ces événements nécessite un message `<LDM_interface>` avec la balise `<action>` définie sur **`unreg-resource-events`**.

Tous les événements

Vous pouvez également vous abonner pour écouter ces trois types d'événements sans vous abonner à chacun individuellement. Pour vous abonner à ces trois types d'événements en même temps, envoyez un message `<LDM_interface>` avec la balise `<action>` définie sur **`reg-all-events`**. Le désabonnement à ces événements nécessite un message `<LDM_interface>` avec la balise `<action>` définie sur **`unreg-all-events`**.

Actions de Logical Domains Manager

Les commandes indiquées dans la balise `<action>`, à l'exception des commandes `*-*-events`, correspondent à celles de l'interface de ligne de commande `ldm`. Pour plus de détails sur les sous-commandes `ldm`, reportez-vous à la page de manuel `ldm(1M)`.

Remarque – L'interface XML ne prend *pas* en charge l'accès ou la commande *aliases* prise en charge par la CLI de Logical Domains Manager.

Les chaînes prises en charge de la balise `<action>` sont les suivantes :

- `add-domain`
- `add-io`
- `add-mau`
- `add-memory`
- `add-spconfig`
- `add-variable`

- add-vconscon
- add-vcpu
- add-vdisk
- add-vdiskserver
- add-vdiskserverdevice
- add-vdpcc
- add-vdpcs
- add-vnet
- add-vswitch
- bind-domain
- cancel-operation
- list-bindings
- list-constraints
- list-devices
- list-domain
- list-services
- list-spconfig
- list-variable
- migrate-domain
- reg-all-events
- reg-domain-events
- reg-hardware-events
- reg-resource-events
- remove-domain
- remove-io
- remove-mau
- remove-memory
- remove-reconf
- remove-spconfig
- remove-variable
- remove-vconscon
- remove-vcpu
- remove-vdisk
- remove-vdiskserver
- remove-vdiskserverdevice
- remove-vdpcc
- remove-vdpcs
- remove-vnet
- remove-vswitch
- set-domain
- set-mau
- set-memory
- set-spconfig
- set-variable

- set-vconscon
- set-vconsole
- set-vcpu
- set-vnet
- set-vswitch
- start-domain
- stop-domain
- unbind-domain
- unreg-all-events
- unreg-domain-events
- unreg-hardware-events
- unreg-resource-events

Ressources et propriétés de Logical Domains Manager

Vous trouverez ci-dessous les ressources et les propriétés de Logical Domains Manager qui peuvent être définies pour chacune de ces ressources. Les ressources et les propriétés sont présentées en **gras** dans les exemples XML. Ces exemples présentent des ressources, non une sortie de liaison. La sortie de contrainte peut être utilisée pour créer une entrée pour les actions de Logical Domains Manager. L'exception à cela est la sortie de la migration de domaine. Reportez-vous à la section "Migration de domaine" à la page 367. Chaque ressource est définie dans une section OVF `<Section>` et est définie par une balise `<rasd:OtherResourceType>`.

Ressource d'informations sur le domaine (ldom_info)

EXEMPLE 17–6 Exemple de sortie XML ldom_info

```
<Envelope>
  <References/>
  <Content xsi:type="ovf:VirtualSystem_Type" id="primary">
    <Section xsi:type="ovf:ResourceAllocationSection_type">
      <Item>
        <rasd:OtherResourceType>ldom_info</rasd:OtherResourceType>
        <uuid>c2c3d93b-a3f9-60f6-a45e-f35d55c05fb6</uuid>
        <rasd:Address>00:03:ba:d8:ba:f6</rasd:Address>
        <gprop:GenericPropertykey="hostid">83d8baf6</gprop:GenericProperty>
        <gprop:GenericProperty key="master">plum</gprop:GenericProperty>
        <gprop:GenericProperty key="failure-policy">reset</gprop:GenericProperty>
        <gprop:GenericProperty key="extended-mapin-space">on</gprop:GenericProperty>
        <gprop:GenericProperty key="progress">45%</gprop:GenericProperty>
        <gprop:GenericProperty key="status">ongoing</gprop:GenericProperty>
        <gprop:GenericProperty key="source">dt90-319</gprop:GenericProperty>
      </Item>
    </Section>
  </Content>
</Envelope>
```

La ressource `ldom_info` est toujours contenue dans une section `<Content>`. Les propriétés suivantes de la ressource `ldom_info` sont des propriétés facultatives :

- Balise `<uuid>`, qui spécifie l'UUID du domaine.
- `<rasd:Address>`, qui définit l'adresse MAC à assigner au domaine.
- `<gprop:GenericPropertykey="extended-mapin-space">`, qui définit si la fonctionnalité Extended Mapin Space est activée (`on`) ou désactivée (`off`) pour le domaine. La valeur par défaut est `off`.
- `<gprop:GenericPropertykey="failure-policy">`, qui définit comment les domaines esclaves doivent se comporter, si le domaine maître est en panne. La valeur par défaut est `ignore`. Les valeurs valides de la propriétés sont indiquées ci-après :
  - `ignore` ignore les erreurs du domaine maître (les domaines esclaves ne sont pas affectés).
  - `panic` panique tous les domaines esclaves lorsque le domaine maître est en panne.
  - `reset` réinitialise tous les domaines esclaves lorsque le domaine maître est en panne.
  - `stop` arrête tous les domaines esclaves lorsque le domaine maître est en panne.
- `<gprop:GenericPropertykey="hostid">`, qui définit l'ID hôte à assigner au domaine.
- `<gprop:GenericPropertykey="master">`, qui définit jusqu'à quatre noms de domaine maître séparés par des virgules.
- `<gprop:GenericPropertykey="progress">`, qui définit le pourcentage de progression effectué par la commande.
- `<gprop:GenericPropertykey="source">`, qui définit le compte-rendu de la machine sur la progression de la commande.
- `<gprop:GenericPropertykey="status">`, qui définit l'état de la commande (`done`, `failed` ou `ongoing`).

Ressource de la CPU (cpu)

L'équivalent des actions de demande XML `add-vcpu`, `set-vcpu` et `remove-vcpu` est défini à la valeur de la balise `<gpropGenericProperty key="wcore">` comme suit :

- Si l'option `-c` est utilisée, définissez la propriété `wcore` sur le nombre du total de serveurs de base spécifié.
- Si l'option `-c` n'est *pas* utilisée, définissez la propriété `wcore` sur `0`.

Notez que la propriété des unités d'allocation, `<rasd:AllocationUnits>`, pour la ressource `cpu`, spécifie toujours le nombre de CPU virtuelles et non le nombre de serveurs de base.

EXEMPLE 17–7 Exemple de XML cpu

L'exemple suivant présente la demande XML équivalente pour la commande `ldm add-vcpu -c 1 ldg1` :

EXEMPLE 17–7 Exemple de XML cpu *(Suite)*

```
<?xml version="1.0"?>
<LDM_interface version="1.3"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="./schemas/combined-v3.xsd"
xmlns:ovf="./schemas/envelope"
xmlns:rasd="./schemas/CIM_ResourceAllocationSettingData"
xmlns:vssd="./schemas/CIM_VirtualSystemSettingData"
xmlns:gprop="./schemas/GenericProperty"
xmlns:bind="./schemas/Binding">
  <cmd>
    <action>add-vcpu</action>
    <data version="3.0">
      <Envelope>
        <References/>
        <Content xsi:type="ovf:VirtualSystem_Type" ovf:id="ldg1">
          <Section xsi:type="ovf:VirtualHardwareSection_Type">
            <Item>
              <rasd:OtherResourceType>cpu</rasd:OtherResourceType>
              <rasd:AllocationUnits>8</rasd:AllocationUnits>
              <gprop:GenericProperty key="wcore">1</gprop:GenericProperty>
            </Item>
          </Section>
        </Content>
      </Envelope>
    </data>
  </cmd>
</LDM_interface>
```

Une ressource cpu est toujours contenue dans une section `<Content>`.

Ressource MAU (mau)

Remarque – La ressource `mau` est une unité cryptographique prise en charge sur un serveur pris en charge. Actuellement, les deux unités cryptographiques prises en charge sont l'unité arithmétique modulaire (MAU) et la file d'attente de mot de contrôle (CWQ).

EXEMPLE 17–8 Exemple de XML mau

```
<Envelope>
  <References/>
  <Content xsi:type="ovf:VirtualSystem_Type" id="ldg1">
    <Section xsi:type="ovf:VirtualHardwareSection_Type">
      <Item>
        <rasd:OtherResourceType>mau</rasd:OtherResourceType>
        <rasd:AllocationUnits>1</rasd:AllocationUnits>
      </Item>
    </Section>
  </Content>
</Envelope>
```

Une ressource mau est toujours contenue dans une section <Content>. La seule propriété est la balise <rasd:AllocationUnits>, qui signifie le nombre de MAU ou d'autres unités cryptographiques.

Ressource de mémoire (memory)

EXEMPLE 17–9 Exemple de XML memory

```
<Envelope>
  <References/>
  <Content xsi:type="ovf:VirtualSystem_Type" id="ldg1">
    <Section xsi:type="ovf:VirtualHardwareSection_Type">
      <Item>
        <rasd:OtherResourceType>memory</rasd:OtherResourceType>
        <rasd:AllocationUnits>4G</rasd:AllocationUnits>
      </Item>
    </Section>
  </Content>
</Envelope>
```

Une ressource memory est toujours contenue dans une section <Content>. La seule propriété est la balise <rasd:AllocationUnits>, qui signifie la quantité de mémoire.

Ressource de serveur de disque virtuel (vds)

EXEMPLE 17–10 Exemple de XML (vds)

```
<Envelope>
  <References/>
  <Content xsi:type="ovf:VirtualSystem_Type" id="ldg1">
    <Section xsi:type="ovf:VirtualHardwareSection_Type">
      <Item>
        <rasd:OtherResourceType>vds</rasd:OtherResourceType>
        <gprop:GenericProperty
          key="service_name">vdstmp</gprop:GenericProperty>
      </Item>
    </Section>
  </Content>
</Envelope>
```

Une ressource de serveur de disque virtuel (vds) peut se trouver dans une section <Content> comme faisant partie de la description du domaine, ou elle peut apparaître seule dans une section <Envelope>. La seule propriété est la balise <gprop:GenericProperty> avec une clé de service_name et qui contient le nom de la ressource vds étant décrite.

Ressource du volume de disque virtuel (vds_volume)

EXEMPLE 17–11 Exemple de XML vds_volume

```
<Envelope>
  <References/>
    <Section xsi:type="ovf:VirtualHardwareSection_Type">
      <Item>
        <rasd:OtherResourceType>vds_volume</rasd:OtherResourceType>
        <gprop:GenericProperty key="vol_name">vdsdev0</gprop:GenericProperty>
        <gprop:GenericProperty key="service_name">primary-vds0</gprop:GenericProperty>
        <gprop:GenericProperty key="block_dev">
          opt/SUNWldm/domain_disks/testdisk1</gprop:GenericProperty>
        <gprop:GenericProperty key="vol_opts">ro</gprop:GenericProperty>
        <gprop:GenericProperty key="mpgroup">mpgroup-name</gprop:GenericProperty>
      </Item>
    </Section>
</Envelope>
```

Une ressource `vds_volume` peut se trouver dans une section `<Content>` comme faisant partie de la description du domaine, ou elle peut apparaître seule dans une section `<Envelope>`. Elle doit comporter des balises `<gprop:GenericProperty>` avec les clés suivantes :

- `vol_name` – Nom du volume
- `service_name` – Nom du serveur de disque virtuel auquel ce volume doit être lié
- `block_dev` – Nom du fichier ou du périphérique à associer à ce volume

Une ressource `vds_volume` peut également avoir les propriétés suivantes :

- `vol_opts` – Une ou plusieurs des chaînes suivantes séparées par des virgules : {`ro,slice,excl`}
- `mpgroup` – Nom du groupe multivoie (basculement)

Ressource de disque (disk)

EXEMPLE 17–12 Exemple de XML disk

```
<Envelope>
  <References/>
  <Content xsi:type="ovf:VirtualSystem_Type" id="ldg1">
    <Section xsi:type="ovf:VirtualHardwareSection_Type">
      <Item>
        <rasd:OtherResourceType>disk</rasd:OtherResourceType>
        <gprop:GenericProperty key="vdisk_name">vdisk0</gprop:GenericProperty>
        <gprop:GenericProperty
          key="service_name">primary-vds0</gprop:GenericProperty>
        <gprop:GenericProperty key="vol_name">vdsdev0</gprop:GenericProperty>
        <gprop:GenericProperty key="timeout">60</gprop:GenericProperty>
      </Item>
    </Section>
```

EXEMPLE 17–12 Exemple de XML disk *(Suite)*

```
  </Content>
</Envelope>
```

Une ressource `disk` est toujours contenue dans une section `<Content>`. Elle doit comporter des balises `<gprop:GenericProperty>` avec les clés suivantes :

- `vdisk_name` – Nom du disque virtuel
- `service_name` – Nom du serveur de disque virtuel auquel ce disque virtuel doit être lié
- `vol_name` – Périphérique du service de disque virtuel avec lequel le disque virtuel doit être lié

La ressource `disk` peut également avoir la propriété `timeout`, qui est la valeur de temporisation en secondes pour l'établissement d'une connexion entre un client de disque virtuel (`vdc`) et un serveur de disque virtuel (`vds`). S'il y a plusieurs chemins vers le disque virtuel (`vdisk`), le `vdc` peut donc essayer de se connecter sur un `vds` différent, et la valeur de temporisation garantit qu'une connexion à un `vds` est établie pendant le délai imparti.

Ressource de commutateur virtuel (vsw)

EXEMPLE 17–13 Exemple de XML vsw

```
<Envelope>
  <References/>
  <Content xsi:type="ovf:VirtualSystem_Type" id="ldg2">
    <Section xsi:type="ovf:VirtualHardwareSection_Type">
      <Item>
        <rasd:OtherResourceType>vsw</rasd:OtherResourceType>
        <rasd:Address>00:14:4f:fb:ec:00</rasd:Address>
        <gprop:GenericProperty key="service_name">test-vsw1</gprop:GenericProperty>
        <gprop:GenericProperty key="inter_vnet_link">on</gprop:GenericProperty>
        <gprop:GenericProperty key="default-vlan-id">1</gprop:GenericProperty>
        <gprop:GenericProperty key="pvid">1</gprop:GenericProperty>
        <gprop:GenericProperty key="mtu">1500</gprop:GenericProperty>
        <gprop:GenericProperty key="dev_path">switch@0</gprop:GenericProperty>
        <gprop:GenericProperty key="id">0</gprop:GenericProperty>
      </Item>
    </Section>
  </Content>
</Envelope>
```

Une ressource `vsw` peut se trouver dans une section `<Content>` comme faisant partie de la description du domaine, ou elle peut apparaître seule dans une section `<Envelope>`. Elle *doit* contenir une balise `<gprop:GenericProperty>` avec la clé `service_name`, qui est le nom à assigner au commutateur virtuel.

La ressource vsw peut également avoir les propriétés suivantes :

- <rasd:Address> – Assigne une adresse MAC au commutateur virtuel
- default-vlan-id – Spécifie le réseau local virtuel (VLAN) par défaut dont un périphérique réseau virtuel ou un commutateur virtuel doit être membre, en mode balisé. Le premier ID de VLAN (*vid1*) est réservé à default-vlan-id.
- dev_path – Chemin du périphérique physique à associer à ce commutateur virtuel
- id – Spécifie l'ID du nouveau périphérique de commutateur virtuel. 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.
- inter_vnet_link – Indique s'il faut assigner les canaux LDC pour la communication inter-vnet. La valeur par défaut est on.
- linkprop – Indique si le périphérique virtuel doit obtenir les mises à jour d'état du lien physique. Lorsque la valeur est phys-state, le périphérique virtuel obtient les mises à jour de l'état du lien physique. Lorsque la valeur est vide, le périphérique virtuel n'obtient pas les mises à jour de l'état du lien physique. Par défaut, le périphérique physique n'obtient pas les mises à jour de l'état du lien physique.
- mode – sc pour la prise en charge de la pulsation du cluster Oracle Solaris.
- pvid – L'identificateur (ID) du réseau local virtuel (VLAN) du port indique le VLAN dont le réseau virtuel doit être membre, en mode non balisé.
- mtu – Spécifie l'unité de transmission maximale (MTU) d'un commutateur virtuel, les périphériques réseau virtuel liés au commutateur virtuel, ou les deux. Les valeurs valides sont dans la plage de 1500-16000. La commande ldm renvoie une erreur si une valeur non valide est spécifiée.
- vid – L'identificateur (ID) du réseau local virtuel (VLAN) indique le VLAN dont le réseau virtuel et le commutateur virtuel doivent être membres, en mode balisé.

Ressource réseau (network)

EXEMPLE 17–14 Exemple de XML network

```
<Envelope>
  <References/>
  <Content xsi:type="ovf:VirtualSystem_Type" id="ldg1">
    <Section xsi:type="ovf:VirtualHardwareSection_Type">
      <Item>
        <rasd:OtherResourceType>network</rasd:OtherResourceType>
        <gprop:GenericProperty key="linkprop">phys-state</gprop:GenericProperty>
        <gprop:GenericProperty key="vnet_name">ldg1-vnet0</gprop:GenericProperty>
        <gprop:GenericProperty
          key="service_name">primary-vsw0</gprop:GenericProperty>
        <rasd:Address>00:14:4f:fc:00:01</rasd:Address>
      </Item>
    </Section>
```

EXEMPLE 17–14 Exemple de XML network *(Suite)*

```
  </Content>
</Envelope>
```

Une ressource `network` est toujours contenue dans une section `<Content>`. Elle doit comporter des balises `<gprop:GenericProperty>` avec les clés suivantes :

- `linkprop` – Indique si le périphérique virtuel doit obtenir les mises à jour d'état du lien physique. Lorsque la valeur est `phys-state`, le périphérique virtuel obtient les mises à jour de l'état du lien physique. Lorsque la valeur est vide, le périphérique virtuel n'obtient pas les mises à jour de l'état du lien physique. Par défaut, le périphérique physique n'obtient pas les mises à jour de l'état du lien physique.
- `vnet_name` – Nom du réseau virtuel (`vnet`)
- `service_name` – Nom du commutateur virtuel (`vswitch`) auquel ce réseau virtuel doit être lié

La ressource `network` peut également avoir les propriétés suivantes :

- `<rasd:Address>` – Assigne une adresse MAC au commutateur virtuel
- `pvid` – L'identificateur (ID) du réseau local virtuel (VLAN) du port indique le VLAN dont le réseau virtuel doit être membre, en mode non balisé.
- `vid` – L'identificateur (ID) du réseau local virtuel (VLAN) indique le VLAN dont le réseau virtuel et le commutateur virtuel doivent être membres, en mode balisé.
- `mode` – `hybrid` pour activer l'E/S hybride pour ce réseau virtuel.

Ressource de concentrateur de console virtuelle (vcc)

EXEMPLE 17–15 Exemple de vcc XML

```
<Envelope>
  <References/>
  <Content xsi:type="ovf:VirtualSystem_Type" id="ldg1">
    <Section xsi:type="ovf:VirtualHardwareSection_Type">
      <Item>
        <rasd:OtherResourceType>vcc</rasd:OtherResourceType>
        <gprop:GenericProperty key="service_name">vcc1</gprop:GenericProperty>
        <gprop:GenericProperty key="min_port">6000</gprop:GenericProperty>
        <gprop:GenericProperty key="max_port">6100</gprop:GenericProperty>
      </Item>
    </Section>
  </Content>
</Envelope>
```

Une ressource vcc peut se trouver dans une section <Content> comme faisant partie de la description du domaine, ou elle peut apparaître seule dans une section <Envelope>. Elle peut comporter des balises <gprop:GenericProperty> avec les clés suivantes :

- service_name – Nom à assigner au service de concentrateur de console virtuelle
- min_port – Numéro de port minimum à associer à ce vcc
- min_port – Numéro de port maximum à associer à ce vcc

Ressource de variable (var)

EXEMPLE 17–16 Exemple de XML var

```
<Envelope>
  <References/>
  <Content xsi:type="ovf:VirtualSystem_Type" id="ldg1">
    <Section xsi:type="ovf:VirtualHardwareSection_Type">
      <Item>
        <rasd:OtherResourceType>var</rasd:OtherResourceType>
        <gprop:GenericProperty key="name">test_var</gprop:GenericProperty>
        <gprop:GenericProperty key="value">test1</gprop:GenericProperty>
      </Item>
    </Section>
  </Content>
</Envelope>
```

Une ressource var est toujours contenue dans une section <Content>. Elle peut comporter des balises <gprop:GenericProperty> avec les clés suivantes :

- name – Name de la variable
- value – Valeur de la variable

Ressource de périphérique d'E/S physique (physio_devise)

EXEMPLE 17–17 Exemple de XML physio_device

```
<Envelope>
  <References/>
  <Content xsi:type="ovf:VirtualSystem_Type" id="ldg1">
    <Section xsi:type="ovf:VirtualHardwareSection_Type">
      <Item>
        <rasd:OtherResourceType>physio_device</rasd:OtherResourceType>
        <gprop:GenericProperty key="name">pci@780</gprop:GenericProperty>
      </Item>
    </Section>
  </Content>
</Envelope>
```

Une ressource `physio_device` est toujours contenue dans une section `<Content>`. La seule propriété est la balise `<gprop:GenericProperty>` avec la valeur de propriété de clé `name`, qui est le nom du périphérique d'E/S étant décrit.

Ressource de configuration du SP (spconfig)

EXEMPLE 17–18 Exemple de XML spconfig

```
<Envelope>
    <Section xsi:type="ovf:ResourceAllocationSection_type">
      <Item>
        <rasd:OtherResourceType>spconfig</rasd:OtherResourceType>
        <gprop:GenericProperty
          key="spconfig_name">primary</gprop:GenericProperty>
        <gprop:GenericProperty
          key="spconfig_status">current</gprop:GenericProperty>
      </Item>
    </Section>
</Envelope>
```

Une configuration de processeur de service (SP) (`spconfig`) apparaît toujours seule dans une section `<Envelope>`. Elle peut comporter des balises `<gprop:GenericProperty>` avec les clés suivantes

- `spconfig_name` – Nom d'une configuration à stocker sur le SP
- `spconfig_status` – Etat actuel d'une configuration de SP spécifique. Cette propriété est utilisée dans la sortie d'une commande `ldm list-spconfig`.

Ressource de configuration de la stratégie DRM (policy)

EXEMPLE 17–19 Exemple policy au format XML

```
<Envelope>
   <Section xsi:type="ovf:VirtualHardwareSection_Type">
     <Item>
       <rasd:OtherResourceType>policy</rasd:OtherResourceType>
       <gprop:GenericProperty key="policy_name">test-policy</gprop:GenericProperty>
       <gprop:GenericProperty key="policy_enable">on</gprop:GenericProperty>
       <gprop:GenericProperty key="policy_priority">1</gprop:GenericProperty>
       <gprop:GenericProperty key="policy_vcpu_min">12</gprop:GenericProperty>
       <gprop:GenericProperty key="policy_vcpu_max">13</gprop:GenericProperty>
       <gprop:GenericProperty key="policy_util_lower">8</gprop:GenericProperty>
       <gprop:GenericProperty key="policy_util_upper">9</gprop:GenericProperty>
       <gprop:GenericProperty key="policy_tod_begin">07:08:09</gprop:GenericProperty>
       <gprop:GenericProperty key="policy_tod_end">09:08:07</gprop:GenericProperty>
       <gprop:GenericProperty key="policy_sample_rate">1</gprop:GenericProperty>
       <gprop:GenericProperty key="policy_elastic_margin">8</gprop:GenericProperty>
```

EXEMPLE 17–19 Exemple policy au format XML *(Suite)*

```
       <gprop:GenericProperty key="policy_attack">8</gprop:GenericProperty>
       <gprop:GenericProperty key="policy_decay">9</gprop:GenericProperty>
     </Item>
   </Section>
</Envelope>
```

Une ressource de stratégie DRM (`policy`) apparaît à la section `<Envelope>` et peut posséder des balises `<gprop:GenericProperty>` avec les clés suivantes :

- `policy_name` : nom de la stratégie DRM
- `policy_enable` : spécifie si la stratégie DRM est activée ou désactivée
- `policy_priority` : priorité de la stratégie DRM
- `policy_vcpu_min` : nombre minimal de ressources de CPU virtuelle pour un domaine
- `policy_vcpu_max` : nombre maximal de ressources de CPU virtuelle pour un domaine
- `policy_util_lower` : niveau d'utilisation inférieur auquel une analyse de stratégie est lancée
- `policy_util_upper` : niveau d'utilisation supérieur auquel une analyse de stratégie est lancé
- `policy_tod_begin` : heure de début effective de la stratégie DRM
- `policy_tod_end` : heure de fin effective de la stratégie DRM
- `policy_sample_rate` : fréquence d'échantillonnage, qui correspond au temps du cycle en secondes
- `policy_elastic_margin` : quantité de tampon entre les limites d'utilisation inférieure et supérieure de la CPU
- `policy_attack` : quantité maximale d'une ressource à ajouter au cours d'un cycle de contrôle de ressource
- `policy_decay` : quantité maximale d'une ressource à supprimer au cours d'un cycle de contrôle de ressource

Ressource de service de canal de plan de données virtuelles (vdpcs)

EXEMPLE 17–20 Exemple de vdpcs du format XML

```
<Envelope>
  <References/>
  <Content xsi:type="ovf:VirtualSystem_Type" id="ldg1">
    <Section xsi:type="ovf:VirtualHardwareSection_Type">
      <Item>
        <rasd:OtherResourceType>vdpcs</rasd:OtherResourceType>
```

EXEMPLE 17–20 Exemple de vdpcs du format XML *(Suite)*

```
        <gprop:GenericProperty key="service_name">dg1-vdpcs</gprop:GenericProperty>
      </Item>
    </Section>
  </Content>
</Envelope>
```

Cette ressource ne présente un intérêt que dans un environnement DPS Netra. Une ressource `vdpcs` peut se trouver dans une section `<Content>` comme faisant partie de la description du domaine, ou elle peut apparaître seule dans une section `<Envelope>`. La seule propriété est la balise `<gprop:GenericProperty>` avec la valeur de propriété de clé`service_name`, qui est le nom de la ressource du service de plan de données virtuelles (`vdpcs`) étant décrite.

Ressource de client de canal de plan de données virtuelles (vdpcc)

EXEMPLE 17–21 Exemple de XML vdpcc

```
<Envelope>
  <References/>
  <Content xsi:type="ovf:VirtualSystem_Type" id="ldg1">
    <Section xsi:type="ovf:VirtualHardwareSection_Type">
      <Item>
        <rasd:OtherResourceType>vdpcc</rasd:OtherResourceType>
        <gprop:GenericProperty key="vdpcc_name">vdpcc</gprop:GenericProperty>
        <gprop:GenericProperty
          key="service_name">ldg1-vdpcs</gprop:GenericProperty>
      </Item>
    </Section>
  </Content>
</Envelope>
```

Cette ressource ne présente un intérêt que dans un environnement DPS Netra. Un ressource de client de canal de plan de données virtuelles est toujours contenue dans une section `<Content>`. Elle peut comporter des balises `<gprop:GenericProperty>` avec les clés suivantes :

- `vdpcc_name` – Nom du client du canal de plan de données virtuelles (`vdpcc`)
- `service_name` – Nom du service de canal de plan de données virtuelles vdpcs auquel vdpcc doit être lié

Ressource de console (console)

EXEMPLE 17–22 Exemple de XML console

```
<Envelope>
  <References/>
  <Content xsi:type="ovf:VirtualSystem_Type" id="ldg1">
    <Section xsi:type="ovf:VirtualHardwareSection_Type">
      <Item>
        <rasd:OtherResourceType>console</rasd:OtherResourceType>
        <gprop:GenericProperty key="port">6000</gprop:GenericProperty>
        <gprop:GenericProperty key="service_name">vcc2</gprop:GenericProperty>
        <gprop:GenericProperty key="group">group-name</gprop:GenericProperty>
      </Item>
    </Section>
  </Content>
</Envelope>
```

Une ressource `console` est toujours contenue dans une section `<Content>`. Elle peut comporter des balises `<gprop:GenericProperty>` avec les clés suivantes :

- `port` – Port sur lequel passer sur cette console virtuelle (`console`)
- `service_name` – Service de concentrateur de console virtuelle (`vcc`) auquel associer cette `console`
- `group` – Nom du groupe auquel associer cette `console`

Migration de domaine

Cet exemple présente le contenu de la section `<data>` pour une sous-commande `migrate-domain`.

EXEMPLE 17–23 Exemple de section `migrate-domain` <data>

```
<Envelope>
  <References/>
  <Content xsi:type="ovf:VirtualSystem_Type" ovf:id="ldg1"/>
  <Content xsi:type="ovf:VirtualSystem_Type" ovf:id="ldg1"/>
    <Section xsi:type="ovf:ResourceAllocationSection_Type">
      <Item>
        <rasd:OtherResourceType>ldom_info</rasd:OtherResourceType>
        <gprop:GenericProperty key="target">target-host</gprop:GenericProperty>
        <gprop:GenericProperty key="username">user-name</gprop:GenericProperty>
        <gprop:GenericProperty key="password">password</gprop:GenericProperty>
      <Item>
    </Section>
  </Content>
</Envelope>
```

où :

- Le premier noeud <Content> (sans section <ldom_info>) est le domaine source à migrer.
- Le second noeud <Content> (avec une section <ldom_info>) est le domaine cible vers lequel migrer. Les noms des domaines source et cible peuvent être identiques.
- La section <ldom_info> du domaine cible décrit la machine que laquelle migrer et les détails nécessaires à la migration sur cette machine :
  - target-host est la machine cible vers laquelle migrer.
  - user-name est le nom d'utilisateur pour la connexion à la machine cible. Doit être codé en SASL 64 bits.
  - password est le mot de passe à utiliser pour la connexion à la machine cible. Doit être codé en SASL 64 bits.

Remarque – Logical Domains Manager utilise sasl_decode64() pour décoder le nom et le mot de passe de l'utilisateur cible et utilise sasl_encode64() pour coder ces valeurs. Le codage SASL 64 est équivalent au codage base64.

Schémas XML

La section suivante répertorie le nom du fichier de schéma XML dans le répertoire /opt/SUNWldm/bin/schemas. Ces schémas sont utilisés par Logical Domains Manager.

- Schéma cim-common.xsd – cim-common.xsd
- Schéma cim-rasd.xsd – cim-rasd.xsd
- Schéma cim-vssd.xsd – cim-vssd.xsd
- Schéma cli-list-constraint-v3.xsd – cli-list-constraint-v3.xsd
- Schéma XML combined-v3.xsd – LDM_interface
- Schéma XML event-v3.xsd – LDM_Event
- Schéma XML ldmd-binding.xsd – Binding_Type
- Schéma XML ldmd-property.xsd – GenericProperty
- Schéma ovf-core.xsd – ovf-core.xsd
- Schéma ovf-envelope.xsd – ovf-envelope.xsd
- Schéma ovf-section.xsd – ovf-section.xsd
- Schéma ovf-strings.xsd – ovf-strings.xsd
- Schéma ovfenv-core.xsd – ovfenv-core.xsd
- Schéma ovfenv-section.xsd – ovfenv-section.xsd

Glossaire

Cette liste 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

`auditreduce` Fusionne et sélectionne des enregistrements d'audit à partir de fichiers de piste d'audit (voir la page de manuel `auditreduce`(1M)).

audit Utilisation du contrôle du SE Oracle Solaris pour identifier la source des modifications de sécurité

autorisation Définition d'une autorisation à l'aide du RBAC du SE Oracle Solaris

B

`bge` Pilote Broadcast Gigabit Ethernet sur périphériques BCM57xx

BSM Module de sécurité de base

`bsmconv` Active le BSM (voir la page de manuel `bsmconv`(1M)).

`bsmunconv` Désactive le BSM (voir la page de manuel `bsmunconv`(1M)).

C

CD Disque compact

CLI Interface de ligne de commande

CMT Chip multithreading

conformité Détermine si la configuration d'un système est en conformité avec un profil de sécurité prédéfini

configuration Nom de la configuration de domaine logique enregistrée sur le processeur de service

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.

domaine de contrôle Domaine privilégié qui crée et gère d'autres services et domaines logiques à l'aide de Logical Domains Manager

CPU Unité de calcul centrale

CWQ CWQ (Control Word Queue - File d'attente de mots de contrôle) ; unité cryptographique des plates-formes Sun UltraSPARC T2 d'Oracle

D

DHCP 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 domaine logique.

Logical Domains Manager CLI permettant de créer et de gérer des domaines logiques et d'allouer les ressources aux domaines

DPS Date plane software (Logiciel de plan de données)

DR Reconfiguration dynamique

`drd` Démon de reconfiguration dynamique du SE Oracle Solaris 10 pour Logical Domains Manager (voir la page de manuel `drd`(1M)).

DRM Gestion dynamique des ressources

DS Module de services du domaine (SE Oracle Solaris 10)

DVD Disque versatile numérique

E

EFI Interface de microprogramme extensible

ETM Module de gestion des tables de codage (SE Oracle Solaris 10)

F

FC_AL Boucle arbitrée de canal de fibre

FMA Architecture de gestion des pannes (Fault Management Architecture)

`fmd` Démon du gestionnaire de défauts du SE Oracle Solaris 10 (voir la page de manuel `fmd`(1M)).

`format` Utilitaire de partitionnement et de maintenance de disque (voir la page de manuel `format`(1M)).

`fmthard` Remplit les étiquettes sur les disques durs (voir la page de manuel `fmthard`(1M)).

FTP File Transfer Protocol

G

Gb Gigabit

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.

H

Sécurisation Modification de la configuration du SE Oracle Solaris pour améliorer la sécurité.

HDD Unité de disque dur

hyperviseur Couche de microprogramme interposé entre le système d'exploitation et la couche matérielle

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

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

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.

`ioctl` Appel de contrôle d'entrée/sortie

IP Internet Protocol

IPMP Internet Protocol Network Multipathing

ISO Organisation internationale de normalisation

K

`kaio` Entrée/sortie de noyau asynchrone

Ko Kilooctet

KU Mise à jour du noyau

L

LAN Réseau local

LDAP 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

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*.

LUN Numéro d'unité logique

M

MAC Adresse de contrôle d'accès au média, que Logical Domains peut assigner automatiquement ou que vous pouvez assigner manuellement

MAU Unité arithmétique modulaire

Mo Méga-octet

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`), kilooctets (`K`) ou méga-octets (`M`). Mémoire virtualisée du serveur qui peut être allouée aux domaines invités.

`metadb` Crée et supprime les répliques de la base de données d'état des métapériphériques Solaris Volume Manager (voir la page de manuel `metadb`(1M)).

`metaset` Configure les ensembles de disques (voir la page de manuel `metaset`(1M)).

`mhd` Opérations de contrôle de disque multihôte (voir la page de manuel `mhd`(7i)).

MIB Base MIB (Management Information Base)

réduction Installation du nombre minimum de packages nécessaires du SE Oracle Solaris principal

MMF Fibre multimode

MMU Unité de gestion de la mémoire

`mpgroup` Nom du groupe multipathing pour le basculement du disque virtuel

`mtu` Unité de transmission maximale

N

NAT Network Address Translation

`ndpsldcc` Client de canal de domaine logique DPS Netra *Voir aussi* `vdpcc`.

`ndpsldcs` Service de canal de domaine logique DPS Netra. *Voir aussi* `vdpcs`.

NFS Network File System

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 MIB (Management Information Base, base d'informations de gestion) de manière unique.

SE Système d'exploitation

OVF Format de virtualisation ouvert

P

P2V Outil de conversion physique-à-virtuel de Logical Domains

PA Adresse physique

PCI Bus d'interconnexion de composants périphériques

PCIe Bus PCI EXPRESS

PCI-X Bus étendu PCI

pcpu CPU physique

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.

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 Imprime le contenu d'un fichier de piste d'audit (voir la page de manuel `praudit`(1M)).

PRI Priorité

PROM Mémoire morte programmable

R

RA Adresse réelle

RAID Redundant Array of Inexpensive Disks

RBAC Contrôle d'accès basé sur les rôles

RPC Appel de procédure à distance

S

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.

contrôleur système (SC) Voir aussi processeur de services

SCSI Small Computer System Interface

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.

SMA Agent de gestion du système

SMF Utilitaire de gestion des services

SMI Structure des informations de gestion

SNMP Protocole SNMP

processeur de service (SP) SP, également appelé contrôleur de système (SC), surveille et exécute la machine physique.

SR-IOV Virtualisation d'E/S à racine unique

SSH Shell sécurisé

`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

TCP Protocole TCP

TLS Transport Layer Security

U

UDP Protocole UDP (User Diagram Protocol)

UFS Système de fichiers UNIX

unicast Communication réseau ayant lieu entre un émetteur unique et un récepteur unique

USB Universal Serial Bus

uscsi Interface de commande SCSI utilisateur (voir la page de manuel `uscsi`(7I)).

UTP Paire torsadée non blindée

V

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. Par exemple, un serveur Sun Fire T2000 8 coeurs d'Oracle a 32 CPU virtuelles qui peuvent être allouées parmi les domaines logiques.

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.

vdpcc Client de canal de plan de données virtuelles dans un environnement DPS Netra

vdpcs Client de service de plan de données virtuelles dans un environnement DPS Netra

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.

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.

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.

VLAN Réseau local virtuel

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`)

vNTS Service de terminal réseau virtuel

vntsd Démon du serveur de terminal réseau virtuel du SE Oracle Solaris 10 pour les consoles Logical Domains (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

W

WAN Réseau étendu

X

XFP eXtreme Fast Path

XML Extensible Markup Language

XMPP Extensible Messaging and Presence Protocol

Z

ZFS Système de fichiers Zettabyte (SE Oracle Solaris 10)

`zpool` Pool de stockage ZFS (voir la page de manuel `zpool`(1M)).

ZVOL Pilote d'émulation de volume ZFS

Index

L

M

O

P

R

S

T

U

V