

Guía de administración del servidor Oracle VM para SPARC 2.0

Copyright © 2007, 2010, Oracle y/o sus subsidiarias. Todos los derechos reservados.

Este software y la documentación relacionada están sujetos a un contrato de licencia que incluye restricciones de uso y revelación, y se encuentran protegidos por la legislación sobre la propiedad intelectual. A menos que figure explícitamente en el contrato de licencia o esté permitido por la ley, no se podrá utilizar, copiar, reproducir, traducir, emitir, modificar, conceder licencias, transmitir, distribuir, exhibir, representar, publicar ni mostrar ninguna parte, de ninguna forma, por ningún medio. Queda prohibida la ingeniería inversa, desensamblaje o descompilación de este software, excepto en la medida en que sean necesarios para conseguir interoperabilidad según lo especificado por la legislación aplicable.

La información contenida en este documento puede someterse a modificaciones sin previo aviso y no se garantiza que se encuentre exenta de errores. Si detecta algún error, le agradeceremos que nos lo comuniqué por escrito.

Si este software o la documentación relacionada se entrega al Gobierno de EE.UU. o a cualquier entidad que adquiera licencias en nombre del Gobierno de EE.UU. se aplicará la siguiente disposición:

U.S. GOVERNMENT RIGHTS Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle America, Inc., 500 Oracle Parkway, Redwood City, CA 94065

Este software o hardware se ha desarrollado para uso general en diversas aplicaciones de gestión de la información. No se ha diseñado ni está destinado para utilizarse en aplicaciones de riesgo inherente, incluidas las aplicaciones que pueden causar daños personales. Si utiliza este software o hardware en aplicaciones de riesgo, usted será responsable de tomar todas las medidas apropiadas de prevención de fallos, copia de seguridad, redundancia o de cualquier otro tipo para garantizar la seguridad en el uso de este software o hardware. Oracle Corporation y sus subsidiarias declinan toda responsabilidad derivada de los daños causados por el uso de este software o hardware en aplicaciones de riesgo.

Oracle y Java son marcas comerciales registradas de Oracle y/o sus subsidiarias. Todos los demás nombres pueden ser marcas comerciales de sus respectivos propietarios.

AMD, Opteron, el logotipo de AMD y el logotipo de AMD Opteron son marcas comerciales o marcas comerciales registradas de Advanced Micro Devices. Intel e Intel Xeon son marcas comerciales o marcas comerciales registradas de Intel Corporation. Todas las marcas comerciales de SPARC se utilizan con licencia y son marcas comerciales o marcas comerciales registradas de SPARC International, Inc. UNIX es una marca comercial registrada con acuerdo de licencia de X/Open Company, Ltd.

Este software o hardware y la documentación pueden ofrecer acceso a contenidos, productos o servicios de terceros o información sobre los mismos. Ni Oracle Corporation ni sus subsidiarias serán responsables de ofrecer cualquier tipo de garantía sobre el contenido, los productos o los servicios de terceros y renuncian explícitamente a ello. Oracle Corporation y sus subsidiarias no se harán responsables de las pérdidas, los costos o los daños en los que se incurra como consecuencia del acceso o el uso de contenidos, productos o servicios de terceros.

Copyright © 2007, 2010, Oracle et/ou ses affiliés. Tous droits réservés.

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

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

Si ce logiciel, ou la documentation qui l'accompagne, est concédé sous licence au Gouvernement des Etats-Unis, ou à toute entité qui délivre la licence de ce logiciel ou l'utilise pour le compte du Gouvernement des Etats-Unis, la notice suivante s'applique :

U.S. GOVERNMENT RIGHTS. Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle America, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

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

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

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

Contenido

Prefacio	13
1 Información general sobre el software del Oracle VM Server for SPARC	17
Hypervisor y Dominios lógicos	17
Administrador de Dominios lógicos	19
Funciones para los dominios	20
Interfaz de la línea de comandos	21
Entrada/salida virtual	21
Configuración de recursos	22
Configuraciones permanentes	22
Herramienta de conversión física a virtual del Oracle VM Server for SPARC	23
Asistente para la configuración de Oracle VM Server for SPARC	23
2 Instalación y habilitación del software	25
Instalación del software del Oracle VM Server for SPARC en un nuevo sistema	26
Actualización del SO de Solaris de Oracle	26
Actualización del firmware del sistema	26
Descarga del Administrador de Dominios lógicos	29
Instalación del Administrador de Dominios lógicos	29
Habilitación del daemon del Administrador de Dominios lógicos	32
Actualización de un sistema que ya usa el Oracle VM Server for SPARC	33
Actualización del SO de Solaris de Oracle	33
Actualización del Administrador de Dominios lógicos y el firmware del sistema	35
Actualización del software Oracle VM Server for SPARC 2.0	36
Configuración predeterminada de fábrica y deshabilitación de Dominios lógicos	37
▼ Quite todos los dominios huésped.	37
▼ Quite todas las configuraciones de los dominios lógicos	37

▼ Restablezca la configuración predeterminada de fábrica.	38
▼ Deshabilite el Administrador de Dominios lógicos	38
▼ Eliminación del Administrador de Dominios lógicos	38
▼ Restaure la configuración predeterminada de fábrica desde el procesador de servicios.	39
3 Seguridad	41
Autorización de Administrador de Dominios lógicos	41
Creación de la autorización y perfiles y asignación de funciones para las cuentas de usuario ...	42
Administración de las autorizaciones de usuario	43
Administración de los perfiles de usuario	43
Asignación de funciones a usuarios	44
Configuración de RBAC para el acceso de la consola huésped	45
Habilitación y uso de auditoría del BSM	46
▼ Habilitación de auditoría de BSM	46
▼ Compruebe que la auditoría BSM está habilitada.	47
▼ Deshabilitación de la auditoría BSM	47
▼ Impresión de una salida de auditoría	47
▼ Giro de los registros de la auditoría	47
4 Configuración de servicios y el dominio de control	49
Mensajes de salida	49
Creación de servicios predeterminados	50
▼ Creación de servicios predeterminados	50
Configuración inicial del dominio de control	51
▼ Configuración del dominio de control	51
Reinicio para el uso de Dominios lógicos	52
▼ Reinicio	52
Habilitación de las funciones de red entre dominio de control/servicio y otros dominios	53
▼ Configuración del conmutador virtual como interfaz primaria.	53
Habilitación del daemon del servidor del terminal de red virtual	54
▼ Habilite el daemon del servidor del terminal de red virtual	54
5 Configuración de los dominios huésped	55
Creación e inicio de dominio huésped	55

▼ Creación e inicio de un dominio huésped	55
Instalación del SO de Solaris de Oracle en un dominio huésped	58
▼ Instalación del SO de Solaris de Oracle en el dominio huésped desde un DVD	58
▼ Instale el SO de Solaris de Oracle en el dominio huésped desde un archivo ISO de Solaris de Oracle	60
▼ Realización de una operación JumpStart en un dominio huésped	61
6 Configuración de dominios E/S	63
Información general sobre los dominios E/S	63
Asignación de buses PCIe	64
▼ Creación de un dominio E/S asignando un bus PCIe	65
Asignación de dispositivos de punto final PCIe	69
Requisitos de hardware y software para E/S directas	71
Limitaciones de E/S directas	71
Planificación de la configuración del dispositivo de punto final PCIe	72
Reinicio del dominio primary	73
Realización de cambios hardware en PCIe	74
▼ Creación un dominio E/S asignando el dispositivo de punto final PCIe	75
7 Uso de discos virtuales	81
Introducción a los discos virtuales	81
Administración de discos virtuales	82
▼ Agregación de disco virtual	83
▼ Exportación del componente posterior de un disco virtual varias veces	83
▼ Cambio de las opciones del disco virtual	84
▼ Cambio de la opción de tiempo de espera	84
▼ Eliminación de disco virtual	84
Identificador de disco virtual y nombre del dispositivo	85
Apariencia del disco virtual	85
Disco lleno	86
Disco de segmento único	86
Opciones del componente posterior del disco virtual	86
Opción de sólo lectura (ro)	87
Opción exclusiva (excl)	87
Opción segmento (slice)	88

Componente posterior de un disco virtual	88
Disco físico o LUN de disco	88
▼ Exportación de un disco físico como disco virtual	89
Segmento de disco físico	89
▼ Exportación de un segmento de disco físico como disco virtual	90
▼ Exportación del segmento 2	90
Archivo y volumen	90
Configuración de ruta múltiple de disco virtual	94
▼ Configuración de la ruta múltiple de disco virtual	96
CD, DVD e imágenes ISO	97
▼ Exportación de un CD o DVD desde el dominio de servicio al dominio huésped	98
▼ Exportación de una imagen ISO desde el dominio primary para la instalación en un dominio huésped	99
Tiempo de espera de disco virtual	100
Disco virtual y SCSI	101
Disco virtual y el comando format(1M)	102
Uso de ZFS con discos virtuales	102
Configuración de un grupo de ZFS en un dominio de servicio	102
Almacenamiento de imágenes de disco con ZFS	103
Creación de una instantánea de la imagen del disco	104
Uso de un clon para ofrecer un dominio nuevo	105
Uso de los administradores de volumen en el entorno Dominios lógicos	106
Uso de discos virtuales para administradores de volúmenes	106
Uso de administradores de volumen para discos virtuales	109
8 Uso de las redes virtuales	111
Introducción a una red virtual	111
Conmutador virtual	112
Dispositivo de red virtual	112
Administración de conmutador virtual	114
▼ Agregación de conmutador virtual	114
▼ Configuración de opciones para un conmutador virtual existente	115
▼ Eliminación de un interruptor virtual	115
Administración del dispositivo de red virtual	116
▼ Agregado de un dispositivo de red virtual	116

▼ Opciones configuradas para un dispositivo de red virtual existente	117
▼ Eliminación del dispositivo de red virtual	117
Identificador del dispositivo virtual y nombre de interfaz de red	118
▼ Búsqueda del nombre de la interfaz de red de SO de Solaris de Oracle	119
Asignación de direcciones MAC automática o manualmente	120
Rango de las direcciones MAC asignadas a Dominios lógicos	121
Algoritmo de asignación automática	121
Detección de duplicación de direcciones MAC	122
Direcciones MAC liberadas	122
Uso de adaptadores de red con Dominios lógicos	123
▼ Determine si el adaptador de una red es compatible con GLDV3	123
Configuración del conmutador virtual y dominio de servicio para NAT y enrutamiento	124
▼ Configuración del conmutador virtual para ofrecer conectividad externa a los dominios	125
Configuración de IPMP en un entorno Dominios lógicos	126
Configuración de dispositivos de red virtual en un grupo IPMP en un dominio	126
Configuración y uso de IPMP en el dominio de servicio	127
Uso de IPMP basado en enlace en funciones de redes virtuales de Dominios lógicos	128
Configuración y uso de IPMP en versiones anteriores a Dominios lógicos 1.3	131
Uso de etiquetado VLAN	133
Puerto VLAN ID (PVID)	134
VLAN ID (VID)	134
▼ Asignación de una VLAN a un conmutador virtual y dispositivo de red virtual	135
▼ Instalación de un dominio huésped cuando el servidor de instalación es una VLAN	135
Uso de E/S híbridas de NIU	136
▼ Configuración de un conmutador virtual con un dispositivo de red NIU	139
▼ Habilitación del modo híbrido	139
▼ Deshabilitación del modo híbrido	139
Uso de la agregación de enlaces con un conmutador virtual	139
Configuración de marcos Jumbo	141
▼ Configuración de la red virtual y de los dispositivos de conmutador virtual para el uso de marcos Jumbo	141
Compatibilidad con versiones anteriores (no preparadas para Jumbo) de los controladores vnet y vsw	144

9 Migración de dominios	145
Introducción a la migración de dominios	145
Información general sobre la operación de migración	146
Compatibilidad de software	146
Autenticación para las operaciones de migración	147
Migración de un dominio	147
Realización de una simulación	147
Realización de migraciones no interactivas	147
Migración de un dominio activo	148
Migración de CPUs en un dominio activo	148
Migración de memoria en un dominio activo	149
Migración de dispositivos de E/S físicos en un dominio activo	150
Migración de dispositivos virtuales de E/S en un dominio activo	150
Migración de una entrada/salida híbrida de NIU en un dominio activo.	151
Migración de unidades criptográficas en un dominio activo	152
Reconfiguración retrasada en un dominio activo	152
Migración mientras un dominio activo está en modo elástico.	152
Operaciones en otros dominios	153
Migración de dominios enlazados o inactivos	153
Migración de CPU en dominio enlazado o inactivo	153
Migración de entrada/salida virtual en un dominio enlazado o inactivo	153
Migración de dispositivos de punto final PCIe en dominios enlazados o inactivos	153
Seguimiento de una migración en curso	154
Cancelación de una migración en curso	155
Recuperación de una migración fallida	155
Ejemplos de migración	156
10 Administración de recursos	159
Reconfiguración de recursos	159
Reconfiguración dinámica	159
Reconfiguración retrasada	160
Asignación de recursos	161
Asignación de CPU	161
Habilitación de la restricción de núcleo completo	162
Deshabilitación de la restricción de núcleo completo	163

Asignación de CPU al dominio de control	163
Interacciones entre la restricción de núcleo completo y otras características del dominio	164
Uso de la reconfiguración dinámica de memoria	166
Agregación de memoria	166
Eliminación de memoria	166
Solicitudes parciales de DR de memoria	167
Reconfiguración de memoria del dominio de control	167
Reconfiguración dinámica y retrasada	168
Alineación de memoria	168
Ejemplos de DR de memoria	170
Uso de la administración de energía	174
Enumeración de cables con administración de energía de CPU	175
Uso de la administración de recursos dinámicos	177
Enumeración de recursos de dominios	180
Salida informatizada	180
Definiciones de marcadores	181
Utilización de la definición estadística	181
Ver varias listas	182
Enumeración de restricciones	185
11 Administración de las configuraciones	187
Operación para guardar las configuraciones del dominio para una reconstrucción en el futuro	187
▼ Operación para guardar las configuraciones del dominio	188
▼ Restablecimiento de la configuración de un dominio desde un archivo XML (<code>l3dm add-domain</code>)	188
▼ Restauración de la configuración de un dominio desde un archivo XML (<code>l3dm init-system</code>)	189
Administración de las configuraciones Dominios lógicos	190
▼ Modificación de la normativa de auto recuperación	192
12 Realización de otras tareas administrativas	195
Introducción de nombres en la CLI	195
Nombres de archivo (<i>file</i>) y nombres de variables (<i>var-name</i>)	195
<i>backend</i> del servidor de disco virtual y nombres del dispositivo del conmutador virtual .	196

Nombre de configuración (<i>config-name</i>)	196
Todos los otros nombres	196
Conexión a una consola huésped sobre una red	196
Uso de grupos de consolas	197
▼ Combine múltiples consolas en un grupo	197
Paro de un dominio muy cargado que puede provocar un retraso de la conexión	198
Funcionamiento del SO de Solaris de Oracle con el Oracle VM Server for SPARC	198
El firmware OpenBoot no está disponible cuando se arranca el SO de Solaris de Oracle .	198
Apagado y reencendido de un servidor	199
No use el comando <code>ps radm(1M)</code> en CPU activas en un dominio gestionado por energía	199
Resultado de las interrupciones de SO de Solaris de Oracle	199
Resultados de detener o reiniciar el dominio de control	199
Uso de Dominios lógicos con el procesador de servicio	200
▼ Restablezca la configuración del dominio a la configuración predeterminada u otra.	200
Configuración de las dependencias de dominio	201
Ejemplo de dependencia de dominios	202
Ciclos de dependencia	203
Determinación de dónde ocurren los errores por la asignación de CPU y direcciones de memoria	205
Asignación de CPU	205
Asignación de memoria	205
Ejemplos de asignación de CPU y memoria	206
Uso de los identificadores únicos universales	207
Comando de información de dominio virtual y API	208
A Herramienta de conversión física a virtual del Oracle VM Server for SPARC	209
Información general de la herramienta P2V del Oracle VM Server for SPARC	209
Fase de recogida	210
Fase de preparación	210
Fase de conversión	211
Dispositivos de componente posterior	212
Instalación de la herramienta P2V de Oracle VM Server for SPARC	213
Requisitos previos	213
Limitaciones	214
▼ Instalación de la herramienta P2V del Oracle VM Server for SPARC	214

Uso del comando <code>ldmp2v</code>	215
B Asistente para la configuración de Oracle VM Server for SPARC	223
Uso del asistente para la configuración (GUI)	223
Uso del asistente para la configuración (<code>ldmconfig</code>)	224
Instalación del asistente para la configuración	224
<code>ldmconfig</code> : Características	225
C Descubrimiento del Administrador de Dominios lógicos	229
Descubrimiento de sistemas que ejecutan el Administrador de Dominios lógicos	229
Comunicación multidifusión	229
Formato del mensaje	230
▼ Descubrimiento del Administrador de Dominios lógicos en ejecución en la subred	230
D Uso de la interfaz XML con el Administrador de Dominios lógicos	233
Transporte de XML	233
Servidor XMPP	234
Conexiones locales	234
Protocolo XML	234
Mensajes de solicitud y respuesta	235
Mensajes de eventos	239
Registro y anulación de registro	239
Los mensajes <code><LDM_event></code>	240
Tipos de eventos	240
Acciones de Administrador de Dominios lógicos	243
Recursos y propiedades de Administrador de Dominios lógicos	244
Recurso de información de dominio (<code>ldom_info</code>)	245
Recurso de CPU (<code>cpu</code>)	246
Recurso de MAU (<code>mau</code>)	247
Recurso de memoria (<code>memory</code>)	247
Recurso de servidor de disco virtual (<code>vds</code>)	248
Recurso del volumen del servidor del disco virtual (<code>vds_volume</code>)	248
Recurso de disco (<code>disk</code>)	249
Recurso de conmutador virtual (<code>vsw</code>)	249

Recurso de red (network)	250
Recurso del concentrador de consola virtual (vcc)	251
Recurso de variable (var)	252
Recurso de dispositivo de E/S físico (physio_device)	253
Recurso de configuración SP (spconfig)	253
Recurso del servicio de canal plano de datos virtual (vdpcs)	254
Recurso de cliente de canal plano de datos virtuales (vdpsc)	254
Recurso de consola (console)	255
Migración de dominio	255
E Esquemas XML de Administrador de Dominios lógicos	257
LDM_interface Esquema XML	257
LDM_Event (Esquema XML)	259
El esquema ovf-envelope.xsd	260
El esquema ovf-section.xsd	262
El esquema ovf-core.xsd	262
El esquema ovf-virtualhardware.xsc	267
El esquema cim-rasd.xsd	268
El esquema cim-vssd.xsd	272
El esquema cim-common.xsd	272
El esquema XML GenericProperty	276
Binding_Type (Esquema XML)	276
Glosario	279
Índice	289

Prefacio

El *Guía de administración del servidor Oracle VM Server for SPARC 2.0* ofrece información y procedimientos detallados que incluyen información general, consideraciones sobre seguridad, instalación, configuración, modificación y ejecución de las tareas detalladas para el programa Oracle VM Server for SPARC 2.0 en los servidores admitidos, servidores blade y módulos de servidor. Véase “Plataformas admitidas” en *Notas sobre la versión del servidor Oracle VM para SPARC 2.0*.

Esta guía ha sido elaborada para los administradores de sistema de dichos servidores que tienen un conocimiento de trabajo de los sistemas UNIX y del sistema operativo Solaris de Oracle (SO de Solaris de Oracle).

Documentación relacionada

La tabla siguiente muestra la documentación disponible para la versión Oracle VM Server for SPARC 2.0. Si no se indica diversamente, estos documentos están disponibles en formato HTML y PDF.

TABLA P-1 Documentación relacionada

Aplicación	Título	Número de referencia
Software Oracle VM Server for SPARC 2.0	<i>Guía de administración del servidor Oracle VM para SPARC 2.0</i>	821-2931
		821-2935
	<i>Notas sobre la versión del servidor Oracle VM para SPARC 2.0</i>	821-1486
	<i>Oracle VM Server for SPARC 2.0 Reference Manual</i> <i>Solaris de Oracle 10 Reference Manual Collection</i>	
	▪ <i>drd(1M)</i> (página de comando man)	
	▪ <i>vntsd(1M)</i> (página de comando man)	
Nociones básicas sobre el programa Dominios lógicos	<i>Beginners Guide to LDOMs: Understanding and Deploying Logical Domains Software</i> (PDF)	820-0832

TABLA P-1 Documentación relacionada (Continuación)

Aplicación	Título	Número de referencia
Dominios lógicos Management Information Base (MIB)	<i>Logical Domains (LDoms) MIB 1.0.1 Administration Guide</i>	820-2319-10
	<i>Logical Domains (LDoms) MIB 1.0.1 Release Notes</i>	820-2320-10
Instalación y configuración de SO de Solaris de Oracle:	<i>Solaris de Oracle 10 9/10 Release and Installation Collection</i>	N/D

Puede encontrar la documentación relativa a su servidor, software o SO de Solaris de Oracle en <http://docs.sun.com>. Use el cuadro de búsqueda para localizar los documentos e información que necesita.

Documentación, asistencia técnica y formación

Consulte los siguientes sitios web para obtener recursos adicionales:

- **Documentación** (<http://docs.sun.com>)
- **Asistencia técnica** (<http://www.oracle.com/us/support/systems/index.html>)
- **Formación** (<http://education.oracle.com>) – Haga clic en el enlace Sun en la parte izquierda de la barra de navegación.

Oracle agradece sus comentarios

Oracle agradece sus comentarios y sugerencias sobre la calidad y utilidad de la documentación. Si detecta errores o tiene alguna sugerencia para implementar mejoras, vaya a <http://docs.sun.com> y haga clic en Feedback. Indique el título y el número de la parte de la documentación junto con el capítulo, sección y número de página, si están disponibles. Díganos si desea que le contestemos.

Oracle Technology Network (<http://www.oracle.com/technetwork/index.html>) ofrece una amplia gama de recursos relacionados con los programas Oracle:

- Hable sobre los problemas técnicos y soluciones en los **Foros** (<http://forums.oracle.com>).
- Obtenga tutoriales con de las operaciones paso a paso en **Oracle By Example** (<http://www.oracle.com/technetwork/tutorials/index.html>).
- Descargue **código de proyectos** (http://www.oracle.com/technology/sample_code/index.html).

Convenciones tipográficas

La siguiente tabla describe las convenciones tipográficas que se usan en este libro.

TABLA P-2 Convenciones tipográficas

Tipo de letra	Significado	Ejemplo
AaBbCc123	Los nombres de los comandos, archivos y directorios y de la salida de ordenador en pantalla	Edite el archivo <code>.login</code> . Use <code>ls -a</code> para enumerar todos los archivos. <code>machine_name%</code> tiene un mensaje.
AaBbCc123	Cuando escribe, la salida en pantalla se resalta	<code>machine_name% su</code> Contraseña:
<i>aabbcc123</i>	Marcador de posición: sustituir con un nombre o un valor real	El comando para eliminar un archivo es <code>rm filename</code> .
<i>AaBbCc123</i>	Títulos de libros, nuevos términos y términos que se resaltar	Lea el capítulo 6 en la <i>Guía del usuario</i> . Un <i>cache</i> es una copia guardada localmente. No guarde el archivo. Nota: Algunos elementos con énfasis aparecen en negrita en línea.

Mensajes del comando de intérprete en ejemplos de comandos

La siguiente tabla muestra los indicadores de sistema predeterminados UNIX y los indicadores de superusuario incluidos en el SO Solaris de Oracle. Tenga en cuenta que el mensaje de sistema predeterminado que se muestra en comando varía, dependiendo de la versión de Solaris de Oracle.

TABLA P-3 Indicadores del comando de intérprete

Comando de intérprete	Indicador
Indicador Bash, indicador Korn e indicador Bourne	\$
Indicador Bash, indicador Korn e indicador Bourne para superusuario	#

TABLA P-3 Indicadores del comando de intérprete *(Continuación)*

Comando de intérprete	Indicador
Indicador de C	machine_name%
Indicador de C para superusuario	machine_name#

Información general sobre el software del Oracle VM Server for SPARC

Este capítulo ofrece la información general sobre el software del Oracle VM Server for SPARC.

El programa del Oracle VM Server for SPARC depende de la versión del SO de Solaris de Oracle específica los parches de software necesarios y versiones especiales del firmware del sistema. Para más información, véase “SO Solaris de Oracle necesario recomendado” en *Notas sobre la versión del servidor Oracle VM para SPARC 2.0*.

Este capítulo trata sobre los siguientes temas:

- “Hypervisor y Dominios lógicos” en la página 17
- “Administrador de Dominios lógicos” en la página 19
- “Herramienta de conversión física a virtual del Oracle VM Server for SPARC” en la página 23
- “Asistente para la configuración de Oracle VM Server for SPARC” en la página 23

Hypervisor y Dominios lógicos

Esta sección ofrece información general sobre el hypervisor de SPARC que admite Dominios lógicos.

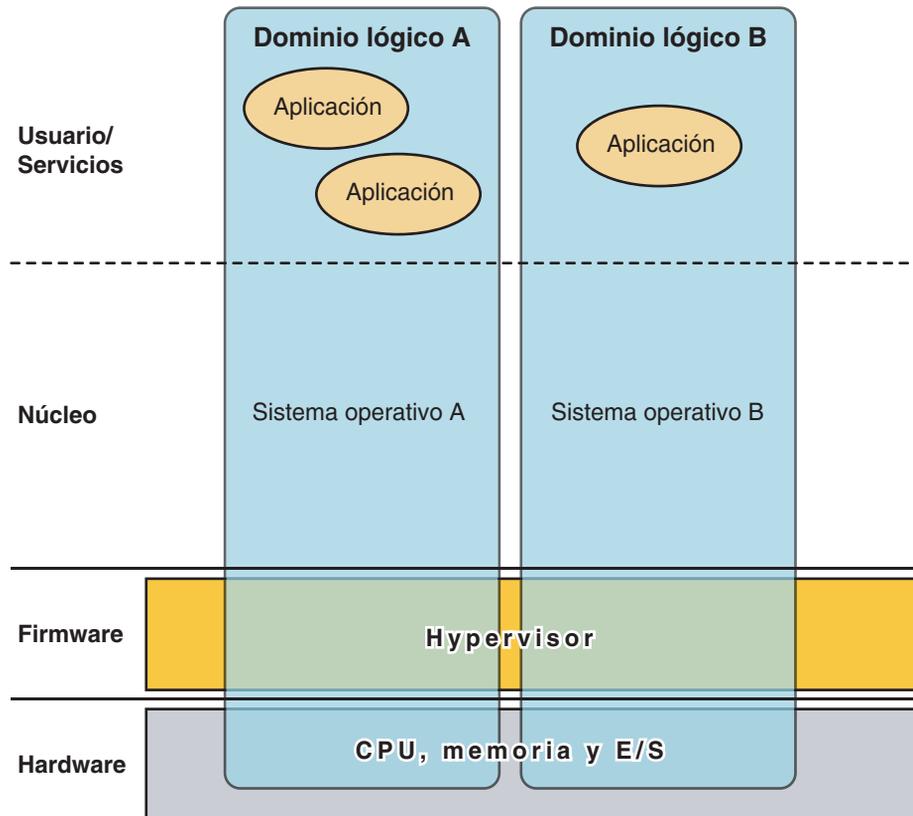
El hypervisor SPARC *hypervisor* es una pequeña capa de firmware que ofrece una arquitectura de equipo virtualizado estable en la que puede escribirse un sistema operativo. Los servidores de Sun de Oracle que usan el hypervisor ofrecen características de hardware para admitir el control del hypervisor sobre las actividades del sistema operativo lógico.

Un *dominio lógico* es un equipo virtual que incluye agrupaciones lógicas discretas de los recursos. Un dominio lógico tiene su propio sistema operativo e identidad en un sistema individual de equipo. Cada uno de los dominios lógicos pueden crearse, destruirse, reconfigurarse y reiniciarse independientemente, sin que sea necesario apagar y volver a encender el servidor. Se puede ejecutar una gran variedad de aplicaciones en diferentes dominios lógicos y mantenerlos independientes por razones de seguridad y rendimiento.

Los dominios lógicos pueden observar e interactuar sólo con los recursos del servidor que el hypervisor ha puesto a disposición. El Administrador de Dominios lógicos le permite especificar qué puede hacer el hypervisor en el dominio de control. Por lo tanto, el hypervisor fuerza la partición de los recursos del servidor y ofrece subconjuntos limitados a múltiples entornos de sistemas operativos. La partición y configuración es el mecanismo fundamental para crear dominios lógicos. El siguiente diagrama muestra un hypervisor que admite dos dominios lógicos. También muestra las siguientes capas que conforman la función Dominios lógicos:

- Aplicaciones, o usuario/servicios
- Núcleo o sistemas operativos
- Firmware o hypervisor
- Hardware, incluyendo CPU, memoria y E/S

FIGURA 1-1 Hypervisor que admite dos dominios



El número y las capacidades de cada dominio lógico que admite un servidor SPARC específico depende de las características del servidor. El hypervisor puede asignar subconjuntos de la CPU, memoria y recursos E/S generales de un servidor a un determinado dominio lógico. Esto hace que se admitan múltiples sistemas operativos simultáneamente, cada uno con su propio dominio lógico. Los recursos pueden reorganizarse entre dominios lógicos separados con una granularidad arbitraria. Por ejemplo, se pueden asignar CPU a un dominio lógico con la granularidad de un subproceso de CPU.

Cada dominio lógico puede administrarse como un equipo completamente independiente con sus propios recursos, como:

- Núcleo, parches y parámetros de ajuste
- Cuentas de usuario y administradores
- Discos
- Interfaces de red, direcciones MAC y direcciones IP

Cada dominio lógico puede pararse, ponerse en marcha y reiniciarse independientemente de cada uno de los otros sin que sea necesario apagar y volver a encender el servidor.

El software del hypervisor es responsable del mantenimiento de la separación entre dominios lógicos. El software del hypervisor también ofrece canales de dominio lógico (LDC) que permiten la comunicación entre los diferentes dominios lógicos. El LDC permite que los dominios ofrezcan servicios unos a otros, como funciones de redes o servicios de disco.

El procesador de servicios (SP), también conocido como controlador de sistema (SC), efectúa un seguimiento y ejecuta el equipo físico, pero no administra los dominios lógicos. El Administrador de Dominios lógicos administra el de los dominios lógicos.

Administrador de Dominios lógicos

El Administrador de Dominios lógicos se usa para crear y administrar los dominios lógicos y asignar los dominios lógicos a recursos físicos. Sólo se puede ejecutar un Administrador de Dominios lógicos en un servidor.

Funciones para los dominios

Todos los dominios lógicos son iguales y pueden distinguirse unos de otros basándose en las funciones que se especifican para cada uno de ellos. A continuación se indican las funciones que pueden realizar los dominios lógicos:

- **Dominio de control.** El Administrador de Dominios lógicos se ejecuta en este dominio, lo que le permite crear y administrar otros dominios lógicos y asignar recursos virtuales a otros dominios. Sólo puede haber un dominio de control por servidor. El dominio de control es el primer dominio creado cuando se instala el software del Oracle VM Server for SPARC. El dominio de control se llama `primary`.
- **Dominio de servicios.** Un dominio de servicios ofrece servicios de dispositivos virtuales a otros dominios, como un conmutador virtual, un concentrador de consola virtual y un servidor de disco virtual. Cualquier dominio puede configurarse como un dominio de servicio.
- **Dominio E/S.** Un dominio E/S tiene acceso directo a un dispositivo de E/S físico, como una tarjeta de red en un controlador PCI EXPRESS (PCIe). Un dominio E/S puede poseer un complejo de raíz PCIe, o puede poseer una ranura PCIe o un dispositivo PCIe integrado usando la características de E/S directa (DIO). Véase [“Asignación de dispositivos de punto final PCIe” en la página 69](#)

Un dominio E/S puede compartir dispositivos E/S físicos con otros dominios en forma de dispositivos virtuales cuando el dominios E/S también se usa como dominio de servicios.

- **Dominio raíz.** Un dominio raíz tiene un complejo de raíz PCIe asignado. Este dominio posee la estructura PCIe y ofrece todos los servicios relacionados con la estructura, como el manejo de error de estructura. Un dominio raíz también es un dominio E/S, ya que posee y tiene acceso directo a los dispositivos de E/S físicos.

El número de dominios raíz que puede tener depende de la arquitectura de la plataforma. Por ejemplo, si usa un servidor Sun SPARC Enterprise T5440, puede tener hasta cuatro dominios raíz.

- **Dominio huésped.** Un dominio huésped es un dominio no E/S que consume servicios del dispositivo virtual que están suministrados por uno o varios dominios de servicios. Un dominio huésped no tiene ningún dispositivo físico E/S, sólo dispositivos virtuales de E/S, como los discos virtuales y las interfaces de redes virtuales.

Puede instalar el Administrador de Dominios lógicos en un sistema existente que aun no esté configurado con Dominios lógicos. En este caso, la instancia actual del SO se convierte en el dominio de control. Asimismo, el sistema está configurado como un sistema Dominios lógicos que sólo tiene un dominio, el dominio de control. Después de la configuración del dominio de control, puede equilibrar la carga de aplicaciones en los dominios para conseguir un uso más eficiente de todo el sistema. Esta operación es realiza agregando dominios y moviendo estas aplicaciones desde el dominio de control a los nuevos dominios.

Interfaz de la línea de comandos

El Administrador de Dominios lógicos usa una interfaz de línea de comandos (CLI) para crear y configurar los dominios lógicos. La CLI es un comando individual, `ldm`, que tiene múltiples subcomandos. Véase la página de comando `man ldm(1M)`.

El daemon el Administrador de Dominios lógicos `daemon, ldmd`, debe estar en ejecución para usar el Administrador de Dominios lógicos CLI.

Entrada/salida virtual

En un entorno Dominios lógicos, puede configurar hasta 128 dominios en un sistema UltraSPARC sistema de procesador T2 Plus y un procesador SPARC T3 . Estos sistemas tienen un número limitado de buses de E/S y ranuras E/S físicas. Como resultado, no puede ofrecer acceso exclusivo a un disco físico y los dispositivos de red a todos los dominios en estos sistemas. Puede asignar un bus PCIe o dispositivo de punto final a un dominio para ofrecerle acceso a un dispositivo físico. Tenga en cuenta que esta solución no es suficiente para ofrecer acceso exclusivo al dispositivo a todos los dominios. Véase [Capítulo 6, “Configuración de dominios E/S”](#). Esta limitación del número de dispositivos de E/S físicos a los que puede accederse directamente está dirigida por la implementación de un modelo de E/S virtualizado.

Cualquier dominio lógico que no tiene acceso físico de E/S se configura con dispositivos virtuales de E/S que se comunican con un dominio de servicios. El dominio de servicios ejecuta un servicio de dispositivo virtual para ofrecer acceso al dispositivo físico o sus funciones. En este modelo cliente-servidor, los dispositivos virtuales de E/S bien se comunican unos con otros o bien con un equivalente de servicios a través de los canales de comunicación entre dominios llamados canales de dominio lógico (LCD). La función E/S virtualizada incluye asistencia para las funciones de red, almacenamiento y consolar virtuales.

Red virtual

Dominios lógicos usa el dispositivo de red virtual y el conmutador de red virtual para implementar las funciones de redes virtuales. El dispositivo de la red virtual (`vnet`) emula un dispositivo Ethernet y se comunica con otros dispositivos `vnet` en el sistema usando un canal punto-a-punto.. El dispositivo de conmutador virtual (`vsw`) funciona principalmente como multiplexor de todas los paquetes de entrada y salida de la red virtual. El dispositivo `vsw` se comunica mediante interfaz directamente con un adaptador de red física en un dominio de servicio, y envía y recibe paquetes en nombre de una red virtual. El dispositivo `vsw` también funciona como conmutador simple de 2 capas y cambia paquetes entre los dispositivos `vnet` conectados al mismo en el sistema.

Almacenamiento virtual

La infraestructura de almacenamiento virtual usa un modelo cliente-servidor para habilitar el acceso de los dominios lógicos a un almacenamiento a nivel de bloque que no les está directamente asignado. El modelo usa los siguientes componentes:

- Cliente de disco virtual (vdc) que exporta una interfaz de dispositivo de bloque
- Servicio de disco virtual (vds) que procesa los requisitos del disco en nombre del cliente del disco virtual y los envía al almacenamiento de componente posterior que reside en el dominio de servicio.

A pesar de que los discos virtuales aparecen como discos normales en el dominio del cliente, la mayoría de las operaciones con el disco se envían al servicio de disco virtual y son procesadas en el dominio de servicios.

Consola virtual

En un entorno Dominios lógicos, las E/S de la consola desde el dominio `primary` está dirigidas al procesador de servicios. La consola E/S desde todos los otros dominios se redirige al dominio de servicios que está ejecutando el concentrador de la consola virtual (vcc). El dominio que ejecuta el vcc es normalmente el dominio `primary`. El servicio de concentrador de consola virtual funciona como concentrador para el tráfico de la consola de todos los dominios y las interfaces con el daemon del servidor terminal de la red virtual (vntsd) para ofrecer acceso a cada consola a través de un punto de conexión UNIX.

Configuración de recursos

Un sistema que ejecuta el software del Oracle VM Server for SPARC puede configurar recursos, como CPU virtuales, dispositivos virtuales de E/S, unidades criptográficas y memoria. Algunos recursos pueden configurarse dinámicamente en un dominio en ejecución mientras que otros deben configurarse en un dominio parado. Si no puede configurarse dinámicamente un recurso en el dominio de control, primero debe iniciar una reconfiguración retrasada. La reconfiguración retrasada pospone las actividades de configuración hasta después del reinicio del dominio de control. Para más información, véase [“Reconfiguración de recursos” en la página 159](#).

Configuraciones permanentes

Puede usar el comando `ldm` para almacenar la configuración actual de un dominio lógico en el procesador de servicios. Puede agregar una configuración, especificar qué configuración usar, eliminar una configuración y enumerar las configuraciones. Véase la página de comando `ldm(1M)`. También puede especificar una configuración para el inicio desde la SP. Véase [“Uso de Dominios lógicos con el procesador de servicio” en la página 200](#).

Para más información sobre la administración de las configuraciones, véase “[Administración de las configuraciones Dominios lógicos](#)” en la página 190.

Herramienta de conversión física a virtual del Oracle VM Server for SPARC

La herramienta de conversión física a virtual del Oracle VM Server for SPARC (P2V) convierte automáticamente un sistema físico existente que se ejecuta en un dominio lógico en un sistema de multiprocesamiento de chip (CMT). El sistema de origen puede ser cualquiera de los siguientes:

- Cualquier sistema sun4u SPARC que ejecute como mínimo el sistema operativo 8 de Solaris
- Cualquier sistema sun4v que ejecute el SO 10 de Solaris de Oracle pero no ejecute el software del Oracle VM Server for SPARC

Para más información sobre la herramienta y cómo instalarla, véase [Apéndice A, “Herramienta de conversión física a virtual del Oracle VM Server for SPARC”](#). Para más información sobre el comando `ldmp2v`, véase la página de comando `man ldmp2v(1M)`.

Asistente para la configuración de Oracle VM Server for SPARC

El asistente para la configuración de Oracle VM Server for SPARC le guiará a través de la configuración del dominio lógico configurando las propiedades básicas. Puede usarse para configurar cualquier sistema en el que está instalado el software del Oracle VM Server for SPARC pero que no esté aún configurado.

Después de obtener los datos de configuración, el asistente para la configuración crea una configuración que es adecuada para el inicio como dominio lógico. También puede usar los valores predeterminados seleccionados por el asistente para la configuración para crear una configuración del sistema utilizable.

El asistente para configuración está disponible tanto como interfaz gráfica de usuario (GUI) y como herramienta basada en terminal.

Para más información, véase [Apéndice B, “Asistente para la configuración de Oracle VM Server for SPARC”](#) y la página de comando `man ldmconfig(1M)`.

Instalación y habilitación del software

Este capítulo describe cómo instalar o actualizar los diferentes componentes del software necesarios para habilitar el software Oracle VM Server for SPARC 2.0. El uso del software de Oracle VM Server for SPARC requiere los siguientes componentes:

- Para conocer las plataformas admitidas, consulte “Plataformas admitidas” en *Notas sobre la versión del servidor Oracle VM para SPARC 2.0* para obtener una lista de las plataformas admitidas.
- Un dominio de control en ejecución en un sistema operativo equivalente al menos a SO 10 9/10 Solaris de Oracle con cualquier parche recomendado en “Parches y programas necesarios” en *Notas sobre la versión del servidor Oracle VM para SPARC 2.0*. Véase “Actualización del SO de Solaris de Oracle” en la página 33.
- Como mínimo la versión 7.3.0 del firmware del sistema para la plataforma Sun UltraSPARC T2 o T2 Plus y la versión 8.0.0 para la plataforma SPARC T3. Véase “Actualización del firmware del sistema” en la página 26.
- Software Oracle VM Server for SPARC 2.0 instalado y habilitado en el dominio de control. Véase “Instalación del Administrador de Dominios lógicos” en la página 29.
- (Opcional) el paquete de software del Management Information Base (MIB) Oracle VM Server for SPARC. Consulte *Logical Domains (LDoms) MIB 1.0.1 Administration Guide* para más información sobre cómo usar el MIB de Dominios lógicos.

El SO de Solaris de Oracle y el firmware de sistema deben ser instalados o actualizados en el servidor antes de instalar o actualizar el Administrador de Dominios lógicos. Si el sistema ya usa el software de Oracle VM Server for SPARC, véase “Actualización de un sistema que ya usa el Oracle VM Server for SPARC” en la página 33. En caso contrario, véase “Instalación del software del Oracle VM Server for SPARC en un nuevo sistema” en la página 26.

Este capítulo trata sobre los siguientes temas:

- “Instalación del software del Oracle VM Server for SPARC en un nuevo sistema” en la página 26
- “Actualización de un sistema que ya usa el Oracle VM Server for SPARC” en la página 33

- “Configuración predeterminada de fábrica y deshabilitación de Dominios lógicos” en la página 37

Nota – El software Solaris Security Toolkit (SST) ya no se incluye con el software del Oracle VM Server for SPARC. Si desea usar la versión más reciente del software SST, véase *Notas sobre la versión del servidor Oracle VM para SPARC 2.0*.

Instalación del software del Oracle VM Server for SPARC en un nuevo sistema

Las plataformas Sun de Oracle que admiten el software del Oracle VM Server for SPARC llevan preinstalado el SO 10 de Solaris de Oracle. Inicialmente la plataforma aparece como un sistema individual que aloja un solo sistema operativo. Después de haber instalado el SO de Solaris de Oracle, el firmware del sistema, y el Administrador de Dominios lógicos, el sistema original y la instancia del SO de Solaris de Oracle se convierten en el dominio de control. El primer dominio de la plataforma se llama `primary` y no puede cambiar este nombre o destruir ese dominio. Desde aquí, la plataforma puede reconfigurarse para tener múltiples dominios que alojarán diferentes estancias del SO de Solaris de Oracle.

Actualización del SO de Solaris de Oracle

En un sistema completamente nuevo, puede desear reinstalar el SO de manera que se adecúe a la normativa de instalación. En este caso, consulte “SO Solaris de Oracle necesario recomendado” en *Notas sobre la versión del servidor Oracle VM para SPARC 2.0* para encontrar el SO 10 de Solaris de Oracle que debe usar para esta versión del software del Oracle VM Server for SPARC. Consulte la guía para la instalación de Solaris de Oracle 10 para conocer las instrucciones completas para la instalación del SO de Solaris de Oracle. Puede adaptar la instalación a las necesidades del sistema.

Si el sistema ya está instalado entonces necesita ser actualizado para el SO 10 de Solaris de Oracle adecuado que debe usarse para esta versión del software del Oracle VM Server for SPARC. Consulte “Software y parches necesarios” en *Notas sobre la versión del servidor Oracle VM para SPARC 2.0* para encontrar el SO 10 de Solaris de Oracle que debe usar para esta versión del software del Oracle VM Server for SPARC y los parches necesarios y recomendados. Consulte [Solaris 10 9/10 Release and Installation Collection \(http://docs.sun.com/app/docs/doc/11/1236.11\)](http://docs.sun.com/app/docs/doc/11/1236.11) para las instrucciones completas para la actualización del SO.

Actualización del firmware del sistema

Las siguientes tareas describen cómo efectuar la actualización del firmware del sistema usando el software Integrated Lights Out Manager (ILOM).

Para más información sobre la actualización del firmware del sistema usando el software ILOM, véase “Update the Firmware” en *Sun SPARC Enterprise T5120 and T5220 Servers Topic Set* y “Updating ILOM Firmware” en *Sun Integrated Lights Out Manager (ILOM) 3.0 CLI Procedures Guide* (<http://d1c.sun.com/pdf/820-6412-12/820-6412-12.pdf>).

▼ Actualización del firmware del sistema

Puede encontrar el firmware del sistema para la plataforma en <http://www.oracle.com/technetwork/systems/patches/firmware/index.html>.

Para más información sobre el firmware de sistema necesario para los servidores admitidos, véase “Parches de firmware del sistema necesarios” en *Notas sobre la versión del servidor Oracle VM para SPARC 2.0*.

Para actualizar el firmware del sistema desde el dominio de control, consulte las notas sobre la versión del firmware del sistema.

Consulte las guías de administración o las notas de producto para los servidores admitidos para más información sobre la instalación y actualización del firmware del sistema para estos servidores.

También puede usar la interfaz web de ILOM para actualizar el firmware del sistema, véase “Actualización del firmware de ILOM” en *Sun Integrated Lights Out Manager (ILOM) 3.0 Web Interface Procedures Guide* (<http://d1c.sun.com/pdf/820-6411-12/820-6411-12.pdf>).

1 Descargue la imagen del firmware del sistema a otro sistema que esté ejecutando el servicio `tftp`.

a. Asegúrese de que el servicio `tftp` está en línea en el servidor.

```
# svcs tftp/udp6
STATE          STIME      FMRI
online         Mar_26    svc:/network/tftp/udp6:default
```

b. Habilite el servicio `tftp` sino no está en estado en línea.

```
# svcadm enable tftp/udp6
```

c. Descargue la imagen del firmware del sistema al directorio `/tftpboot`.

2 Compruebe que está configurado el puerto de administración de red del procesador de servicios de ILOM.

Esta configuración es necesaria para poder acceder a la nueva imagen de actualización a través de la red. Véase “To Configure the Service Processor Network Management Port” en *Sun SPARC Enterprise T5120 and T5220 Servers Topic Set* y “Updating ILOM Firmware” en *Sun Integrated Lights Out Manager (ILOM) 3.0 CLI Procedures Guide* (<http://d1c.sun.com/pdf/820-6412-12/820-6412-12.pdf>).

3 Inicie la sesión en SSH para conectarse al procesador de servicios.

```
$ ssh root@system-name
...
Are you sure you want to continue connecting (yes/no)? yes
...
Password: password
...
->
```

4 Verifique que el host esté apagado.

a. Escriba el comando siguiente:

```
-> show /SYS power_state
```

b. Si el host no está apagado, escriba el siguiente comando:

```
-> stop /SYS
```

5 Verifique que el parámetro `keyswitch_state` esté establecido en `normal`.

a. Escriba el comando siguiente:

```
-> show /SYS keyswitch_state
```

b. Si el valor es diferente de `normal`, fíjelo usando el siguiente comando:

```
-> set /SYS keyswitch_state=normal
```

6 Actualice la imagen de actualización del procesador de servicios y el firmware del host.

```
-> load -source \
tftp://IP-addr/pathname/Sun_System_Firmware-x_x_x_build_nn-server-name.pkg
```

La opción `-source` especifica la dirección IP y el nombre de la ruta completo (URL) a la imagen de actualización del firmware del sistema.

- *IP-addr* es la dirección IP de un servidor `tftp` en una red que puede tener acceso a la imagen de actualización.
- *pathname* es el nombre de la ruta completo a la imagen de actualización en el servidor `tftp`.
- *x_x_x* es el número de versión del firmware del sistema
- *nn* es el número de versión que se aplica a esta aplicación.
- *server-name* es el nombre del servidor.

Por ejemplo, para el servidor SPARC Enterprise T5440, *server-name* es `SPARC_Enterprise_T5440`.

Por ejemplo, la opción `-source`

```
tftp://192.168.1.1/Sun_System_Firmware-7_3_0-SPARC_Enterprise_T5440.pkg
```

apunta hacia el archivo `/tftpboot/Sun_System_Firmware-7_3_0-SPARC_Enterprise_T5440.pkg` en el servidor con la dirección IP `192.168.1.1`.

Tras la actualización de la imagen, el sistema se reiniciará automáticamente.

El procesador de servicios se reinicia, ejecuta las pruebas de diagnóstico y vuelve a presentar el indicador de inicio de sesión (en la consola serie).

Descarga del Administrador de Dominios lógicos

▼ Descarga del software

1 Descargue el archivo ZIP (OVM_Server_SPARC-2_0.zip).

Puede encontrar el software en <http://www.oracle.com/virtualization>.

2 Descomprima el archivo ZIP.

```
$ unzip OVM_Server_SPARC-2_0.zip
```

Véase “Location of Oracle VM Server for SPARC 2.0 Software” en *Notas sobre la versión del servidor Oracle VM para SPARC 2.0* para más detalles sobre la estructura del archivo y lo que incluye.

Instalación del Administrador de Dominios lógicos

Hay tres métodos para instalar el software del Administrador de Dominios lógicos:

- Uso de la secuencia de comandos de instalación para instalar los paquetes y parches. Esta operación instala automáticamente el software del Administrador de Dominios lógicos. Véase “[Instalación del software del Administrador de Dominios lógicos automáticamente](#)” en la página 30.
- Uso de JumpStart para instalar los paquetes como parte de la instalación de una red de Solaris de Oracle. Véase “[Uso de JumpStart para instalar el software Oracle VM Server for SPARC 2.0](#)” en la página 31.
- Instalación del paquete manualmente. Véase “[Instalación del software del Administrador de Dominios lógicos manualmente](#)” en la página 32.

Nota – Recuerde que es necesario instalar manualmente el paquete de software MIB de Dominios lógicos MIB después de haber instalado los paquetes del Oracle VM Server for SPARC. No se instala automáticamente con los otros paquetes. Consulte [Logical Domains \(LDoms\) MIB 1.0.1 Administration Guide](#) para más información sobre cómo instalar y usar el MIB Dominios lógicos.

Instalación del software del Administrador de Dominios lógicos automáticamente

Si usa la secuencia de comandos de instalación `install -ldm`, tiene varias opciones para especificar cómo desea que se ejecute la secuencia de comandos. En los siguientes procedimientos se describe cada opción:

- **Si se usa la secuencia de comandos `install -ldm` sin opciones se realizan las siguientes operaciones automáticamente:**
 - Comprueba que la versión del SO de Solaris de Oracle es SO 10 9/10 Solaris de Oracle como mínimo
 - Comprueba que están presentes los subdirectorios del paquete `SUNWldm/` y `SUNWldmp2v/`
 - Comprueba que los paquetes del controlador del Dominios lógicos que se necesitan previamente, `SUNWldomr` y `SUNWldomu`, están presentes
 - Comprueba que los paquetes `SUNWldm` y `SUNWldmp2v` no se han instalado
 - Instala el software Oracle VM Server for SPARC 2.0
 - Comprueba que se han instalado todos los paquetes
 - Si ya está instalado el SST (`SUNWjass`) se le solicita que se reconfigure el SO de Solaris de Oracle en el dominio de control.
 - Determine si es necesario usar el Asistente para la configuración de Oracle VM Server for SPARC (`ldmconfig`) para realizar la instalación.
- **Si se usa la secuencia de comando `install -ldm` con la opción `-c` se ejecuta automáticamente el Asistente para la configuración de Oracle VM Server for SPARC después de haber instalado el software.**
- **Si se usa la secuencia de comandos `install -ldm` con la opción `-s` se evita la ejecución del Asistente para la configuración de Oracle VM Server for SPARC.**
- **Usando la secuencia de comandos `install -ldm` y las siguientes opciones con el software SST le permite realizar las siguientes operaciones:**
 - `install -ldm -d`. Le permite especificar un controlador SST que no sea un controlador acabado en `-secure.driver`. Esta opción realiza automáticamente todas las funciones enumeradas en la elección anterior y reconfigura el SO de Solaris de Oracle en el dominio de control con el controlador SST personalizado que se especifique; por ejemplo `server-secure-myname.driver`.
 - `install -ldm -d none`. Indique que *no* desea reconfigurar el SO de Solaris de Oracle en ejecución en el dominio de control usando el SST. Esta opción realiza automáticamente todas las funciones excepto la reconfiguración indicada en las anteriores opciones. Se desaconseja omitir el uso de SST y debe realizarse sólo cuando desee reconfigurar el dominio de control usando un proceso alternativo.

- `install -ldm -p`. Especifica que sólo desea realizar las acciones de post-instalación de habilitación del daemon del Administrador de Dominios lógicos (`ldmd`) y ejecución del SST. Por ejemplo, se usa esta opción si los paquetes `SUNWldm` y `SUNWjass` están preinstalados en el servidor.

Uso de JumpStart para instalar el software Oracle VM Server for SPARC 2.0

Véase *JumpStart Technology: Effective Use in the Solaris Operating Environment* para una información completa sobre el uso de JumpStart.

▼ Configuración de un servidor JumpStart

Si aun no ha configurado una servidor JumpStart, debe hacerlo. Véase la *Guía de instalación de Oracle Solaris 10 9/10: Instalaciones JumpStart personalizadas y avanzadas* para una información completa sobre este procedimiento.

1 Consulte *Guía de instalación de Oracle Solaris 10 9/10: Instalaciones JumpStart personalizadas y avanzadas*.

Lleve a cabo los siguientes pasos:

- a. Véase “**Mapa de tareas de preparación de instalaciones JumpStart personalizadas**” de *Guía de instalación de Oracle Solaris 10 9/10: Instalaciones JumpStart personalizadas y avanzadas*.
- b. Configure los sistemas de red con los procedimientos indicados en “**Creación de un servidor de perfil para los sistemas de red.**”
- c. Cree un perfil con el procedimiento indicado en “**Creación de un perfil,**” y agregue una línea en el perfil para instalar el paquete `SUNWldm.v` usando la palabra clave del perfil de `package`.
Por ejemplo, agregue la siguiente línea al perfil para instalar el paquete `SUNWldm.v` desde el directorio `extra` desde el servidor HTTP `192.168.254.255`.

```
package          SUNWldm.v http://192.168.254.255/extra timeout 5
```
- d. Cree el archivo `rules` con el procedimiento indicado en “**Creación del archivo rules.**”

2 Valide el archivo `rules` con el procedimiento indicado en “**Validación del archivo rules.**”

Instalación del software del Administrador de Dominios lógicos manualmente

▼ Instalación del software de Oracle VM Server for SPARC 2.0 manualmente

Antes de empezar Descargue el software Oracle VM Server for SPARC 2.0 (los paquetes SUNWldm y SUNWldmp2v). Véase “[Descarga del software](#)” en la [página 29](#) para instrucciones específicas.

1 Use el comando pkgadd para instalar los paquetes SUNWldm.v y SUNWldmp2v.

Para más información sobre el comando pkgadd, véase la página de comandos man [pkgadd\(1M\)](#).

La opción -G instala el paquete en la zona global sólo y la opción -d especifica el directorio que contiene los paquetes SUNWldm.v y SUNWldmp2v.

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

2 Responda y para "Sí" en todas las preguntas en los mensajes interactivos.

3 Use el comando pkginfo para comprobar que los paquetes de Oracle VM Server for SPARC 2.0, SUNWldm y SUNWldmp2v, están instalados.

Para más información sobre el comando pkginfo, véase la página de comando man [pkginfo\(1\)](#).

La información sobre la revisión (REV) mostrada a continuación es un ejemplo.

```
# pkginfo -l SUNWldm | grep VERSION
VERSION=2.0,REV=2010.08.03.10.20
```

Habilitación del daemon del Administrador de Dominios lógicos

La secuencia de comandos de instalación `install -ldm` habilita automáticamente el daemon del Administrador de Dominios lógicos (`ldmd`). El daemon `ldmd` también se habilita automáticamente cuando se instala el paquete `SUNWldm`. Cuando está habilitado, puede crear, modificar y controlar los dominios lógicos.

▼ Habilitación del daemon del Administrador de Dominios lógicos

Use este procedimiento para habilitar el daemon `ldmd` si ha sido deshabilitado.

1 Use el comando svcadm para habilitar el daemon del Administrador de Dominios lógicos, ldmd.

Para más información sobre el comando `svcadm`, véase la página de comando man [svcadm\(1M\)](#).

```
# svcadm enable ldmd
```

- 2 Use el comando `ldm list` para comprobar que el Administrador de Dominios lógicos está en ejecución.

El comando `ldm list` debe enumerar todos los dominios actualmente definidos en el sistema. En especial, el dominio `primary` debe estar enumerado y estar en estado `active`. La siguiente salida de muestra indica que sólo el dominio `primary` está definido en el sistema.

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

Actualización de un sistema que ya usa el Oracle VM Server for SPARC

Esta sección describe el proceso de actualización del firmware de un SO de Solaris de Oracle y componentes del Administrador de Dominios lógicos en un sistema que ya usa el software del Oracle VM Server for SPARC.

Si el sistema ya está configurado con el software del Oracle VM Server for SPARC, es necesario actualizar el dominio de control. También es necesario actualizar los otros dominios existentes si desea poder utilizar todas las características del software de Oracle VM Server for SPARC 2.0.

Actualización del SO de Solaris de Oracle

Consulte “Software y parches necesarios” en *Notas sobre la versión del servidor Oracle VM para SPARC 2.0* para encontrar el SO 10 de Solaris de Oracle que debe usar para esta versión del software del Oracle VM Server for SPARC y los parches necesarios y recomendados para los diferentes dominios. Consulte la guía para la instalación de Solaris de Oracle 10 para conocer las instrucciones completas para la actualización del SO de Solaris de Oracle.

Cuando se reinstala el SO de Solaris de Oracle en el dominio de control, es necesario guardar y restaurar los datos de la configuración de autoguardado de Dominios lógicos y el archivo de la base de datos de restricciones, tal y como se describe en esta sección.

Cómo guardar y restaurar los directorios de configuración de autoguardado

Desde la versión Dominios lógicos 1.2, puede guardar y restaurar los directorios de configuración de autoguardado antes de reinstalar el sistema operativo en el dominio de control. Si en cualquier momento reinstala el sistema operativo en el dominio de control, debe guardar y restaurar los datos de configuración de autoguardado de Dominios lógicos, que se encuentran en los directorios `/var/opt/SUNWldm/autosave-autosave-name`.

Puede usar el comando `tar` o `cpio` para guardar y restaurar todos los contenidos de los directorios.

Nota – Cada directorio de autoguardado incluye una marca de fecha para la última actualización de la configuración de SP para la configuración relativa. Si restaura los archivos de autoguardado, la marca de hora puede no estar sincronizada. En este caso, las configuraciones autoguardadas restauradas se muestran en el estado previo ya sea [newer] o actualizada.

Para más información sobre las configuraciones de autoguardado, véase “[Administración de las configuraciones Dominios lógicos](#)” en la página 190.

▼ Operación para guardar y restaurar los directorios de autoguardado

Este procedimiento muestra cómo guardar y restaurar los directorios de autoguardado.

1 Guarde los directorios de autoguardado.

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

2 (Opcional) Elimine los directorios de autoguardado existentes para asegurarse de que la operación de restauración será limpia.

A veces el directorio de autoguardado puede incluir archivos extraños, que quizás han quedado de una configuración anterior, que podrían corromper la configuración descargada al SP. En estos casos, limpie el directorio de autoguardado antes de efectuar la operación de restauración tal y como se muestra en este ejemplo:

```
# cd /  
# rm -rf var/opt/SUNWldm/autosave-*
```

3 Restaure los directorios de autoguardado.

Estos comandos restauran los archivos y directorios en el directorio `/var/opt/SUNWldm`.

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

Cómo guardar y restaurar el archivo de la base de datos de las restricciones de Dominios lógicos

Si actualiza el sistema operativo en el dominio de control, debe guardar y restaurar el archivo de la base de datos de restricciones de Dominios lógicos que puede encontrar en `/var/opt/SUNWldm/lldom-db.xml`.

Nota – Asimismo, guarde y restaure el archivo `/var/opt/SUNWldm/lldom-db.xml` cuando realice cualquier otra operación destructiva para los datos del archivo del dominio de control, como un intercambio de disco.

Conservación del archivo de la base de datos de restricciones de Dominios lógicos cuando se utiliza la actualización automática.

Si está utilizando la actualización automática en el dominio de control, considere el agregado de la siguiente línea en el archivo `/etc/lu/syncList`:

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

Esto hace que la base de datos se copie automáticamente desde el entorno de inicio activo al nuevo entorno de inicio cuando se cambian los entornos de inicio. Para más información sobre `/etc/lu/syncList` y la sincronización de archivos entre entornos de inicio, consulte [“Sincronización de archivos entre entornos de inicio” de Guía de instalación de Oracle Solaris 10 9/10: Actualización automática de Solaris y planificación de la actualización.](#)

Actualización desde el SO 10 de Solaris de Oracle anterior al SO 10 5/08 de Solaris de Oracle

Si el dominio de control se actualiza desde una versión del SO 10 de Solaris de Oracle que es anterior al SO 10 5/08 de Solaris de Oracle (o sin el parche 127127-11) y si los volúmenes del administrador de volumen se exportan como discos virtuales, entonces los componentes posteriores de los discos virtuales deben volver a exportarse con `options=slice` después de haber actualizado el Administrador de Dominios lógicos. Véase [“Exportación de volúmenes y compatibilidad con versiones anteriores” en la página 93](#) para más información.

Actualización del Administrador de Dominios lógicos y el firmware del sistema

Esta sección le muestra cómo actualizar el software del Oracle VM Server for SPARC 2.0.

Primero descargue el Administrador de Dominios lógicos al dominio de control. Véase [“Descarga del Administrador de Dominios lógicos” en la página 29.](#)

Entonces, pare todos los dominios (excepto el dominio de control) en ejecución en la plataforma:

▼ Pare todos los dominios en ejecución en la plataforma, excepto el dominio de control

- 1 Cancele cada dominio con el mensaje `ok`.
- 2 Pare todos los dominios usando la opción `-a`.

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

- 3 Ejecute el subcomando `unbind-domain` desde el dominio de control para cada dominio.

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

Actualización del software Oracle VM Server for SPARC 2.0

Esta sección le muestra cómo actualizar el software del Oracle VM Server for SPARC 2.0.

Realice el procedimiento “Actualización desde dominios lógicos 1.0 Solo software” en *Notas sobre la versión del servidor Oracle VM para SPARC 2.0* si desea usar las configuraciones 1.0 de Dominios lógicos existentes con el programa de Oracle VM Server for SPARC 2.0. Las configuraciones existentes de Dominios lógicos 1.0 *no* funcionan con el software de Oracle VM Server for SPARC 2.0.

Si está actualizando desde versiones más recientes del software del Oracle VM Server for SPARC, realice el procedimiento “[Actualización al software Oracle VM Server for SPARC 2.0](#)” en la [página 36](#). Estas configuraciones existentes de Dominios lógicos *sí* funcionan con el software de Oracle VM Server for SPARC 2.0.

▼ Actualización al software Oracle VM Server for SPARC 2.0

- 1 Realizar una actualización flash del firmware del sistema.

Para todo el procedimiento, véase “[Actualización del firmware del sistema](#)” en la [página 27](#).

- 2 Deshabilite el daemon del Administrador de Dominios lógicos (`ldmd`).

```
# svcadm disable ldmd
```

- 3 Quite el antiguo paquete `SUNWldm`.

```
# pkgrm SUNWldm
```

- 4 Agregue el nuevo paquete `SUNWldm`.

Si se especifica la opción `-d` se asume que el paquete está en el directorio actual.

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

- 5 Use el comando `ldm list` para comprobar que se está ejecutando el Administrador de Dominios lógicos.

El comando `ldm list` debe enumerar todos los dominios actualmente definidos en el sistema. En especial, el dominio `primary` debe estar enumerado y estar en estado `active`. La siguiente salida de muestra indica que sólo el dominio `primary` está definido en el sistema.

```
# ldm list
NAME          STATE  FLAGS  CONS  VCPU  MEMORY  UTIL  UPTIME
primary      active ---c-  SP    32    3264M  0.3%  19d 9m
```

Configuración predeterminada de fábrica y deshabilitación de Dominios lógicos

La configuración inicial en la que la plataforma aparece como un solo sistema que aloja solo un sistema operativo se llama configuración predeterminada de fábrica. Si desea deshabilitar los dominios lógicos, probablemente también desee restaurar esta configuración de manera que el sistema pueda volver a acceder a todos los recursos (CPUs, memoria, E/S), que pueden haber sido asignados a otros dominios.

Esta sección describe cómo eliminar todos los dominios huéspedes, eliminar las configuraciones Dominios lógicos y volver a la configuración predeterminada de fábrica.

▼ Quite todos los dominios huésped.

- 1 Pare todos los dominios usando la opción `-a`.

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

- 2 Desenlace todos los dominios excepto el dominio `primary`.

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

Nota – Puede no ser posible desenlazar un dominio E/S si éste suministra servicios necesarios para el dominio de control. En esta situación, omite este paso.

- 3 Destruya todos los dominios excepto el dominio `primary`.

```
primary# ldm remove-domain -a
```

▼ Quite todas las configuraciones de los dominios lógicos

- 1 Enumere todas las configuraciones de dominio lógico guardadas en el procesador de servicios (SP).

```
primary# ldm list-config
```

- 2 Elimine todas las configuraciones (*config-name*) anteriormente guardadas al SP excepto la configuración `factory-default`.

Use el siguientes comando para cada una de estas configuraciones:

```
primary# ldm rm-config config-name
```

Después de haber eliminado todas las configuraciones anteriormente guardadas en el SP, el dominio `factory-default` es el siguiente dominio que se debe usar cuando el dominio de control (`primary`) se reinicia.

▼ Restablezca la configuración predeterminada de fábrica.

- 1 Seleccione la configuración predeterminada de fábrica.

```
primary# ldm set-config factory-default
```

- 2 Pare el dominio de control.

```
primary# shutdown -i1 -g0 -y
```

- 3 Apague y encienda el sistema para cargar la configuración predeterminada de fábrica.

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

▼ Deshabilite el Administrador de Dominios lógicos

- Deshabilite el Administrador de Dominios lógicos desde el dominio de control.

```
primary# svcadm disable ldmd
```

Nota – La deshabilitación del Administrador de Dominios lógicos no para los dominios en ejecución, pero sí que deshabilita la posibilidad de crear nuevos dominios, cambiar la configuración de dominios existentes, o efectuar un seguimiento del estado de los dominios.



Precaución – Si deshabilita el Administrador de Dominios lógicos, se deshabilitan algunos servicios, como el informe de error o la administración de energía. En caso de informe de errores, si está en la configuración `factory-default`, puede reiniciar el dominio de control para restablecer el informe de errores. En cualquier caso, no es así con el administrador de energía. Además, algunas herramientas de administración o seguimiento del sistema se basan en el Administrador de Dominios lógicos.

▼ Eliminación del Administrador de Dominios lógicos

Después restaurar la configuración predeterminada de fábrica y deshabilitar el Administrador de Dominios lógicos, puede eliminar el software del Administrador de Dominios lógicos.

- **Quite el software del Administrador de Dominios lógicos.**

```
primary# pkgrm SUNWldm SUNWldmp2v
```

Nota – Si elimina el Administrador de Dominios lógicos antes de restaurar la configuración predeterminada de fábrica, puede restaurar la configuración predeterminada de fábrica desde el procesador de servicios tal y como se muestra en el siguiente procedimiento.

▼ **Restablezca la configuración predeterminada de fábrica desde el procesador de servicios.**

Si elimina el Administrador de Dominios lógicos antes de restaurar la configuración predeterminada de fábrica, puede restaurar la configuración predeterminada de fábrica desde el procesador de servicios.

- 1 **Restablezca la configuración predeterminada de fábrica desde el procesador de servicios.**

```
-> set /HOST/bootmode config=factory-default
```

- 2 **Apague y encienda el sistema para cargar la configuración predeterminada de fábrica.**

```
-> reset /SYS
```


Seguridad

Este capítulo describe algunas de las características de seguridad que puede habilitar en el sistema Dominios lógicos.

Este capítulo trata sobre los siguientes temas:

- “Autorización de Administrador de Dominios lógicos” en la página 41
- “Creación de la autorización y perfiles y asignación de funciones para las cuentas de usuario” en la página 42
- “Configuración de RBAC para el acceso de la consola huésped” en la página 45
- “Habilitación y uso de auditoría del BSM” en la página 46

Autorización de Administrador de Dominios lógicos

La autorización para el Administrador de Dominios lógicos tiene dos niveles:

- Lectura – le permite ver, pero no modificar la configuración.
- Lectura y escritura – le permite ver y cambiar la configuración.

Los cambios no se realizan en el SO de Solaris de Oracle, sino que son agregados al archivo de autorización por la secuencia de comandos del paquete `postinstall` cuando se instala el Administrador de Dominios lógicos. De manera parecida, las entradas de autorización son eliminadas por la secuencia de comandos del paquete `preremove`.

La siguiente tabla indica los subcomandos `ldm` subcomandos con la correspondiente autorización de usuario que es necesaria para realizar los comandos.

TABLA 3-1 Los subcomandos `ldm` y autorizaciones de usuario

Subcomando <code>ldm</code> ¹	Autorización del usuario
<code>add -*</code>	<code>solaris.ldoms.write</code>

¹ Se refiere a todos los recursos que puede agregar, enumerar, eliminar o fijar.

TABLA 3-1 Los subcomandos ldm y autorizaciones de usuario (Continuación)

Subcomando ldm ¹	Autorización del usuario
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 Se refiere a todos los recursos que puede agregar, enumerar, eliminar o fijar.

Creación de la autorización y perfiles y asignación de funciones para las cuentas de usuario

Puede configurar autorizaciones y perfiles y asignar funciones para las cuentas de usuario usando el control de acceso basado en funciones (RBAC) del SO de Solaris de Oracle adaptado para el Administrador de Dominios lógicos. Consulte [Solaris 10 System Administrator Collection \(http://docs.sun.com/app/docs/coll/47.16\)](http://docs.sun.com/app/docs/coll/47.16) para más información sobre RBAC.

La autorización para el Administrador de Dominios lógicos tiene dos niveles:

- Leer - le permite ver, pero no modificar la configuración.
- Leer y escribir – le permite ver y cambiar la configuración.

A continuación se indican las entradas Dominios lógicos automáticamente agregadas al archivo SO de Solaris de Oracle `/etc/security/auth_attr`:

- `solaris.ldoms:::LDom administration::`
- `solaris.ldoms.grant:::Delegate LDom configuration::`
- `solaris.ldoms.read:::View LDom configuration::`
- `solaris.ldoms.write:::Manage LDom configuration::`

Administración de las autorizaciones de usuario

▼ Agregación de una autorización para un usuario

Use los siguientes pasos según sea necesario para agregar autorizaciones en el archivo `/etc/security/auth_attr` para los usuarios Administrador de Dominios lógicos. Dado que el superusuario ya tiene la autorización `solaris.*`, el superusuario ya tiene permiso para las autorizaciones `solaris.ldoms.*`.

- 1 Cree una cuenta de usuario local para cada usuario que necesite autorización para usar los subcomandos `ldm(1M)`.

Nota – Para agregar una autorización de Administrador de Dominios lógicos a un usuario, debe crearse una cuenta local (no LDAP) para dicho usuario. Consulte el [SO de Solaris de Oracle System Administrator Collection \(http://docs.sun.com/app/docs/coll/47.16\)](http://docs.sun.com/app/docs/coll/47.16) para más detalles.

- 2 Realice una de las siguientes operaciones dependiendo de a qué subcomandos `ldm(1M)` desea que pueda acceder el usuario.

Véase [Tabla 3-1](#) para una lista de los comandos `ldm(1M)` y las respectivas autorizaciones de usuario.

- Agregue una autorización de sólo lectura para un usuario que usa el comando `usermod(1M)`.

```
# usermod -A solaris.ldoms.read username
```
- Agregue una autorización de lectura y escritura para un usuario que usa el comando `usermod(1M)`.

```
# usermod -A solaris.ldoms.write username
```

▼ Borrado de todas las autorizaciones para un usuario

- Borrado de todas las autorizaciones para una cuenta de usuario local (la única opción posible).

```
# usermod -A '' username
```

Administración de los perfiles de usuario

El paquete `SUNWldm` agrega dos perfiles RBAC definidos por sistema en el archivo `/etc/security/prof_attr` para su uso en el acceso autorizado al Administrador de Dominios lógicos por no superusuarios. Los dos perfiles específicos de Dominios lógicos son:

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

El paquete SUNWldm también define el siguiente atributo de ejecución que está asociado con el perfil de administración LDOMs:

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

Uno de los anteriores perfiles puede ser asignado a la cuenta de usuario usando el siguiente procedimiento.

▼ **Agregado de un perfil para un usuario**

Los usuarios a los que se ha asignado directamente el perfil de administración LDOMs deben invocar un comando de intérprete de perfil para ejecutar el comando `ldm` con los atributos de seguridad. Para más información, vea el [Solaris de Oracle 10 System Administrator Collection](#).

- **Agregue un perfil administrativo a una cuenta de usuario local; por ejemplo, administración de LDOMs.**

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

▼ **Borrado de todos los perfiles para un usuario**

- **Borrado de todos los perfiles para una cuenta de usuario local (la única opción posible).**

```
# usermod -P '' username
```

Asignación de funciones a usuarios

La ventaja de usar este procedimiento es que sólo un usuario al que ha sido asignada una función puede asumir dicha función. Al asumir una función, es necesaria la contraseña si se ha dado la contraseña a la función. Esto ofrece dos capas de seguridad. Si no se ha asignado una función al usuario, entonces el usuario no puede asumir la función (realizando el comando `su role-name`) incluso si el usuario tiene la contraseña correcta.

▼ **Creación de una función y asignación de la función a un usuario**

- 1 **Cree una función.**

```
# roleadd -P "LDoms Review" ldm_read
```

- 2 **Asigne una contraseña a la función.**

```
# passwd ldm_read
```

- 3 **Asigne una función a un usuario.**

Por ejemplo, `user_1`.

```
# useradd -R ldm_read user_1
```

4 Asigne una contraseña al usuario (user_1).

```
# passwd user_1
```

5 Asigne acceso sólo a la cuenta user_1 para convertirse en cuenta ldm_read.

```
# su user_1
```

6 Introduzca la contraseña de usuario cuando se solicite.**7 Compruebe el id de usuario y acceda a la función ldm_read.**

```
$ id
uid=nn(user_1) gid=nn(group-name)
$ roles
ldm_read
```

8 Ofrezca acceso al usuario a los subcomandos ldm que tienen autorización de lectura.

```
# su ldm_read
```

9 Introduzca la contraseña de usuario cuando se solicite.**10 Introduzca el comando id para mostrar el usuario.**

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

Configuración de RBAC para el acceso de la consola huésped

El daemon `vntsd` ofrece una propiedad SMF llamada `vntsd/authorization`. Esta propiedad puede configurarse para habilitar la comprobación de autorización de usuarios y funciones para una consola de dominio o un grupo de consola. Para habilitar la comprobación de autorización, use el comando `svccfg` para fijar el valor de esta propiedad en `true`. Mientras esta opción está habilitada, `vntsd` escucha y acepta conexiones sólo en `localhost`. Si la propiedad `listen_addr` especifica una dirección IP alternativa cuando `vntsd/authorization` está habilitado, `vntsd` ignora las direcciones IP alternativas y continúa escuchando sólo en `localhost`.

De manera predeterminada, una autorización para acceder a todas las consolas huésped se agrega a la base de datos `auth_attr`, cuando el servicio `vntsd` está habilitado.

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

El superusuario puede usar el comando `usermod` para asignar las autorizaciones necesarias para los otros usuarios o funciones. Esto permite que sólo el usuario o función que tienen las autorizaciones necesarias puedan acceder a determinada consola de dominio o grupos de consola.

El siguiente ejemplo otorga al usuario `terry` la autorización para acceder a todas las consolas de dominio:

```
# usermod -A "solaris.vntsd.consoles" terry
```

El siguiente ejemplo agrega una nueva autorización para una consola de dominio específica con el nombre `ldg1` y asigna dicha autorización a un usuario `sam`:

1. Agregue la nueva entrada de autorización al archivo `auth_attr` para el dominio `ldg1`.

```
solaris.vntsd.console-ldg1::Access Specific LDoms Guest Console::
```

2. Asigne esta autorización al usuario `sam`:

```
# usermod -A "solaris.vntsd.console-ldg1" sam
```

Para más información sobre autorizaciones y RBAC, véase [System Administration Guide: Security Services](#).

Habilitación y uso de auditoría del BSM

El Administrador de Dominios lógicos usa la capacidad de auditoría del módulo de seguridad básico (BSM) del SO de Solaris de Oracle. La auditoría del BSM ofrece los medios para examinar la historia de acciones y eventos en el dominio de control para determinar qué ha sucedido. El historial se guarda en un registro de que lo se ha realizado, cuándo ha sido realizado, por quién y a qué ha afectado.

Para habilitar y deshabilitar la capacidad de auditoría, use los comandos [bsmconv\(1M\)](#) y [bsmunconv\(1M\)](#) del SO de Solaris de Oracle. Esta sección también incluye las tareas que muestran cómo comprobar la capacidad de auditoría, imprimir la salida de auditoría y girar los registros de auditoría. Puede encontrar más información sobre la auditoría de BSM en Solaris 10 *Guía de administración del sistema: Servicios de seguridad*.

▼ Habilitación de auditoría de BSM

1. Agregue `vs` en la línea `flags`: del archivo `/etc/security/audit_control`.

2. Ejecute el comando `bsmconv(1M)`.

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

Para más información sobre este comando, Consulte la página de comandos [man bsmconv\(1M\)](#).

3. Reinicie el SO de Solaris de Oracle para que la auditoría se realice.

▼ Compruebe que la auditoría BSM está habilitada.

- 1 Escriba el comando siguiente.

```
# auditconfig -getcond
```

- 2 Compruebe que `audit condition = auditing` aparece en la salida.

▼ Deshabilitación de la auditoría BSM

- 1 Ejecute el comando `bsmunconv` para deshabilitar la auditoría de BSM.

```
# /etc/security/bsmunconv
```

Para más información sobre este comando, vea la página de comandos [man bsmunconv\(1M\)](#).

- 2 Reinicie el SO de Solaris de Oracle para que se realice la deshabilitación de la auditoría.

▼ Impresión de una salida de auditoría

- Use una de las siguientes opciones para imprimir la salida de la auditoría de BSM:

- Use los comandos para imprimir la salida de la auditoría [auditreduce\(1M\)](#) y [praudit\(1M\)](#).

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

- Use el comando `praudit -x` para imprimir la salida XML.

▼ Giro de los registros de la auditoría

- Use el comando `audit -n` para girar los registros de la auditoría.

Configuración de servicios y el dominio de control

Este capítulo describe los procedimientos necesarios para configurar los servicios predeterminados y el dominio de control.

También puede usar el Asistente para la configuración de Oracle VM Server for SPARC para configurar los dominios lógicos y servicios. Véase [Apéndice B, “Asistente para la configuración de Oracle VM Server for SPARC”](#).

Este capítulo trata sobre los siguientes temas:

- “Mensajes de salida” en la página 49
- “Creación de servicios predeterminados” en la página 50
- “Configuración inicial del dominio de control” en la página 51
- “Reinicio para el uso de Dominios lógicos” en la página 52
- “Habilitación de las funciones de red entre dominio de control/servicio y otros dominios” en la página 53
- “Habilitación del daemon del servidor del terminal de red virtual” en la página 54

Mensajes de salida

Desde la versión 2.0 del Oracle VM Server for SPARC, si un recurso no puede configurarse dinámicamente en el dominio de control, lo es mejor comenzar primero una reconfiguración retrasada. La reconfiguración retrasada pospone las actividades de configuración hasta después del reinicio del dominio de control.

Recibe el siguiente mensaje cuando inicia una configuración retrasada en el dominio 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.
```

Recibe el siguiente aviso en cada operación posterior en el dominio primary hasta el reinicio:

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.

Creación de servicios predeterminados

Los siguientes servicios de dispositivo virtual deben crearse para usar el dominio de control como dominio de servicio y crear dispositivos virtuales para otros dominios:

- vcc – Servicio de concentrador de consola virtual
- vds – Servidor de disco virtual
- vsw – Servicio de conmutador virtual

▼ Creación de servicios predeterminados

- 1 Cree un servicio de concentrador de consola virtual (vcc) para el uso por el daemon del servidor del terminal de red virtual (vntsd) y como concentrador para todas las consolas de dominio lógico.**

Por ejemplo, el siguiente comando agregaría un servicio de concentrador de consola virtual (primary-vcc0) con un rango de puerto de 5000 a 5100 al dominio de control (primary).

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

- 2 Cree un servidor de disco virtual (vds) para permitir la importación de discos virtuales en un dominio lógico.**

Por ejemplo, el siguiente comando agrega un servidor de disco virtual (primary-vds0) al dominio de control (primary).

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

- 3 Cree un servicio de conmutador virtual (vsw) para habilitar los servicios de red entre dispositivos de red virtual (vnet) en dominios lógicos.**

Asigne un adaptador de red compatible con GLDv3 al conmutador virtual si cada uno de los dominios lógicos necesitan comunicarse fuera del cuadro a través del conmutador virtual.

Por ejemplo, el siguiente comando agregaría un servicio de conmutador virtual (primary-vsw0) en el controlador del adaptador de red nxge0 al dominio de control (primary).

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

Este comando asigna automáticamente una dirección MAC al conmutador virtual. Puede especificar su propia dirección MAC como opción al comando `ldm add-vsw`. Sin embargo, en este caso, es responsabilidad suya asegurarse de que la dirección MAC especificada no crea conflictos con una dirección MAC ya existente.

Si el conmutador virtual que se agrega sustituye el adaptador físico subyacente como interfaz de la red primaria, se le debe asignar la dirección MAC del adaptador físico, de manera que el

servidor del protocolo de configuración dinámica de host (DHCP) asigne al dominio la misma dirección IP. Véase “[Habilitación de las funciones de red entre dominio de control/servicio y otros dominios](#)” en la página 53

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

4 Compruebe que se han creado los servicios usando el subcomando `list-services`.

La salida debe ser parecida a la siguiente:

```
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
```

Configuración inicial del dominio de control

Inicialmente, todos los recursos de sistema se asignan al dominio de control. Para permitir la creación de otros dominios lógicos, debe liberar algunos de estos recursos.

No intente usar la reconfiguración dinámica de memoria (DR) para realizar la configuración inicial del dominio de control. A pesar de que puede usar la DR de memoria para realizar esta configuración sin que sea necesario un reinicio, *no* se recomienda realizar esta operación. El enfoque de la DR de memoria puede tardar mucho tiempo (más que un reinicio) y puede incluso fallar. En cambio, use el comando `ldm start-reconf` para situar el dominio de control en modo de reconfiguración retrasada antes de cambiar la configuración de la memoria. Después, puede reiniciar el dominio de control después de haber completado todos los pasos de configuración.

▼ Configuración del dominio de control

Nota – Este procedimiento contiene ejemplos de recursos para configurar para el dominio de control. Estos números son sólo ejemplos, y los valores usados pueden no ser adecuados para su dominio de control.

1 Determine si posee los dispositivos criptográficos en el dominio de control.

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

2 Asigne los recursos criptográficos al dominio de control.

El siguiente ejemplo asigna un recurso criptográfico al dominio de control, `primary`. Esto deja los recursos criptográficos restantes a disposición del dominio huésped.

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

3 Asigne las CPU virtuales al dominio de control.

Por ejemplo, el siguiente comando asignaría 8 CPU virtuales al dominio de control, `primary`. Esto deja las CPU virtuales restantes a disposición del dominio huésped.

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

4 Inicie una reconfiguración retrasada en el dominio de control.

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

5 Asigne memoria al dominio de control.

Por ejemplo, el siguiente comando asignaría 4 gigabytes de memoria al dominio de control, `primary`. Esto deja la memoria restante a disposición del dominio huésped.

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

6 Agregue una configuración de equipo de dominio lógico al procesador de servicio (SP).

Por ejemplo, el siguiente comando agregaría una configuración llamada `initial`.

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

7 Compruebe que la configuración está preparada para el uso en el siguiente reinicio.

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

Este subcomando de lista muestra la configuración `initial` fijada que será usada cuando se apague y encienda.

Reinicio para el uso de Dominios lógicos

Debe reiniciar el dominio de control para que se efectúen los cambios y para que se liberen los recursos para el uso por parte de otros dominios lógicos.

▼ Reinicio

● Apague y reinicie el dominio de control.

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

Nota – Un reinicio o un apagado y posterior encendido crea una nueva instancia en la nueva configuración. Sólo un apagado y encendido inicia la configuración guardada en el procesador de servicios (SP), lo que después se refleja en la salida `list-config`.

Habilitación de las funciones de red entre dominio de control/servicio y otros dominios

De manera predeterminada, las funciones de red entre el dominio de control y los otros dominios en el sistema están deshabilitadas. Para habilitarlas, el dispositivo de conmutador virtual debe ser configurado como dispositivo de red. El conmutador virtual puede reemplazar el dispositivo físico subyacente (`nxge0` en este ejemplo) como interfaz primaria o ser configurado como interfaz de red adicional en el dominio.

Nota – Realice el siguiente procedimiento desde la consola del dominio de control, ya que el procedimiento puede interrumpir temporalmente la conectividad de la red al dominio.

▼ Configuración del conmutador virtual como interfaz primaria.

- 1 Imprima la información sobre la asignación de direcciones para todas las interfaces.

```
primary# ifconfig -a
```

- 2 Instale el conmutador virtual. En este ejemplo, `vsw0` es el conmutador virtual que se está configurando.

```
primary# ifconfig vsw0 plumb
```

- 3 (Opcional) Para obtener la lista de las instancias de conmutadores virtuales en un dominio, puede enumerarlas.

```
primary# /usr/sbin/dladm show-link | grep vsw
vsw0          type: non-vlan mtu: 1500      device: vsw0
```

- 4 Desinstale el dispositivo de red física asignado al conmutador virtual (`net-dev`), que es `nxge0` en este ejemplo.

```
primary# ifconfig nxge0 down unplumb
```

- 5 Para migrar las propiedades del dispositivo de red física (`nxge0`) al dispositivo de conmutador virtual (`vsw0`), realice una de las siguientes operaciones:

- Si los dispositivos de red están configurados usando una dirección IP estática, reutilice la dirección IP y la máscara de red de `nxge0` para `vsw0`.

```
primary# ifconfig vsw0 IP_of_nxge0 netmask netmask_of_nxge0 broadcast + up
```

- Si los dispositivos de red se han configurado usando DHCP, habilite DHCP para vsw0.

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

- 6 Realice las modificaciones de archivo de la configuración necesarias para hacer que este cambio sea permanente.

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

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

Nota – Si es necesario, también puede configurar el conmutador virtual y el dispositivo de red físico. En este caso, instale el conmutador virtual como en el paso 2, y no instale el dispositivo físico (salte el paso 4). Debe configurar el conmutador virtual con una dirección IP estática o con una dirección IP dinámica. Puede obtener una dirección IP dinámica del servidor DHCP. Para más información y un ejemplo de este caso, véase [“Configuración del conmutador virtual y dominio de servicio para NAT y enrutamiento”](#) en la página 124.

Habilitación del daemon del servidor del terminal de red virtual

Debe habilitar el daemon del servidor del terminal de red virtual (vntsd) para ofrecer acceso a la consola virtual de cada dominio lógico. Consulte la página de comandos [man vntsd\(1M\)](#) para más información sobre cómo usar este daemon.

▼ Habilite el daemon del servidor del terminal de red virtual

Nota – Asegúrese de que ha creado el servicio predeterminado vconscon (vcc) en el dominio de control antes de habilitar vntsd. Véase [“Creación de servicios predeterminados”](#) en la página 50 para más información.

- 1 Use el comando [svcadm\(1M\)](#) para habilitar el daemon del servidor del terminal de red virtual, [vntsd\(1M\)](#).

```
primary# svcadm enable vntsd
```

- 2 Use el comando [svcs\(1\)](#) para comprobar que está habilitado el daemon vntsd.

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

Configuración de los dominios huésped

Este capítulo describe los procedimientos necesarios para configurar los dominios huésped.

También puede usar el Asistente para la configuración de Oracle VM Server for SPARC para configurar los dominios lógicos y servicios. Véase [Apéndice B, “Asistente para la configuración de Oracle VM Server for SPARC”](#).

Este capítulo trata sobre los siguientes temas:

- “Creación e inicio de dominio huésped” en la página 55
- “Instalación del SO de Solaris de Oracle en un dominio huésped” en la página 58

Creación e inicio de dominio huésped

El dominio huésped debe ejecutar un sistema operativo que entienda tanto la plataforma sun4v como los dispositivos virtuales presentados por el hypervisor. Actualmente, esto significa que debe ejecutar al menos el So 10 11/06 de Solaris de Oracle. Si ejecuta el SO 10 9/10 Solaris de Oracle le ofrece todas las características de Oracle VM Server for SPARC 2.0. Véase la *Notas sobre la versión del servidor Oracle VM para SPARC 2.0* para cualquier parche específico que pueda ser necesario. Una vez haya creado los servicios predeterminados y reubicado los recursos desde el dominio de control, puede crear e iniciar un dominio huésped.

▼ Creación e inicio de un dominio huésped

1 Cree un dominio lógico.

Por ejemplo, el siguiente comando crearía un dominio huésped llamado `ldg1`.

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

2 Agregue CPU al dominio huésped.

Por ejemplo, el siguiente comando agregaría ocho CPU virtuales al dominio huésped `ldg1`.

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

3 Agregue memoria al dominio huésped.

Por ejemplo, el siguiente comando agregaría 2 gigabytes de memoria al dominio huésped `ldg1`.

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

4 Agregue el dispositivo de red virtual al dominio huésped.

Por ejemplo, el siguiente comando agregaría un dispositivo de red virtual con estas especificaciones al dominio huésped `ldg1`.

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

Donde:

- `vnet1` es un nombre de interfaz único, asignado a la instancia del dispositivo de red virtual como referencia en los siguientes subcomandos `set-vnet` o `remove-vnet`.
- `primary-vsw0` es el nombre de un servicio de red existente (conmutador virtual) al que conectarse.

Nota – Los pasos 5 y 6 son instrucciones simplificadas para agregar un dispositivo del servidor de disco virtual (`vdsdev`) al dominio primario y un disco virtual (`vdisk`) al dominio huésped. Para saber cómo pueden usarse los volúmenes ZFS y los sistemas de archivos como discos virtuales, véase [“Exportación de un volumen ZFS como un disco de segmento único” en la página 92](#) y [“Uso de ZFS con discos virtuales” en la página 102](#).

5 Especifique el dispositivo que debe ser exportado por el servidor de disco virtual como disco virtual al dominio huésped.

Puede exportar un disco físico, un segmento de disco, volúmenes o archivo como dispositivo en bloque. Los siguientes ejemplos muestran un disco físico y un archivo.

- **Ejemplo de disco físico.** El primer ejemplo agrega un disco físico con estas especificaciones.

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

Donde:

- `/dev/dsk/c2t1d0s2` es el nombre de la ruta del dispositivo físico actual. Cuando se agrega un dispositivo, el nombre de la ruta debe asociarse con el nombre del dispositivo.
- `vol1` es un nombre único que debe especificar para el dispositivo que se agrega al servidor de disco virtual. El nombre del volumen debe ser único en esta instancia de servidor de disco virtual, ya que este nombre es exportado por el servidor de disco virtual a los clientes para el agregado. Cuando agregue un dispositivo, el nombre del producto debe asociarse con el nombre de la ruta del dispositivo actual.
- `primary-vds0` es el nombre del servidor de disco virtual al que agregar este dispositivo.

- **Ejemplo de archivo.** Este segundo ejemplo se exporta un archivo como dispositivo en bloque.

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

Donde:

- *backend* es el nombre de la ruta del archivo actual exportado como dispositivo en bloque. Cuando se agrega un dispositivo, el componente posterior debe asociarse con el nombre del dispositivo.
- *vol1* es un nombre único que debe especificar para el dispositivo que se agrega al servidor del disco virtual. El nombre del volumen debe ser único en esta instancia de servidor de disco virtual, ya que este nombre es exportado por el servidor de disco virtual a los clientes para el agregado. Cuando agregue un dispositivo, el nombre del producto debe asociarse con el nombre de la ruta del dispositivo actual.
- *primary-vds0* es el nombre del servidor de disco virtual al que agregar este dispositivo.

6 Agregue el disco virtual al dominio huésped.

El siguiente ejemplo agrega un disco virtual al dominio huésped *ldg1*.

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

Donde:

- *vdisk1* es el nombre del disco virtual.
- *vol1* es el nombre del volumen existente al que conectarse.
- *primary-vds0* es el nombre del servidor de disco virtual existente al que conectarse.

Nota – Los discos virtuales son dispositivos de bloque genéricos que se asocian con diferentes tipos de dispositivos físicos, volúmenes o archivos. Un disco virtual no es sinónimo de un disco SCSI y por lo tanto, excluye el id de destino en la etiqueta del disco. Los discos virtuales en un dominio lógico tienen el siguiente formato: *cNdNsN*, donde *cN* es el controlador virtual, *dN* es el número de disco virtual, y *sN* es el segmento.

7 Fije las variables `auto-boot?` y `boot-device` para el dominio huésped.

El primer ejemplo de comando fija `auto-boot?` en `true` para el dominio huésped *ldg1*.

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

El segundo ejemplo de comando fija `boot-device` en `vdisk1` para el dominio huésped *ldg1*.

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

8 Enlace los recursos al dominio huésped *ldg1* y después efectúe una lista del dominio para comprobar que está enlazado.

```
primary# ldm bind-domain ldg1
primary# ldm list-domain ldg1
```

NAME	STATE	FLAGS	CONS	VCPU	MEMORY	UTIL	UPTIME
ldg1	bound	-----	5000	8	2G		

- 9 Para encontrar el puerto de consola para el dominio huésped, puede mirar en la salida del anterior subcomando `list-domain`.

En el encabezado CONS puede ver que el huésped del dominio lógico 1 (ldg1) tiene la salida de consola enlazada al puerto 5000.

- 10 Conecte a la consola de un dominio huésped desde otro terminal iniciando sesión en el dominio de control y conectando directamente al puerto de la consola en el host local.

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

- 11 Inicie el dominio huésped ldg1.

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

Instalación del SO de Solaris de Oracle en un dominio huésped

Esta sección ofrece las instrucciones necesarias sobre las diferentes maneras de instalar el SO de Solaris de Oracle en un dominio huésped.



Precaución – No desconecte la consola virtual durante la instalación del SO de Solaris de Oracle.

▼ Instalación del SO de Solaris de Oracle en el dominio huésped desde un DVD

- 1 Introduzca el DVD de SO 10 de Solaris de Oracle en la unidad de DVD.
- 2 Pare el daemon de administración del volumen, `vol(1M)`, en el dominio `primary`.

```
primary# svcadm disable volfs
```

- 3 Pare y desenlace el dominio huésped (ldg1).

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

4 Agregue el DVD con los medios DVD-ROM como volumen secundario y disco virtual.

El siguiente ejemplo utiliza `c0t0d0s2` como unidad DVD en la que residen los medios de Solaris de Oracle, `dvd_vol@primary-vds0` como volumen secundario, y `vdisk_cd_media` como disco virtual.

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

5 Compruebe que el DVD se agrega como volumen secundario y disco virtual.

```
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 -----
...
DISK
  NAME          VOLUME          TOUT DEVICE  SERVER
  vdisk1        vol1@primary-vds0
  vdisk_cd_media  dvd_vol@primary-vds0
...

```

6 Enlace e inicie el dominio huésped (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 Muestre los alias del dispositivo en el PROM OpenBoot del cliente.

En este ejemplo, vea los alias del dispositivo para `vdisk_cd_media`, que es el DVD de Solaris de Oracle y `vdisk1`, que es un disco virtual en el que puede instalar el SO de Solaris de Oracle.

```
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 En la consola del dominio huésped, inicie desde `vdisk_cd_media (disk@1)` en el segmento 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 Continúe con el menú de instalación de SO de Solaris de Oracle.

▼ Instale el SO de Solaris de Oracle en el dominio huésped desde un archivo ISO de Solaris de Oracle

1 Pare y desenchufe el dominio huésped (ldg1).

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

2 Agregue el archivo ISO Solaris de Oracle como volumen secundario y disco virtual.

El siguiente ejemplo utiliza solarisdvd.iso como archivo ISO de Solaris de Oracle, iso_vol@primary-vds0 como volumen secundario, y vdisk_iso como disco virtual:

```
primary# ldm add-vdsdev /export/solarisdvd.iso iso_vol@primary-vds0
primary# ldm-vdisk vdisk vdisk_iso iso_vol@primary-vds0 ldg1
```

3 Compruebe que el archivo ISO de Solaris de Oracle se agrega como volumen secundario y disco virtual.

```
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      iso_vol /export/solarisdvd.iso
....
-----
NAME          STATE  FLAGS  CONS  VCPU  MEMORY  UTIL  UPTIME
ldg1         inactive -----  60    6G
...
DISK
NAME          VOLUME  TOUT  DEVICE  SERVER
vdisk1       vol1@primary-vds0
vdisk_iso    iso_vol@primary-vds0
....
```

4 Enlace e inicie el dominio huésped (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 Muestre los alias del dispositivo en el PROM OpenBoot del cliente.

En este ejemplo, véanse los alias del dispositivo para `vdisk_iso`, que es la imagen ISO de Solaris de Oracle y `vdisk_install`, que es el espacio de disco.

```
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 En la consola del dominios huésped, inicie desde `vdisk_iso` (`disk@1`) en el segmento `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 Continúe con el menú de instalación de SO de Solaris de Oracle.

▼ Realización de una operación JumpStart en un dominio huésped

Este procedimiento describe cómo realizar una operación JumpStart en un dominio huésped. Este procedimiento sigue el procedimiento JumpStart normal, pero describe un nombre de dispositivo de disco diferente para usar en el perfil JumpStart para el dominio huésped. Véase la [Guía de instalación de Oracle Solaris 10 9/10: Instalaciones JumpStart personalizadas y avanzadas](#).

Los nombres del dispositivo de disco virtual en un dominio lógico son diferentes de los nombres de un dispositivo de disco físico. Los nombres del dispositivo del disco virtual no contienen un id de destino (`tN`).

En vez del formato habitual `cNtNdNsN`, los nombres de dispositivo de disco virtual usan el formato `cNdNsN`. `cN` es el controlador virtual, `dN` es el número del disco virtual, y `sN` es el número de segmento.

- **Modifique el perfil JumpStart para reflejar este cambio.**

Un disco virtual puede aparecer como disco completo o como disco de segmento único. El SO de Solaris de Oracle puede instalarse en un disco completo usando un perfil JumpStart normal que especifica particiones múltiples. Un disco de segmento único tiene una sola partición, `s0`, que utiliza todo el disco. Para instalar el SO de Solaris de Oracle en un solo disco, debe usar un perfil que tenga una sola partición (`/`) que usa todo el disco. No puede definir ninguna otra

partición, como intercambio. Para más información sobre discos completos y discos de un solo segmento, véase [“Apariencia del disco virtual” en la página 85](#).

- **Perfil JumpStart para la instalación de un sistema de archivos de raíz UFS.**

Véase *Guía de instalación de Oracle Solaris 10 9/10: Instalaciones JumpStart personalizadas y avanzadas*.

Perfil UFS normal

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

Perfil UFS actual para la instalación de un dominio en un disco completo

```
filesys c0d0s0 free /  
filesys c0d0s1 2048 swap  
filesys c0d0s5 120 /spare1  
filesys c0d0s6 120 /spare2
```

Perfil UFS actual para la instalación de un dominio en un disco de segmento único

```
filesys c0d0s0 free /
```

- **Perfil JumpStart para la instalación de un sistema de archivos de raíz ZFS.**

Véase [Capítulo 9, “Instalación de una agrupación raíz ZFS con JumpStart” de *Guía de instalación de Oracle Solaris 10 9/10: Instalaciones JumpStart personalizadas y avanzadas*](#).

Perfil ZFS normal

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

Perfil ZFS real para la instalación de un dominio

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

Configuración de dominios E/S

Este capítulo describe los dominios E/S y cómo configurarlos en un entorno Dominios lógicos.

Este capítulo trata sobre los siguientes temas:

- “Información general sobre los dominios E/S” en la página 63
- “Asignación de buses PCIe” en la página 64
- “Asignación de dispositivos de punto final PCIe” en la página 69

Información general sobre los dominios E/S

Un dominio E/S tiene la propiedad directa y el acceso directo a los dispositivos de E/S físicos. Puede crearse asignando un bus PCI EXPRESS (PCIe) o un dispositivo de punto final PCIe a un dominio. Use el comando `ldm add-io` para asignar un bus o un dispositivo a un dominio.

Puede querer configurar dominios E/S por las siguientes razones:

- Un dominio E/S tiene acceso directo a un dispositivo de E/S físico, que evita la carga adicional indirecta de rendimiento que está asociado con la E/S virtual. Como resultado, el rendimiento de E/S en un dominio E/S coincide más con el rendimiento E/S en un sistema de bajo nivel.
- Un dominio E/S puede alojar servicios E/S virtuales que pueden ser usados por otros dominios huésped.

Para más información sobre cómo configurar los dominios E/S, vea la siguiente información:

- “Asignación de buses PCIe” en la página 64
- “Asignación de dispositivos de punto final PCIe” en la página 69

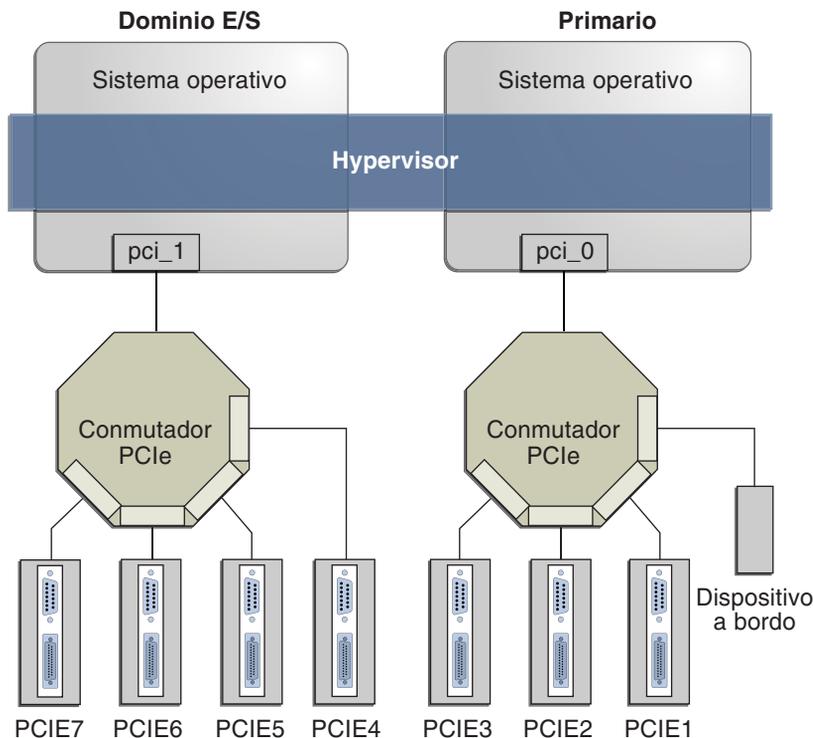
Nota – No puede migrar un dominio E/S que está configurado con dispositivos de punto final PCIe. Para más información sobre las limitaciones de migración, véase [Capítulo 9, “Migración de dominios”](#).

Asignación de buses PCIe

Puede usar el software del Oracle VM Server for SPARC para asignar todo un bus PCIe (también conocido como *complejo de raíz*) a un dominio. Todo el bus PCIe consiste en el mismo bus PCIe y todos los conmutadores y dispositivos PCI. Los buses PCIe que están presentes en un servidor se identifican con nombres como `pci@400` (`pci_0`). Un dominio E/S que está configurado con todo un bus PCIe también se conoce como *dominio raíz*.

El siguiente diagrama muestra un sistema que tiene dos buses PCIe (`pci_0` y `pci_1`). Cada bus se asigna a un dominio diferente. Por lo tanto, el sistema se configura con dos dominios E/S.

FIGURA 6-1 Asignación de un bus PCIe a un dominio E/S.



El número máximo de dominios E/S que puede crear con buses PCIe depende del número de buses PCIe que están disponibles en el servidor. Por ejemplo, si usa un servidor Sun SPARC Enterprise T5440, puede tener hasta cuatro dominios E/S.

Nota – Algunos servidores UltraSPARC de SUN sólo tiene un bus PCIe. En estos casos, puede crear un dominio E/S asignando un dispositivo de punto final PCIe (o E/S directa asignable) a un dominio. Véase “[Asignación de dispositivos de punto final PCIe](#)” en la [página 69](#) Si el sistema tiene una unidad de interfaz de red (NIU), también puede asignar una NIU a un dominio para crear un dominio E/S.

Quando asigna un bus PCIe a un dominio E/S, todos los dispositivos en ese bus son propiedad del dominio E/S. No se permite asignar cualquiera de los dispositivos de punto final de PCIe en ese bus a otros dominios. Sólo los dispositivos de punto final PCIe en los buses PCIe que están asignados al dominio `primary` pueden ser asignados a otros dominios.

Quando un servidor se configura inicialmente en un entorno Dominios lógicos o está usando la configuración `factory-default`, el dominio `primary` tiene acceso a todos los recursos de dispositivos físicos. Esto significa que el dominio `primary` es el único dominio E/S configurado en el sistema y que posee todos los buses PCIe.

▼ Creación de un dominio E/S asignando un bus PCIe

Este procedimiento de ejemplo muestra cómo crear un nuevo dominio E/S desde una configuración inicial donde varios buses son propiedad del dominio `primary`. De manera predeterminada el dominio `primary` posee todos los buses presentes en el sistema. Este ejemplo es para un servidor SPARC Enterprise T5440 de Sun. Este procedimiento también puede ser usado en otros servidores. Las instrucciones para los diferentes servidores pueden variar ligeramente de éstas, pero puede obtener los principios básicos de este ejemplo.

Primero, debe conservar el bus que tiene el disco de inicio del dominio `primary`. Después, elimine otro bus del dominio `primary` y asígnelo a otro dominio.



Precaución – Todos los discos internos en los servidores admitidos están conectados a un solo bus PCIe. Si un dominio se inicia desde un disco interno, no quite ese bus del dominio. Asimismo, asegúrese que no está eliminando un bus con dispositivos (como puertos de red) usados por un dominio. Si quita el bus equivocado, el dominio podría no poder acceder a los dispositivos necesarios y podría quedar no utilizable. Para eliminar un bus que tiene dispositivos usados por un dominio, reconfigure ese dominio para usar dispositivos de otros buses. Por ejemplo, quizás sea necesario reconfigurar el dominio para que use un puerto de red integrado o una tarjeta PCIe de una ranura PCIe diferente.

En este ejemplo, el dominio `primary` sólo usa un grupo ZFS (`rpool (c0t1d0s0)`) y la interfaz de red (`nxge0`). Si el dominio `primary` usa más dispositivos, repita los pasos 2-4 para cada dispositivo para asegurarse de que ninguno está ubicado en el bus que se ha eliminado.

1 Compruebe que el dominio `primary` tiene más de un bus PCIe.

```
primary# ldm list-io
IO                PSEUDONYM        DOMAIN
--                -
pci@400           pci_0             primary
pci@500           pci_1             primary
pci@600           pci_2             primary
pci@700           pci_3             primary

PCIe              PSEUDONYM        STATUS  DOMAIN
----              -
pci@400/pci@0/pci@d MB/PCIE0         EMP     -
pci@400/pci@0/pci@c MB/PCIE1         OCC     primary
pci@400/pci@0/pci@l MB/HBA           OCC     primary
pci@500/pci@0/pci@d MB/PCIE4         EMP     -
pci@500/pci@0/pci@9 MB/PCIE5         EMP     -
pci@500/pci@0/pci@c MB/NET0          OCC     primary
pci@600/pci@0/pci@c MB/PCIE2         OCC     primary
pci@600/pci@0/pci@9 MB/PCIE3         OCC     primary
pci@700/pci@0/pci@c MB/PCIE6         OCC     primary
pci@700/pci@0/pci@9 MB/PCIE7         EMP     -
```

2 Determine la ruta del dispositivo del disco de inicio, que es necesario guardar.

- Para los archivos de sistema UFS, ejecute el comando `df /` para determinar la ruta del dispositivo del disco de inicio.

```
primary# df /
/                (/dev/dsk/c0t1d0s0 ): 1309384 blocks  457028 files
```

- Para los sistemas de archivos ZFS, primero ejecute el comando `df /` para determinar el nombre del grupo, y después ejecute el comando `zpool status` para determinar la ruta del dispositivo del disco de inicio.

```
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 Determine el dispositivo físico con el que está enlazado el dispositivo en bloque.

El siguiente ejemplo usa un dispositivo de bloques `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
```

En este ejemplo, el dispositivo físico para el disco de inicio del dominio `primary` está conectado al bus `pci@400`, que corresponde a la enumeración anterior de `pci_0`. Esto significa que *no se puede asignar* `pci_0` (`pci@400`) a otro dominio.

4 Determine la interfaz de red usada por el sistema.

```
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
```

Las interfaces que están en estado `unknown` no se configuran, de manera que no se usan. En este ejemplo, se usa la interfaz `nxge0`.

5 Determine el dispositivo físico con el que está enlazado la interfaz de red.

El siguiente comando usa la interfaz de red `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
```

En este ejemplo, el dispositivo físico para la interfaz de red usada por el dominio `primary` está bajo el bus `pci@500`, que corresponde a la enumeración anterior de `pci_1`. Así pues, los otros dos buses, `pci_2` (`pci@600`) y `pci_3` (`pci@700`), pueden ser asignados de manera segura porque no son usados por el dominio `primary`.

Si la interfaz de red usada por el dominio `primary` estaba en un bus que desea asignar a otro dominio, el dominio `primary` deberá reconfigurarse para usar una interfaz de red diferente.

6 Elimine los buses que contiene el disco de inicio o la interfaz de red del dominio `primary`.

En este ejemplo, se están eliminando el bus `pci_2` y el bus `pci_3` del dominio `primary`. Puede ver un mensaje desde el comando `ldm` indicando que el dominio `primary` está entrando en modo de reconfiguración retrasada.

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

7 Guarde esta configuración en el procesador de servicio

En este ejemplo, la configuración es `io-domain`.

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

Esta configuración, `io-domain`, también se fija como la siguiente configuración que se debe usar después del reinicio.

Nota – Actualmente, existe un límite de 8 configuraciones que pueden guardarse en el SP, sin incluir la configuración predeterminada de fábrica.

8 Reinicie el dominio primary para que se realice el cambio.

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

9 Pare el dominio en el que desea agregar el bus PCIe.

El siguiente ejemplo para el dominio ldg1:

```
primary# ldm stop ldg1
```

10 Agregue el bus disponible al dominio que necesita acceso directo.

El bus disponible es pci_2 y el dominio es ldg1.

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

11 Reinicie el dominio para que se efectúe el cambio.

Los siguientes comandos reinician el dominio ldg1:

```
primary# ldm start ldg1
```

12 Confirme que el bus correcto aun está asignado al dominio primary y el bus correcto está asignado a dominio ldg1.

```
primary# ldm list-io
IO                PSEUDONYM          DOMAIN
--                -
pci@400           pci_0              primary
pci@500           pci_1              primary
pci@600           pci_2              ldg1
pci@700           pci_3

PCIE              PSEUDONYM          STATUS  DOMAIN
----              -
pci@400/pci@0/pci@d MB/PCIE0          EMP     -
pci@400/pci@0/pci@c MB/PCIE1          OCC     primary
pci@400/pci@0/pci@1 MB/HBA            OCC     primary
pci@500/pci@0/pci@d MB/PCIE4          EMP     -
pci@500/pci@0/pci@9 MB/PCIE5          EMP     -
pci@500/pci@0/pci@c MB/NET0           OCC     primary
pci@600/pci@0/pci@c MB/PCIE2          UNK     -
pci@600/pci@0/pci@9 MB/PCIE3          UNK     -
pci@700/pci@0/pci@c MB/PCIE6          UNK     -
pci@700/pci@0/pci@9 MB/PCIE7          UNK     -
```

Esta salida confirma que los buses PCIe pci_0 y pci_1 y los dispositivos debajo de éstos se asignan al dominio primary y que pci_2 y sus dispositivos se asignan a ldg1.

Asignación de dispositivos de punto final PCIe

Desde la versión 2.0 del Oracle VM Server for SPARC y el SO 10 9/0 Solaris de Oracle, puede asignar un dispositivo de punto final individual (o E/S directa asignable) a un dominio. El uso de estos dispositivos de punto final PCIe aumenta la granularidad de la asignación de dispositivos a los dominios E/S. Esta capacidad se ofrece con la característica de E/S directa (DIO).

La característica DIO le permite crear más dominios E/S que el número de buses PCIe en un sistema. El número posible de dominios E/S ahora está limitado sólo por el número de dispositivos de punto final PCIe.

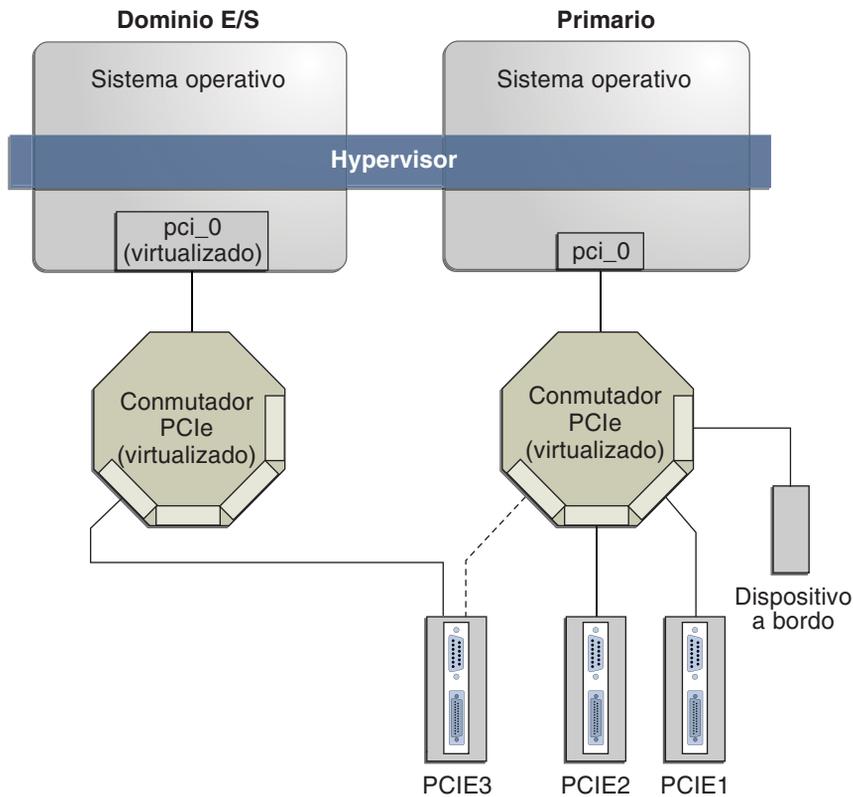
Un dispositivo de punto final PCIe puede ser uno de los siguientes:

- Una tarjeta PCIe en una ranura.
- Un dispositivo PCIe incorporado que es identificado por la plataforma

El siguiente diagrama muestra que el dispositivo de punto final PCIe, PCIE3, se asigna a un dominio E/S. Ambos bus `pci_0` y el conmutador en el dominio E/S son virtuales. No se puede tener acceso al dispositivo de punto final PCIE3 en el dominio `primary`.

En el dominio E/S, el bloqueo `pci_0` y el conmutador son un complejo de raíz virtual y un conmutador PCIe virtual respectivamente. Este bloque y conmutador son muy parecidos al bloque `pci_0` y el conmutador en el dominio `primary`. En el dominio `primary`, los dispositivos en la ranura PCIE3 son una forma enmascarada de los dispositivos originales y se identifican como `SUNW, assigned`.

FIGURA 6-2 Asignación de un dispositivo de punto final PCIe a un dominio E/S.



Use el comando `ldm list -io` para enumerar los dispositivos de punto final PCIe.

A pesar de que la característica DIO permite que cualquier tarjeta PCIe en una ranura sea asignada a un dominio E/S, sólo se admiten determinadas tarjetas PCIe. Véase “Requisitos de hardware y software para E/S directas” en *Notas sobre la versión del servidor Oracle VM para SPARC 2.0*.

Nota – No se admiten las tarjetas PCIe que tengan un conmutador o puente. Tampoco se admiten la asignación de nivel-función PCIe. La asignación de una tarjeta PCIe no admitida a un dominio E/S puede provocar un comportamiento impredecible.

A continuación se incluyen algunos detalles sobre la característica DIO:

- Esta característica está habilitada sólo cuando se cumplen todos los requisitos de software. Véase “Requisitos de hardware y software para E/S directas” en *Notas sobre la versión del servidor Oracle VM para SPARC 2.0*.
- Sólo los dispositivos de punto final PCIe que están conectados a un bus PCIe asignado al dominio `primary` pueden ser asignados a otro dominio con la característica DIO.
- Los dominios E/S que usan DIO tienen acceso a los dispositivos de punto final PCIe sólo cuando el dominio `primary` está en ejecución.
- El reinicio del dominio `primary` afecta a los dominios E/S que tienen dispositivos de punto final PCIe. Véase “Reinicio del dominio `primary`” en la página 73. El dominio `primary` también tiene las siguientes responsabilidades:
 - Inicializa el bus PCIe y gestiona el bus.
 - Gestiona todos los errores accionados por los dispositivos de punto final PCIe asignados a los dominios E/S. Tenga en cuenta que sólo el dominio `primary` recibe todos los errores relacionados con el bus PCIe.

Requisitos de hardware y software para E/S directas

Para usar correctamente la característica DIO, debe ejecutar el software adecuado y asignar sólo las tarjetas PCIe que son admitidas por la característica DIO a los dominios E/S. Para los requisitos de hardware y software, véase “Requisitos de hardware y software para E/S directas” en *Notas sobre la versión del servidor Oracle VM para SPARC 2.0*.

Nota – Todas las tarjetas PCIe que se admiten en una plataforma están admitidas en el dominio `primary`. Véase la documentación para la plataforma para la lista de tarjetas PCIe admitidas. Sin embargo, *sólo* las tarjetas PCIe admitidas de E/S directa pueden ser asignadas a dominios E/S.

Limitaciones de E/S directas

Para información sobre cómo solucionar las siguientes limitaciones, véase “Planificación de la configuración del dispositivo de punto final PCIe” en la página 72.

- Una reconfiguración retrasada se inicia cuando asigna o elimina un dispositivo de punto final PCIe a o del dominio `primary`, lo que significa que los cambios se aplican sólo después del reinicio del dominio `primary`.

El reinicio del dominio `primary` afecta la E/S directa, planee cuidadosamente los cambios en la configuración de E/S para maximizar los cambios relacionados con E/S directas en el dominio `primary` y minimizar los reinicios del dominio `primary`.

- La asignación o eliminación del dispositivo de punto final PCIe a cualquier dominio sólo se permite cuando ese dominio está parado o inactivo.

Planificación de la configuración del dispositivo de punto final PCIe

Planee cuidadosamente con tiempo la asignación o eliminación de dispositivos de punto final PCIe para evitar paradas del dominio `primary`. El reinicio de un dominio `primary` no sólo afecta a los servicios que están disponibles en el dominio `primary` mismo, sino que también afecta a los dominios E/S que tienen dispositivos de punto final PCIe asignados. A pesar de que los cambios en cada dominio E/S no afectan a los otros dominios, planificar la operación con tiempo le ayuda a minimizar las consecuencias en los servicios ofrecidos por ese dominio.

La reconfiguración retrasada se inicia la primera vez que asigna o elimina un dispositivo. Como resultado, puede continuar agregando o eliminando más dispositivos y después reiniciar el dominio `primary` sólo una vez para que se efectúen todos los cambios.

Por ejemplo, véase [“Creación un dominio E/S asignando el dispositivo de punto final PCIe” en la página 75.](#)

A continuación se describen los pasos generales que debe seguir para planificar y realizar la configuración del dispositivo DIO:

1. Entienda y grabe la configuración hardware del sistema.

Específicamente, grabe la información sobre los números de las piezas y otros detalles de las tarjetas PCIe en el sistema.

Use los comandos `ldm list-io -l` y `prtdiag -v` para obtener y guardar la información para consultarla más adelante.

2. Determine qué dispositivos de punto final PCIe son necesarios en el dominio `primary`.

Por ejemplo, determine los dispositivos de punto final PCIe que ofrecen acceso a los siguientes:

- Dispositivo de disco de inicio
- Dispositivo de red
- Otros dispositivos que el dominio `primary` ofrece como servicios

3. Elimine todos los dispositivos de punto final PCIe que pueda usar en los dominios E/S.

Este paso le ayuda a evitar realizar operaciones posteriores de reinicio en el dominio `primary`, ya que los reinicios afectan a los dominios E/S.

Use el comando `ldm rm-io` para eliminar los dispositivos de punto final PCIe. Use seudónimos en vez de rutas de dispositivos para especificar los dispositivos a los subcomandos `rm-io` y `add-io`.

Nota – A pesar de que la primera eliminación de un dispositivo de punto final PCIe puede iniciar una reconfiguración retrasada, puede continuar eliminando dispositivos. Después de haber eliminado todos los dispositivos que desee, sólo necesita reiniciar el dominio `primary` una vez para que se efectúen todos los cambios.

4. Guarde esta configuración en el procesador de servicios (SP).
Use el comando `ldm add-config`.
5. Reinicie el dominio `primary` para liberar los dispositivos de punto final PCIe que ha eliminado en el paso 3.
6. Confirme que los dispositivos de punto final PCIe que ha eliminado ya no están asignados al dominio `primary`.
Use el comando `ldm list-io -l` para comprobar que los dispositivos que ha eliminado aparecen como `SUNW,assigned-device` en la salida.
7. Asigne un dispositivo de punto final PCIe disponible a un dominio huésped para ofrecer acceso directo al dispositivo físico.
Después de haber realizado esta asignación, ya no puede migrar el dominio huésped a otro sistema físico con la característica de migración de dominio.
8. Agregue a o elimine del dominio huésped el dispositivo de punto final PCIe.
Use el comando `ldm add-io`.
Minimice los cambios en los dominios E/S reduciendo las operaciones de reinicio y evitando paradas de los servicios ofrecidos por ese dominio.
9. (Opcional) Realice cambios al hardware PCIe.
Véase [“Realización de cambios hardware en PCIe” en la página 74](#).

Reinicio del dominio `primary`

El dominio `primary` es el propietario del bus PCIe y es responsable de iniciar y administrar el bus. El dominio `primary` debe estar activo y ejecutar una versión del SO de Solaris de Oracle que admita la característica DIO. El apagado, paro o reinicio del dominio `primary` interrumpe el acceso al bus PCIe. Cuando el bus PCIe no está disponible, los dispositivos PCIe en ese bus se ven afectados y pueden no estar disponibles.

El comportamiento de los dominios E/S con dispositivos de punto final PCIe es impredecible cuando el dominio `primary` se reinicia mientras los dominios E/S están en ejecución. Por ejemplo, los dominios E/S con dispositivos de punto final PCIe pueden generar un error crítico durante o después del reinicio. En caso de reinicio del dominio `primary`, necesitará parar e iniciar manualmente cada dominio.

Para proporcionar una solución alternativa a estos temas, siga uno de los siguientes pasos:

- Apague manualmente cualquier dominio en el sistema que tenga dispositivos de punto final PCIe asignados a él *antes* de apagar el dominio `primary`.

Este paso le asegura que esos dominios se han apagado bien antes de que usted apague, pare o reinicie el dominio `primary`.

Para encontrar todos los dominios que tienen dispositivos de punto final PCIe asignados a ellos, ejecute el comando `ldm list -io`. Este comando le permite enumerar los dispositivos de punto final PCIe que han sido asignados a los dominios en el sistema. Así pues, utilice esta información para ayudarle a planificar. Para una descripción detallada de este comando, véase la página de comando `man ldm(1M)`.

Para cada dominio que se encuentra, pare el dominio ejecutando el comando `ldm stop`.

- Configure la relación de dependencia de un dominio entre el dominio `primary` y los dominios a los que se han asignado dispositivos de punto final PCIe.

Esta relación de dependencia asegura que los dominios con dispositivos de punto final PCIe reinician automáticamente cuando el dominio `primary` reinicia por cualquier razón.

Tenga en cuenta que esta relación de dependencia reinicia por la fuerza todos los dominios, y no pueden apagarse correctamente. En cualquier caso, la relación de dependencia no afecta a los dominios que se han cerrado manualmente.

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

Realización de cambios hardware en PCIe

Los siguientes pasos le ayudan a evitar errores de configuración en las asignaciones de punto final de PCIe. Para información sobre plataformas específicas sobre la instalación y eliminación de hardware específico, véase la documentación para la plataforma.

- No es necesaria ninguna acción si instala una tarjeta PCIe en una ranura vacía. La tarjeta PCIe es propiedad automáticamente del dominio que posee el bus PCIe.
Para asignar la nueva tarjeta PCIe a un dominio E/S, use el comando `ldm rm -io` para quitar primero la tarjeta del dominio `primary`. Entonces, use el comando `ldm add -io` para asignar la tarjeta a un dominio E/S.
- No es necesaria ninguna acción si una tarjeta PCIe se quita del sistema y se asigna al dominio `primary`.
- Para eliminar una tarjeta PCIe que está asignada a un dominio E/S, primero elimine el dispositivo del dominio E/S. Después, agregue el dispositivo al dominio `primary` antes de quitar el dispositivo del sistema físicamente.
- Para sustituir una tarjeta PCIe que está asignada a un dominio E/S, compruebe que la nueva tarjeta es admitida por la característica DIO.

Si es así, no es necesaria ninguna acción para asignar automáticamente la nueva tarjeta al dominio E/S actual.

Si no es así, primero elimine esa tarjeta PCIe del dominio E/S usando el comando `ldm rm -io`. Después, use el comando `ldm add -io` para reasignar esta tarjeta PCIe al dominio `primary`. Entonces, sustituya físicamente la tarjeta PCIe que ha asignado al dominio `primary` con una tarjeta PCIe diferente. Estos pasos le permiten evitar una configuración no admitida por la característica DIO.

▼ Creación un dominio E/S asignando el dispositivo de punto final PCIe

Planifique todas las implementaciones DIO con tiempo para minimizar el tiempo de paro.

Para un ejemplo sobre cómo agregar un dispositivo de punto final PCIe para crear un dominio E/S, véase [“Planificación de la configuración del dispositivo de punto final PCIe” en la página 72.](#)

1 Identifique y archive los dispositivos que están actualmente instalados en el sistema.

La salida del comando `ldm list -io -l` muestra cómo están configurados actualmente los dispositivos de E/S. Puede obtener más información detallada usando el comando `prt diag -v`.

Nota – Después de haber asignado los dispositivos a los dominios E/S, la identidad de los dispositivos sólo puede ser determinada en los dominios E/S.

```
# ldm list-io -l
IO                PSEUDONYM          DOMAIN
--                -
pci@400           pci_0               primary
pci@500           pci_1               primary

PCIe              PSEUDONYM          STATUS  DOMAIN
-----
pci@400/pci@0/pci@c PCIe1               EMP     -
pci@400/pci@0/pci@9 PCIe2               OCC     primary
network@0
network@0,1
network@0,2
network@0,3
pci@400/pci@0/pci@d PCIe3               OCC     primary
SUNW,emlxs/fp/disk
SUNW,emlxs@0,1/fp/disk
SUNW,emlxs@0,1/fp@0,0
pci@400/pci@0/pci@8 MB/SASHBA          OCC     primary
scsi@0/tape
scsi@0/disk
scsi@0/sd@0,0
scsi@0/sd@1,0
```

```

pci@500/pci@0/pci@9 PCIE0 EMP -
pci@500/pci@0/pci@d PCIE4 OCC primary
network@0
network@0,1
pci@500/pci@0/pci@c PCIE5 OCC primary
SUNW,qlc@0/fp/disk
SUNW,qlc@0/fp@0,0
SUNW,qlc@0,1/fp/disk
SUNW,qlc@0,1/fp@0,0/ssd@w21000011c605dbab,0
SUNW,qlc@0,1/fp@0,0/ssd@w21000011c6041434,0
SUNW,qlc@0,1/fp@0,0/ssd@w21000011c6053652,0
SUNW,qlc@0,1/fp@0,0/ssd@w21000011c6041b4f,0
SUNW,qlc@0,1/fp@0,0/ssd@w21000011c605dbb3,0
SUNW,qlc@0,1/fp@0,0/ssd@w21000011c60413bc,0
SUNW,qlc@0,1/fp@0,0/ssd@w21000011c604167f,0
SUNW,qlc@0,1/fp@0,0/ssd@w21000011c6041b3a,0
SUNW,qlc@0,1/fp@0,0/ssd@w21000011c605dabf,0
SUNW,qlc@0,1/fp@0,0/ssd@w21000011c60417a4,0
SUNW,qlc@0,1/fp@0,0/ssd@w21000011c60416a7,0
SUNW,qlc@0,1/fp@0,0/ssd@w21000011c60417e7,0
SUNW,qlc@0,1/fp@0,0/ses@w215000c0ff082669,0
pci@500/pci@0/pci@8 MB/NET0 OCC primary
network@0
network@0,1
network@0,2
network@0,3

```

2 Determine la ruta del dispositivo del disco de inicio, que es necesario guardar.

- Para los archivos de sistema UFS, ejecute el comando `df /` para determinar la ruta del dispositivo del disco de inicio.

```

primary# df /
/ (dev/dsk/c0t1d0s0 ): 1309384 blocks 457028 files

```

- Para los sistemas de archivos ZFS, primero ejecute el comando `df /` para determinar el nombre del grupo, y después ejecute el comando `zpool status` para determinar la ruta del dispositivo del disco de inicio.

```

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 Determine el dispositivo físico con el que está enlazado el dispositivo en bloque.

El siguiente ejemplo usa el dispositivo en bloque `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
```

En este ejemplo, el dispositivo físico para el disco de inicio del dominio `primary` está conectado al dispositivo de punto final PCIe (`pci@400/pci@0/pci@8`), que corresponde a la enumeración de MB/SASHBA en el paso 1. Si se elimina este dispositivo se evitará que el dominio `primary` inicie, por lo tanto *no* elimine este dispositivo del dominio `primary`.

4 Determine la interfaz de red usada por el sistema.

```
# 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
```

En este ejemplo, se usa la interfaz `nxge0` como interfaz de red para el dominio `primary`.

5 Determine el dispositivo físico con el que está enlazado la interfaz de red.

El siguiente comando usa la interfaz de red `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
```

En este ejemplo, el dispositivo físico para la interfaz de red usado por el dominio `primary` está conectado al dispositivo de punto final PCIe (`pci@500/pci@0/pci@8`), que corresponde a la enumeración de MB/NET0 en el paso 1. Así que *no* desea eliminar este dispositivo del dominio `primary`. Puede asignar con seguridad todos los otros dispositivos PCIe a otros dominios que no son usados por el dominio `primary`.

Si la interfaz de red usada por el dominio `primary` es un bus que desea asignar a otro dominio, el dominio `primary` debe ser reconfigurado para usar una interfaz de red diferente.

6 Elimine los dispositivos de punto final PCIe que pueda usar en los dominios E/S.

En este ejemplo, puede eliminar los dispositivos de punto final PCIE2, PCIE3, PCIE4 y PCIE5 porque no son usados por el dominio `primary`.

a. Elimine los dispositivos de punto final PCIe.



Precaución – No elimine los dispositivos que se usan en el dominio `primary`.

Si ha eliminado por error un dispositivo equivocado, use el comando `ldm cancel -op reconf primary` para cancelar la reconfiguración retrasada en el dominio `primary`.

Puede eliminar varios dispositivos al mismo tiempo para evitar múltiples reinicios.

```
# 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. Guarde la nueva configuración en el procesador de servicio (SP).

El siguiente comando guarda la configuración en un archivo llamado `dio`:

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

c. Reinicie el sistema para reflejar la eliminación de los dispositivos de punto final PCIe.

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

7 Inicie la sesión en el dominio `primary` y compruebe que los dispositivos de punto final PCIe ya no están asignados al dominio.

```

# ldm list-io
IO                PSEUDONYM          DOMAIN
--                -
pci@400           pci_0              primary
pci@500           pci_1              primary

PCIe              PSEUDONYM          STATUS  DOMAIN
-----
pci@400/pci@/pci@c PCIe1              EMP     -
pci@400/pci@/pci@9 PCIe2              OCC
pci@400/pci@/pci@d PCIe3              OCC
pci@400/pci@/pci@8 MB/SASHBA          OCC     primary
pci@500/pci@/pci@9 PCIe0              EMP     -
pci@500/pci@/pci@d PCIe4              OCC
pci@500/pci@/pci@c PCIe5              OCC
pci@500/pci@/pci@8 MB/NET0            OCC     primary

```

Nota – La salida `ldm list-io -l` puede mostrar `SUNW, assigned-device` para los dispositivos de punto final PCIe que han sido eliminados. La información actual ya no está disponible desde el dominio `primary`, pero el dominio al que se ha asignado el dispositivo tiene esta información.

8 Asignación de un dispositivo de punto final PCIe a un dominio.

a. Agregue el dispositivo PCIe2 al dominio `ldg1`.

```
# ldm add-io PCIe2 ldg1
```

b. Enlace e inicie el dominio ldg1.

```
# ldm bind ldg1
# ldm start ldg1
LDom ldg1 started
```

9 Inicie la sesión en el dominio ldg1 y compruebe que el dispositivo está disponible para el uso.

Use el comando `dladm show-dev` para comprobar que el dispositivo de red está disponible. Después, configure el dispositivo de red para el uso en el dominio.

```
# 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
```


Uso de discos virtuales

Este capítulo describe cómo usar los discos virtuales con el software del Oracle VM Server for SPARC.

Este capítulo trata sobre los siguientes temas:

- “Introducción a los discos virtuales” en la página 81
- “Administración de discos virtuales” en la página 82
- “Identificador de disco virtual y nombre del dispositivo” en la página 85
- “Apariencia del disco virtual” en la página 85
- “Opciones del componente posterior del disco virtual” en la página 86
- “Componente posterior de un disco virtual” en la página 88
- “Configuración de ruta múltiple de disco virtual” en la página 94
- “CD, DVD e imágenes ISO” en la página 97
- “Tiempo de espera de disco virtual” en la página 100
- “Disco virtual y SCSI” en la página 101
- “Disco virtual y el comando `format(1M)`” en la página 102
- “Uso de ZFS con discos virtuales” en la página 102
- “Uso de los administradores de volumen en el entorno Dominios lógicos” en la página 106

Introducción a los discos virtuales

Un disco virtual contiene dos componentes: el mismo disco virtual que aparece en un dominio huésped, y el componente posterior del disco virtual, que es donde se almacenan los datos y donde acaban las E/S virtuales. El componente posterior del disco virtual es exportado desde un dominio de servicio por el controlador del servidor de disco virtual (vds). El controlador vds se comunica con el controlador del cliente del disco virtual (vdc) en el dominio huésped a través del hypervisor usando un canal del dominio lógico (LDC). Finalmente, aparece un disco virtual como dispositivos `/dev/[r]dsk/cXdYsZ` en el dominio huésped.

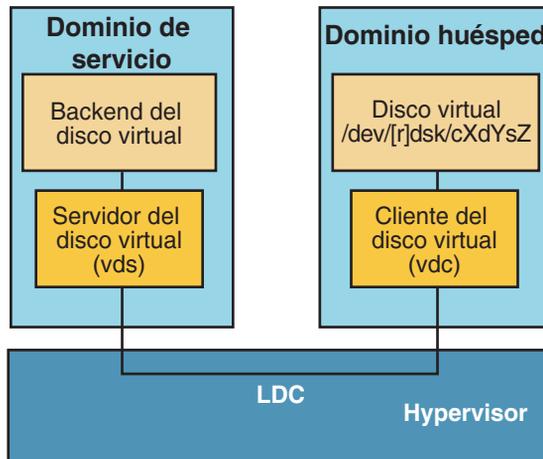
El componente posterior de un disco virtual puede ser físico o lógico. Los dispositivos físicos pueden incluir:

- Disco físico o número de unidad lógica del disco (LUN)
- Segmento de disco físico

Los dispositivos lógicos pueden ser uno de los siguientes:

- Archivo en un sistema de archivos, como ZFS o UFS
- Volumen lógico de un administrador de volúmenes, como ZFS, VxVM, o Solaris Volume Manager
- Cualquier pseudo dispositivo de disco que se puede acceder desde el dominio de servicio

FIGURA 7-1 Discos virtuales con Dominios lógicos



Administración de discos virtuales

Esta sección describe cómo agregar un disco virtual a un dominio huésped, cambiar las opciones de disco virtual y tiempo de espera y eliminar un disco virtual de un dominio de servicio. Véase [“Opciones del componente posterior del disco virtual” en la página 86](#) para una descripción de las opciones del disco virtual. Véase [“Tiempo de espera de disco virtual” en la página 100](#) para una descripción del tiempo de espera del disco virtual.

▼ Agregación de disco virtual

- 1 Exporte el componente posterior de un disco virtual desde el dominio de servicio.

```
# ldm add-vdsdev [options={ro,slice,excl}] [mpgroup=mpgroup] \  
backend volume-name@service-name
```

- 2 Asigne el componente posterior a un dominio huésped.

```
# ldm add-vdisk [timeout=seconds] [id=disk-id] disk-name volume-name@service-name ldom
```

Puede especificar un id de un nuevo dispositivo de disco virtual configurando la propiedad `id`. De manera predeterminada, estos valores de ID se generan automáticamente, así que debe configurar esta propiedad si necesita que coincida con un nombre de un dispositivo existente en el SO. Véase “Identificador de disco virtual y nombre del dispositivo” en la página 85.

Nota – En realidad un componente posterior se exporta del dominio de servicio y es asignado al dominio huésped cuando el dominio huésped (*ldom*) está enlazado.

▼ Exportación del componente posterior de un disco virtual varias veces

Un componente posterior de un disco virtual puede ser exportado varias veces a través del mismo o de diferentes servidores de disco virtual. Cada instancia exportada del componente posterior del disco virtual puede entonces ser asignada con el mismo o con diferentes dominios huésped.

Cuando un componente posterior del disco virtual se exporta varias veces, no debe ser exportado con la opción exclusiva (`excl`). Si se especifica la opción `excl` se permitirá la exportación del componente posterior sólo una vez. El componente posterior puede ser exportado de manera segura varias veces como dispositivo de sólo lectura con la opción `ro`.



Precaución – Cuando el componente posterior de un disco virtual se exporta varias veces, las aplicaciones en ejecución en los dominios huésped y que usan ese disco virtual son responsables de la coordinación y sincronización a los accesos de escritura concurrentes para asegurar la coherencia de los datos.

El siguiente ejemplo describe cómo agregar el mismo disco virtual a dos dominios huésped diferentes a través del mismo servicio de disco virtual.

- 1 Exporte el componente posterior del disco virtual dos veces desde un dominio de servicio usando los siguientes comandos.

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

Tenga en cuenta que el segundo comando `ldm add-vdsdev` usa la opción `-f` para la segunda exportación del componente posterior. Use esta opción cuando usa la misma ruta de componente posterior para ambos comandos y cuando los servidores del disco virtual están ubicados en el mismo dominio de servicio.

- 2 **Asigne el componente posterior exportado a cada dominio huésped usando los siguientes comandos.**

El *disk-name* puede ser diferente para `ldom1` y `ldom2`.

```
# ldm add-vdisk [timeout=seconds] disk-name volume1@service-name ldom1
# ldm add-vdisk [timeout=seconds] disk-name volume2@service-name ldom2
```

▼ Cambio de las opciones del disco virtual

Para más información sobre las opciones del disco virtual véase [“Opciones del componente posterior del disco virtual” en la página 86](#).

- Después de haber exportado el componente posterior desde el dominio de servicio, puede cambiar las opciones del disco virtual usando el siguiente comando.

```
# ldm set-vdsdev options=[{ro,slice,excl}] volume-name@service-name
```

▼ Cambio de la opción de tiempo de espera

Para más información sobre las opciones del disco virtual véase [“Opciones del componente posterior del disco virtual” en la página 86](#).

- Después de la asignación de un disco virtual a un dominio huésped, puede cambiar el tiempo de espera del disco virtual usando el siguiente comando.

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

▼ Eliminación de disco virtual

- 1 **Elimine un disco virtual del dominio huésped usando el siguiente comando.**

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

- 2 **Pare la exportación del componente posterior correspondiente desde el dominio de servicio usando el siguiente comando.**

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

Identificador de disco virtual y nombre del dispositivo

Cuando use el comando `ldm add-vdisk` para agregar un disco virtual a un dominio, puede especificar el número de dispositivo configurando la propiedad `id`.

```
# ldm add-vdisk [id=disk-id] disk-name volume-name@service-name ldom
```

Cada disco virtual de un dominio tiene un número de dispositivo único que se asigna cuando el dominio está enlazado. Si un disco virtual se ha agregado con un número de dispositivo explícito (configurando la propiedad `id`), se usa el número de dispositivo especificado. En caso contrario, el sistema asignará automáticamente el número de dispositivo más bajo posible. En este caso, el número de dispositivo asignado depende de cómo se agregan los discos virtuales al dominio. El número de dispositivo eventualmente asignado a un disco virtual se puede ver en la salida del comando `ldm list-bindings` cuando un dominio está enlazado.

Cuando un dominio con discos virtuales ejecuta el SO de Solaris de Oracle, cada disco virtual aparece como un dispositivo de disco `c0dn`, donde `n` es el número del dispositivo del disco virtual.

En el siguiente ejemplo, el dominio `ldg1` tiene dos discos virtuales: `rootdisk` y `pdisk`. `rootdisk` tiene un número de dispositivo de `0` (`disk@0`) y aparece en el dominio como el dispositivo de disco `c0d0`. `pdisk` tiene un número de dispositivo de `1` (`disk@1`) y aparece en el dominio como el dispositivo del disco `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
...
```



Precaución – Si no se asigna explícitamente un número de dispositivo a un disco virtual, el número de dispositivo puede cambiar cuando el dominio se desenlaza y se enlaza de nuevo. En este caso, el nombre del dispositivo asignado por el SO en ejecución en el dominio también puede cambiar e interrumpir la configuración existente en el sistema. Esto puede suceder, por ejemplo, cuando se elimina un disco virtual de la configuración del dominio.

Apariencia del disco virtual

Cuando un componente posterior se exporta como disco virtual, puede aparecer en el dominio huésped como disco completo o como disco de segmento único. La manera en que aparece depende del tipo de componente posterior y de las opciones usadas para exportarlo.

Disco lleno

Cuando un componente posterior se exporta a un dominio como disco completo, aparece en dicho dominio como disco normal con 8 segmentos (`s0` a `s7`). Este disco puede verse con el comando `format(1M)`. La tabla de particiones del disco del disco puede cambiarse usando el comando `fmthard(1M)` o bien `format(1M)`.

El disco completo también es visible desde el software de instalación del SO y puede ser seleccionado como disco en el que instalar el SO.

Cualquier componente posterior puede exportarse como disco completo excepto segmentos de disco físico que sólo pueden exportarse como disco de segmento único.

Disco de segmento único

Cuando un componente posterior se exporta a un dominio como disco de segmento único, aparece en dicho dominio como disco normal con 8 segmentos (`s0` a `s7`). En cualquier caso, sólo se puede usar el primer segmento (`s0`). Este tipo de disco es visible con el comando `format(1M)`, pero la tabla de partición del disco no puede cambiarse.

Un disco de segmento único también es visible para el software de instalación del SO y puede ser seleccionado como disco en el que instalar el SO. En este caso, si instala el SO usando el sistema de archivos UNIX (UFS), sólo debe definirse la partición de raíz (`/`) y esta partición debe usar todo el espacio del disco.

Cualquier componente posterior puede exportarse como disco de segmento único excepto los discos físicos que sólo pueden exportarse como discos completos.

Nota – Antes de la versión SO 10 10/08 de Solaris de Oracle, un disco de segmento único aparecía como un disco con una sola partición (`s0`). Este tipo de disco no es visible con el comando `format(1M)`. Este disco no era visible para el software de instalación del SO y no podía ser seleccionado como dispositivo de disco en el que podía instalarse el SO.

Opciones del componente posterior del disco virtual

Pueden especificarse diferentes opciones cuando se exporta el componente posterior de un disco virtual. Estas opciones se indican en el argumento `options=` del comando `ldm add-vdsdev` como una lista de valores separados por comas. Las opciones válidas son: `ro`, `slice` y `excl`.

Opción de sólo lectura (ro)

La opción de sólo lectura (ro) especifica que el componente posterior debe exportarse como dispositivo de sólo lectura. En este caso, se puede acceder al disco virtual asignado al dominio huésped sólo para operaciones de lectura, y fallará cualquier operación de escritura en el disco virtual.

Opción exclusiva (excl)

La opción exclusiva (excl) especifica que el componente posterior en el dominio de servicio tiene que abrirse en exclusiva por el servidor del disco virtual cuando es exportado como disco virtual a otro dominio. Cuando un componente posterior se abre de forma exclusiva, las otras aplicaciones en el dominio de servicio no pueden acceder a él. Esto evita que se ejecuten aplicaciones en el dominio de servicio de forma inadvertida usando un componente posterior que también está siendo usado por un dominio de servicio.

Nota – Algunos controladores no respetan la opción `excl` y no permitirán que los componentes posteriores del disco virtual se abran de forma exclusiva. Se sabe que la opción `excl` funciona con discos físicos y segmentos, pero la opción no funciona con archivos. Puede funcionar o no con pseudo dispositivos, como los volúmenes de disco. Si el controlador del componente posterior no cumple la apertura exclusiva, se ignora la opción `excl` del componente posterior, y el componente posterior no se puede abrir de manera exclusiva.

Dado que la opción `excl` evita que las aplicaciones en ejecución en el dominio de servicio accedan al componente posterior exportado a un dominio de servicio, no fije la opción `excl` en las siguientes situaciones:

- Cuando los dominios huésped están en ejecución, puede desear usar comandos como `format(1M)` o `luxadm(1M)` para administrar los disco físicos, entonces no exporte estos discos con la opción `excl`.
- Cuando exporta un volumen Solaris Volume Manager, como un RAID o un volumen reflejado, no fije la opción `excl`. En caso contrario, esto evitará que el Solaris Volume Manager comience algunas operaciones de recuperación en caso que un componente del RAID o del volumen reflejado falle. Véase “[Uso de discos virtuales para Solaris Volume Manager](#)” en la página 107 para más información.
- Si está instalado el Veritas Volume Manager (VxVM) en el dominio de servicio y el Veritas Dynamic Multipathing (VxDMP) está habilitado para discos físicos, entonces los discos físicos tienen que ser exportados sin la opción no predeterminada `excl`. En caso contrario, falla la exportación, ya que el servidor de disco virtual (vds) no puede abrir el dispositivo de disco físico. Véase “[Uso de discos virtual cuando VxVM está instalado](#)” en la página 108 para más información.

- Si está exportando el mismo componente posterior de disco virtual varias veces desde el mismo servicio de disco virtual, véase [“Exportación del componente posterior de un disco virtual varias veces” en la página 83](#) para más información.

De manera predeterminada, el componente posterior se abre de manera no exclusiva. De esa manera el componente posterior puede ser usado por aplicaciones en ejecución en el dominio de servicio mientras se exporta a otro dominio. Tenga en cuenta que éste es un nuevo comportamiento desde la versión SO 10 5/08 de de Solaris de Oracle. Antes de la versión SO 10 5/08 de Solaris de Oracle, los componentes posteriores del disco se abrían siempre exclusivamente, y no era posible que un componente posterior se abriese de manera no exclusiva.

Opción segmento (slice)

Un componente posterior normalmente se exporta como disco completo o bien como disco de segmento único dependiendo del tipo. Si se especifica la opción `slice`, el componente posterior se exporta por la fuerza como disco de segmento único.

Esta opción es útil si desea exportar el contenido sin formato de un componente posterior. Por ejemplo, si tiene un volumen ZFS o Solaris Volume Manager donde ya ha guardado datos y desea que el dominio huésped acceda a estos datos, entonces debe exportar el volumen ZFS o Solaris Volume Manager usando la opción `slice`.

Para más información sobre esta opción, véase [“Componente posterior de un disco virtual” en la página 88](#).

Componente posterior de un disco virtual

El componente posterior de un disco virtual es la ubicación donde se guardan los datos del disco virtual. El componente posterior puede ser un disco, un segmento de disco, un archivo, un volumen, como ZFS, Solaris Volume Manager o VxVM. Un componente posterior aparece en un dominio huésped como disco completo o disco de segmento único, dependiendo de si la opción `slice` está configurada cuando se exporta el componente posterior desde el dominio de servicio. De manera predeterminada, el componente posterior de un disco virtual se exporta de manera no exclusiva como disco completo en el que se puede leer y escribir.

Disco físico o LUN de disco

Un disco físico o un LUN de disco siempre se exporta como disco lleno. En este caso, los controladores de disco virtual (`vds` y `vdsc`) envían E/S desde el disco virtual y actúan como paso a través a los discos físicos o al LUN del disco.

Un disco físico o un LUN de disco se exporta desde un dominio de servicio exportando el dispositivo que corresponde al segmento 2 (s2) de ese disco sin configurar la opción `slice`. Si exporta el segmento 2 de un disco con la opción `slice`, sólo se exporta ese segmento y no todo el disco.

▼ Exportación de un disco físico como disco virtual

1 Exporte un disco físico como disco virtual.

Por ejemplo, para exportar el disco físico `c1t48d0` como un disco virtual, debe exportar el segmento 2 de dicho disco (`c1t48d0s2`).

```
primary# ldm add-vdsdev /dev/dsk/c1t48d0s2 c1t48d0@primary-vds0
```

2 Asignación del disco al dominio huésped

Por ejemplo, asigne el disco (`pdisk`) al dominio huésped `ldg1`.

```
primary# ldm add-vdisk pdisk c1t48d0@primary-vds0 ldg1
```

3 Después del inicio del dominio huésped y la ejecución del SO de Solaris de Oracle, compruebe que se puede acceder al disco y que es un disco completo.

Un disco completo es un disco normal que tiene ocho (8) segmentos.

Por ejemplo, el disco que se está comprobando es `c0d1`.

```
ldg1# ls -l /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
```

Segmento de disco físico

Un segmento de disco físico siempre se exporta como disco de segmento único. En este caso, los controladores de disco virtual (`vds` y `vdc`) envían E/S desde el disco virtual y actúan como paso a través a los discos físicos o al segmento del disco físico.

Un segmento del disco físico se exporta de un dominio de servicio exportando el dispositivo de segmento correspondiente. Si el dispositivo es diferente del segmento 2 se exporta automáticamente como disco de segmento único si especifica la opción `slice` o no. Si el dispositivo es el segmento 2 del disco, debe configurar la opción `slice` para exportar sólo el segmento 2 como disco de segmento único; en caso contrario, se exporta todo el disco como disco completo.

▼ Exportación de un segmento de disco físico como disco virtual

1 Exporte un segmento de disco físico como disco virtual.

Por ejemplo, para exportar el segmento 0 del disco físico `c1t57d0` como disco virtual, debe exportar el dispositivo que corresponde a ese segmento (`c1t57d0s0`) de la siguiente manera.

```
primary# ldm add-vdsdev /dev/dsk/c1t57d0s0 c1t57d0s0@primary-vds0
```

No es necesario especificar la opción `slice`, ya que un slice siempre se exporta como disco de una sola slice.

2 Asignación del disco al dominio huésped

Por ejemplo, asigne el disco (`pslice`) al dominio huésped `ldg1`.

```
primary# ldm add-vdisk pslice c1t57d0s0@primary-vds0 ldg1
```

3 Después de haber iniciado el dominio huésped y ejecutado el SO de Solaris de Oracle, puede enumerar el disco (`c0d13`, por ejemplo) y ver que el disco es accesible.

```
ldg1# ls -l /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
```

A pesar de que hay 8 dispositivos, dado que el disco es un disco de segmento único, sólo se puede usar el primer segmento (`s0`).

▼ Exportación del segmento 2

- Para exportar el segmento 2 (disco `c1t57d0s2`, por ejemplo) debe especificar la opción `slice`; en caso contrario, se exporta todo el disco.

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

Archivo y volumen

Un archivo o volumen (por ejemplo, de ZFS o Solaris Volume Manager) se exporta como disco completo o como disco de segmento único dependiendo de si está fijada o no la opción `slice`.

Archivo o volumen exportado como disco lleno

Si no configura la opción `slice`, un archivo o volumen se exporta como disco completo. En este caso, los controladores de disco virtual (`vds` y `vdc`) envían E/S desde el disco virtual y administran la partición del disco virtual. El archivo o volumen eventualmente se convierte en una imagen de disco que contiene datos de todos los segmentos del disco virtual y metadatos usados para administrar la partición y estructura del disco.

Cuando se exporta un archivo o volumen en blanco como disco completo, aparece en el dominio huésped como disco sin formato, esto es, un disco sin partición. Entonces es necesario ejecutar el comando `format(1M)` en el dominio huésped para definir las particiones que se pueden usar y escribir una etiqueta de disco válida. Cualquier E/S al disco virtual falla cuando el disco no tiene formato.

Nota – Antes de la versión SO 5/08 de Solaris de Oracle, cuando se exporta un archivo en blanco como disco virtual, el sistema escribía una etiqueta de disco predeterminada y creaba una partición predeterminada. Esta operación ya no se realiza con la versión SO 5/08 de Solaris de Oracle, debe ejecutar `format(1M)` en el dominio huésped para crear particiones.

▼ Exportación de un archivo como disco lleno

- 1 Desde el dominio de servicio, cree un archivo (`fdisk0` por ejemplo) que se usará como disco virtual.

```
service# mkfile 100m /ldoms/domain/test/fdisk0
```

El tamaño del archivo define el tamaño del disco virtual. Este ejemplo crea un archivo en blanco de 100 megabytes para obtener un disco virtual de 100 megabytes.

- 2 Para el dominio de control, exporte el archivo como disco virtual.

```
primary# ldm add-vdsdev /ldoms/domain/test/fdisk0 fdisk0@primary-vds0
```

En este ejemplo, la opción `slice` no se ha fijado, así que el archivo se exporta como disco completo.

- 3 Desde el dominio de control, asigne el disco al dominio huésped.

Por ejemplo, asigne el disco (`fdisk`) al dominio huésped `ldg1`.

```
primary# ldm add-vdisk fdisk fdisk0@primary-vds0 ldg1
```

- 4 Después del inicio del dominio huésped y la ejecución del SO de Solaris de Oracle, compruebe que se puede acceder al disco y que es un disco completo.

Un disco completo es un disco normal con 8 segmentos.

El siguiente ejemplo muestra cómo enumerar el disco, `c0d5` y comprueba que es accesible y que es un disco completo.

```
ldg1# ls -l /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
```

Archivo o volumen exportado como disco de segmento único

Si se fija la opción `slice`, entonces el volumen se exporta como disco de segmento único. En ese caso, el disco virtual sólo tiene una partición (`s0`), que se asigna directamente al componente posterior del archivo o del volumen. El archivo o volumen sólo contiene datos escritos en el disco virtual sin datos extra como información sobre la partición o la estructura del disco.

Cuando se exporta un archivo o volumen como disco de segmento único, el sistema simula una partición del disco falsa que hace que el archivo o volumen aparezca como un segmento del disco. Dado que la partición del disco es simulada, no puede crear una partición para ese disco.

▼ Exportación de un volumen ZFS como un disco de segmento único

1 Cree un volumen ZFS para usar como disco de segmento único.

El siguiente ejemplo muestra cómo crear un volumen ZFS, `zdisk0`, para usarlo como disco de segmento único.

```
service# zfs create -V 100m ldoms/domain/test/zdisk0
```

El tamaño del volumen define el tamaño del disco virtual. Este ejemplo crea un volumen de 100 megabytes para obtener un disco virtual de 100 megabyte.

2 Desde el dominio de control, exporte el dispositivo correspondiente a dicho volumen ZFS, y configure la opción `slice` de manera que el volumen se exporte como disco de segmento único.

```
primary# ldm add-vdsdev options=slice /dev/zvol/dsk/ldoms/domain/test/zdisk0 \
zdisk0@primary-vds0
```

3 Desde el dominio de control, asigne el volumen al dominio huésped.

A continuación se muestra cómo asignar el volumen `zdisk0` al dominio huésped `ldg1`.

```
primary# ldm add-vdisk zdisk0 zdisk0@primary-vds0 ldg1
```

4 Después de haber iniciado el dominio huésped y ejecutado el SO de Solaris de Oracle, puede enumerar el disco (`c0d9`, por ejemplo) y ver que el disco es accesible y es un disco de segmento único (`s0`).

```
ldg1# ls -l /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

```

Exportación de volúmenes y compatibilidad con versiones anteriores

Antes de la versión SO 10 5/08 de Solaris de Oracle, no existía la opción `slice` y los volúmenes se exportaban como discos de segmento único. Si tiene una configuración que exporta volúmenes como discos virtuales y actualiza el sistema al SO 10 5/08 de Solaris de Oracle, los volúmenes se exportan como discos completos en vez de discos de segmento único. Para mantener el antiguo comportamiento y que los volúmenes se exporten como discos de segmento único, debe realizar una de estas acciones:

- Use el comando `ldm set -vdsdev` en el software de Oracle VM Server for SPARC 2.0 y configure la opción `slice` para todos los volúmenes que desea exportar como discos de segmento único. Consulte la página de comandos [man `ldm\(1M\)`](#) para más información sobre este comando.
- Agregue la siguiente línea al archivo `/etc/system` en el dominio de servicio.

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

Nota – Si configura esta opción fuerza la exportación de todos los volúmenes como discos de segmento único, y no puede exportar ningún volumen como disco completo.

Resumen de cómo se exportan los diferentes tipos de componentes posteriores

Componente posterior	Sin opción de segmento	Opción de segmento fijada
Disco (segmento de disco 2)	Disco lleno ¹	Disco de segmento único ²
Segmento de disco (no segmento 2)	Disco de segmento único ³	Disco de segmento único
Archivo	Disco lleno	Disco de segmento único
Volumen, incluyendo ZFS, Solaris Volume Manager, o VxVM	Disco lleno	Disco de segmento único

¹ Exporte todo el disco.

² Exportar solo segmento 2

³ Un segmento siempre se exporta como disco de segmento único.

Recomendaciones para la exportación de un archivo y un segmento de disco como discos virtuales

Esta sección incluye recomendaciones para la exportación de un archivo o un segmento de disco como disco virtual.

Uso del controlador del archivo de bucle invertido (lofi)

Se puede usar el controlador del archivo de bucle invertido (`lofi`) para exportar un archivo como disco virtual. En cualquier caso, esto agrega una capa extra al controlador y afecta el rendimiento del disco virtual. En vez de ello, puede exportar directamente un archivo como disco completo o como disco de segmento único. Véase “[Archivo y volumen](#)” en la [página 90](#).

Exportación directa o indirecta de segmento de disco

Para exportar un segmento como disco virtual directa o indirectamente (por ejemplo, mediante un volumen Solaris Volume Manager), asegúrese de que el segmento no inicia en el primero bloque (bloque 0) del disco físico usando el comando `prtvtoc(1M)`.

Si exporta directa o indirectamente un segmento de disco que inicia en el primer bloque de un disco físico, puede sobrescribir la tabla de particiones del disco físico y dejar todas las particiones del disco inaccesibles.

Configuración de ruta múltiple de disco virtual

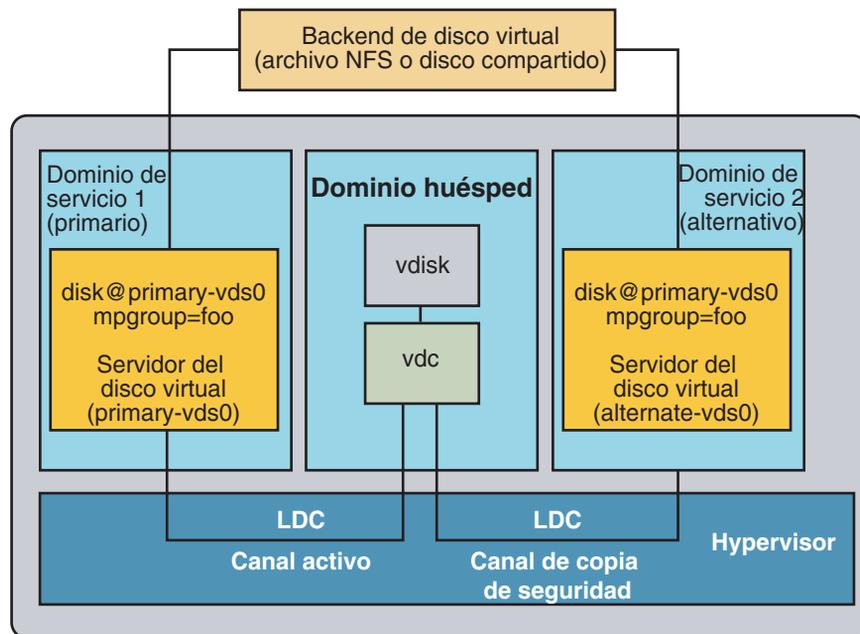
La ruta múltiple de disco virtual le permite configurar un disco virtual en un dominio huésped para acceder al almacenamiento del componente posterior por más de una ruta. La ruta lleva a diferentes dominios de servicio que ofrecen acceso al mismo almacenamiento de componente posterior, como un LUN de disco. Esta característica permite que un disco virtual en un dominio huésped sea accesible incluso si uno de los dominios de servicio se apaga. Por ejemplo, puede configurar una ruta múltiple de disco virtual para acceder a un archivo en un servidor de sistema de archivos de red (NFS). O puede utilizar esta configuración para acceder a un LUN desde un almacenamiento compartido que está conectado a más de un dominio de servicio. Así pues, cuando el dominio huésped accede al disco virtual, el controlador del disco virtual pasa por uno de los dominios de servicio para acceder al almacenamiento del componente posterior. Si el controlador del disco virtual no puede conectar con el dominio de servicio, el disco virtual intenta alcanzar el almacenamiento del componente posterior a través de un dominio de servicio diferente.

Nota – Desde la versión 2.0 del Oracle VM Server for SPARC, la característica de ruta múltiple del disco virtual puede detectar cuando el dominio de servicio no puede acceder al almacenamiento de componente posterior. En esta instancia, el controlador de disco posterior intenta acceder al almacenamiento del componente posterior por otra ruta.

Para habilitar la ruta múltiple de disco virtual, debe exportar el componente posterior del disco virtual de cada dominio de servicio y agregar el disco virtual al mismo grupo de ruta múltiple (mpgroup). El mpgroup se identifica con un nombre y se configura cuando exporta el componente posterior del disco virtual.

La siguiente imagen muestra la configuración de ruta múltiple de un disco virtual, que se usa como ejemplo en el procedimiento “[Configuración de la ruta múltiple de disco virtual](#)” en la [página 96](#). En este ejemplo, un grupo de ruta múltiple llamado foo se usa para crear un disco virtual, cuyo componente posterior es accesible desde dos dominios de servicio: primary y alternativo.

FIGURA 7-2 Configuración de ruta múltiple de disco virtual



▼ Configuración de la ruta múltiple de disco virtual

- 1 Exporte el componente posterior del disco virtual desde el dominio de servicio `primary`.

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

donde `backend-path1` es la ruta al componente posterior del disco virtual desde el dominio `primary`.

- 2 Exporte el mismo componente posterior del disco virtual desde el dominio de servicio `alternativo`.

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

donde `backend-path2` es la ruta al componente posterior del disco virtual desde el dominio `alternativo`.

Nota – `backend-path1` y `backend-path2` son rutas al mismo componente posterior del disco virtual, pero desde dos dominios diferentes (`primary` y `alternativo`). Estas rutas pueden ser iguales o diferentes, dependiente de la configuración de los dominios `primary` y `alternativo`. El usuario puede elegir el nombre del *volumen*. Puede ser igual o diferente para los dos comandos.

- 3 Exporte el disco virtual al dominio huésped.

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

Nota – A pesar de que el componente posterior del disco virtual se exporta varias veces a través de diferentes dominios de servicio, se asigna sólo un disco virtual al dominio huésped y lo asocia con el componente posterior del disco virtual a través de cualquiera de los dominios de servicio.

Más información Resultado de ruta múltiple de disco virtual

Después de haber configurado el disco virtual con ruta múltiple e iniciado el dominio huésped, el disco virtual accede a su componente posterior a través del dominio de servicio al que se ha asociado (dominio `primary` en este ejemplo). Si este dominio de servicio no está disponible, entonces en disco virtual intenta acceder al componente posterior a través de un dominio de servicio diferente que forme parte del mismo grupo de ruta múltiple.



Precaución – Cuando defina un grupo de ruta múltiple (`mpgroup`), asegúrese de que los componentes posteriores del disco virtual que forman parte del mismo `mpgroup` son efectivamente el mismo componente posterior del disco virtual. Si agrega componentes posteriores diferentes en el mismo `mpgroup`, podrá ver algún comportamiento inesperado, y puede potencialmente perder o corromper datos almacenados en los componentes posteriores.

CD, DVD e imágenes ISO

Puede exportar un disco compacto (CD) o un disco versátil digital (DVD) de la misma manera que exporta cualquier disco normal. Para exportar un CD o un DVD a un dominio huésped, exporte el segmento 2 del dispositivo CD o DVD como disco completo, esto es sin la opción `slice`.

Nota – No puede exportar la misma unidad CD o DVD, sólo puede exportar el CD o DVD que está dentro de la unidad CD o DVD. Por lo tanto, debe haber un CD o DVD en la unidad antes de la exportación. Asimismo, para poder exportar un CD o DVD, ese CD o DVD no puede estar en uso en el dominio de servicio. En especial, el servicio de sistema de archivos de administración de volumen, `volfs(7FS)` no debe usar el CD o DVD. Véase [“Exportación de un CD o DVD desde el dominio de servicio al dominio huésped” en la página 98](#) para instrucciones sobre cómo eliminar el dispositivo que está usando `volfs`.

Si tiene una imagen de organización internacional para la estandarización (ISO) de un CD o DVD almacenada en un archivo o volumen, y exporta ese archivo como disco completo entonces aparece como CD o DVD en el dominio huésped.

Cuando exporta un CD, DVD o una imagen ISO, aparece automáticamente como dispositivo de sólo lectura en el dominio huésped. En cualquier caso, no puede realizar cualquier operación de control de CD desde el dominio huésped; esto es, no puede iniciar, parar o expulsar un CD del dominio huésped. Si el CD, DVD o imagen ISO exportada es inicializable, el dominio huésped puede ser iniciado en el disco virtual correspondiente.

Por ejemplo, si exporta un DVD de instalación de SO de Solaris de Oracle, puede iniciar el dominio huésped en el disco virtual que corresponde a ese DVD e instalar el dominio huésped desde ese DVD. Para ello, cuando el dominio huésped alcance el mensaje ok, use el siguiente comando.

```
ok boot /virtual-devices@100/channel-devices@200/disk@n:f
```

Donde *n* es el índice de disco virtual que representa el DVD exportado.

Nota – Si exporta un DVD de instalación de SO de Solaris de Oracle e inicia un dominio huésped en el disco virtual que corresponde a ese DVD para instalar el dominio huésped, entonces no puede cambiar el DVD durante la instalación. Puede ser necesario omitir cualquier paso de la instalación que solicite un CD/DVD diferente, o deberá ofrecer una ruta alternativa para acceder a los medios solicitados.

▼ Exportación de un CD o DVD desde el dominio de servicio al dominio huésped

- 1 Desde el dominio de servicio, compruebe si el daemon de administración del volumen, **vol(1M)**, está en ejecución y en línea.

```
service# svcs volfs
STATE          STIME      FMRI
online         12:28:12  svc:/system/filesystem/volfs:default
```

- 2 Elija una de estas posibilidades.

- Si el daemon de administración del volumen no está en ejecución o en línea, vaya al paso 3.
- Si el daemon de administración del volumen está en ejecución y en línea, como en el ejemplo en el paso 1, realice los siguientes pasos:

- a. Edite el archivo `/etc/vold.conf` y agregue comentarios a la línea que comienza con las siguientes palabras.

```
use cdrom drive...
```

Véase la página de comando man `vold.conf(4)`.

- b. Introduzca el CD o DVD en la unidad de CD o DVD.

- c. Desde el dominio de servicio, reinicie el servicio del sistema de archivos de administración de volumen.

```
service# svcadm refresh volfs
service# svcadm restart volfs
```

- 3 Desde el dominio de servicio, busque la ruta del disco para el dispositivo CD-ROM.

```
service# cdrw -l
Looking for CD devices...
  Node                               Connected Device                               Device type
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
/dev/rdisk/c1t0d0s2 | MATSHITA CD-RW CW-8124 DZ13 | CD Reader/Writer
```

- 4 Exporte un dispositivo de disco de CD o DVD como disco completo.

```
primary# ldm add-vdsdev /dev/dsk/c1t0d0s2 cdrom@primary-vds0
```

5 Asigne el CD o DVD exportador al dominio huésped.

A continuación se muestra como asignar el CD o DVD exportado al dominio `ldg1`:

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

Más información Exportación de un CD o DVD varias veces

Un CD o DVD puede exportarse varias veces y asignarse a diferentes dominios huésped. Véase [“Exportación del componente posterior de un disco virtual varias veces” en la página 83](#) para más información.

▼ Exportación de una imagen ISO desde el dominio primary para la instalación en un dominio huésped

Este procedimiento muestra cómo exportar una imagen ISO desde un dominio `primary` y usarla para instalar un dominio huésped. Para este procedimiento se considera que tanto el dominio `primary` como el dominio huésped están configurados.

Por ejemplo, el siguiente `ldm list` muestra que tanto el dominio `primary` como el `ldom1` están configurados:

```
# 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 Agregue un dispositivo de servidor de disco virtual para exportar la imagen ISO.

En este ejemplo, la imagen ISO es `/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 Pare el dominio huésped.

En este ejemplo, el dominio lógico es `ldom1`.

```
# ldm stop-domain ldom1
LDom ldom1 stopped
```

3 Agregue el disco virtual para la imagen ISO al dominio lógico.

En este ejemplo, el dominio lógico es `ldom1`.

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

4 Reinicie el dominio huésped.

En este ejemplo, el dominio lógico es `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

En este ejemplo el comando `ldm list` muestra que el dominio `ldom1` se acaba de iniciar.

5 Conecte el dominio huésped.

```
# 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 Compruebe la existencia de la imagen ISO como disco virtual.

```
{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
```

En este ejemplo, el dispositivo que se acaba de agregar es

```
/virtual-devices@100/channel-devices@200/disk@1.
```

7 Inicie el dominio huésped que se debe instalar desde la imagen ISO.

En este ejemplo, inicie desde el segmento f del disco

```
/virtual-devices@100/channel-devices@200/disk@1.
```

```
{0} ok boot /virtual-devices@100/channel-devices@200/disk@1:f
```

Tiempo de espera de disco virtual

De manera predeterminada, si el dominio de servicio que ofrece acceso a un componente posterior de disco virtual está apagado, todas las E/S del dominio huésped al correspondiente disco virtual están bloqueadas. Las E/S se reinician automáticamente cuando el dominio de servicio está en funcionamiento y atiende solicitudes de E/S al componente posterior del disco virtual.

Sin embargo, en algunos casos los sistemas de archivos y aplicaciones no deben bloquear la operación de E/S, sino que deben fallar e informar del error si el dominio de servicio está apagado durante demasiado tiempo. Ahora se puede fijar un periodo de tiempo de espera de conexión para cada disco virtual, que puede usarse para establecer una conexión entre el cliente de disco virtual en un dominio huésped y el servidor de disco virtual en el dominio de servicio. Cuando finaliza el periodo de tiempo de espera, cualquier E/S pendiente y cualquier E/S nueva fallarán mientras el dominio de servicio esté apagado y la conexión entre el cliente del disco virtual y el servidor no se restablezca.

Este tiempo de espera puede configurarse de una de estas maneras:

- Uso del comando `ldm add-vdisk`.


```
ldm add-vdisk timeout=seconds disk-name volume-name@service-name ldom
```
- Usando el comando `ldm set-vdisk`.


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

Especifique el tiempo de espera en segundos. Si el tiempo de espera se fija en 0, se deshabilita el tiempo de espera y la E/S se bloquea mientras el dominio de servicio está apagado (esta es la configuración y comportamiento predeterminados).

Alternativamente, el tiempo de espera puede configurarse agregando la siguiente línea al archivo `/etc/system` en el dominio huésped.

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

Nota – Si se fija este ajuste, sobrescribe cualquier configuración de tiempo de espera realizada usando la CLI `ldm`. Asimismo, el ajuste fija el tiempo de espera para todos los discos virtuales en el dominio huésped.

Disco virtual y SCSI

Si un disco SCSI físico o un LUN se exporta como disco completo, el disco virtual correspondiente admite la interfaz de comando de SCSI de usuario, `uscsi(7I)` y las operaciones de control de disco multihost `mhd(7i)`. Otros discos virtuales, como los discos virtuales que usan un archivo o un volumen como componente posterior, no admiten estas interfaces.

Como consecuencia las aplicaciones o características del producto que usan los comando SCSI (como Solaris Volume Manager `metaset` o Clúster de Solaris de Oracle `shared devices`) pueden usarse en dominios huésped sólo con dispositivos virtuales que tengan un SCSI físico como componente posterior.

Nota – Las operaciones SCSI se ejecutan efectivamente por el dominio de servicio, que administra el disco SCSI físico o LUN usado como componente posterior del disco virtual. En especial, las reservas SCSI son realizadas por el dominio de servicio. Por lo tanto, las aplicaciones en ejecución en el dominio de servicio y en los dominios huésped no deben enviar comandos SCSI a los mismos discos SCSI físicos, en caso contrario, puede producir un estado inesperado del disco.

Disco virtual y el comando `format(1M)`

El comando `format(1M)` reconoce todos los discos virtuales que están presentes en un dominio. Sin embargo, para los discos virtuales que se exportan como discos de segmento único, el comando `format` no puede cambiar la tabla de particiones del disco virtual. Los comandos como `label` fallarán a menos que intente escribir una etiqueta de disco parecida a la que ya está asociada con el disco virtual.

Los discos virtuales cuyos componente posterior son discos SCSI admiten todos los subcomandos `format(1M)`. Los disco virtuales cuyos componentes posteriores no son discos SCSI no admiten algunos subcomandos `format(1M)`, como `repair` y `defect`. En ese caso, el comportamiento de `format(1M)` es parecido al comportamiento de los discos de controlador electrónico integrado (IDE).

Uso de ZFS con discos virtuales

Esta sección describe el uso de Zettabyte File System (ZFS) para almacenar componentes posteriores de discos virtuales exportados a dominios huésped. ZFS ofrece una solución conveniente y potente para crear y administrar componentes posteriores de discos virtuales. ZFS habilita:

- Almacenamiento de imágenes de disco en volúmenes ZFS o archivos ZFS
- Uso de instantáneas para realizar copias de seguridad de imágenes de disco
- Uso de clones para duplicar imágenes de disco y ofrecer dominios adicionales

Consulte *Guía de administración de Oracle Solaris ZFS* para más información sobre el uso del ZFS.

En las siguientes descripciones y ejemplos, el dominio `primary` también es el dominio de servicio donde se almacenan las imágenes del disco.

Configuración de un grupo de ZFS en un dominio de servicio

Para almacenar las imágenes del disco, primero cree un grupo de almacenamiento de ZFS en el dominio de servicio. Por ejemplo, este comando crea el grupo de almacenamiento de ZFS `ldmpool` que contiene el disco `c1t50d0` en el dominio `primary`.

```
primary# zpool create ldmpool c1t50d0
```

Almacenamiento de imágenes de disco con ZFS

El siguiente comando crea una imagen del disco para el dominio huésped `ldg1`. Se crea un sistema de archivos ZFS para este dominio huésped, y todas las imágenes de disco de este dominio huésped se almacenarán en ese sistema de archivos.

```
primary# zfs create ldmpool/ldg1
```

Las imágenes de disco se pueden guardar en volúmenes ZFS o en archivos ZFS. La creación de un volumen ZFS, sea cual sea su tamaño, es rápida usando el comando `zfs create -V`. Por otro lado, los archivos ZFS deben crear usando el comando `mkfile`. Puede ser necesario algún tiempo para realizar el comando, especialmente si el archivo que se debe crear es un poco grande, que a menudo es el caso cuando se crea una imagen de disco.

Los volúmenes ZFS y los archivos ZFS pueden aprovecharse de las características de ZFS como las instantáneas y la clonación, pero un volumen ZFS es un pseudo dispositivo mientras que un archivo ZFS es un archivo normal.

Si la imagen de disco debe usarse como un disco virtual en el que se instala el SO, la imagen del disco debe ser lo suficientemente amplia como para acomodar los requisitos de instalación del SO. El tamaño depende de la versión del SO y del tipo de instalación realizada. Si instala el SO de Solaris de Oracle, puede usar un tamaño de disco de 20 Gbytes para acomodar cualquier tipo de instalación del SO de Solaris de Oracle.

Ejemplos de almacenamiento de imágenes de disco con ZFS

Los siguientes ejemplos:

1. Cree una imagen de 20 gigabyte en un volumen o archivo ZFS.
2. Exporte el volumen o archivo ZFS como disco virtual. La sintaxis para exportar un volumen o archivo ZFS es la misma, pero la ruta al componente posterior es diferente.
3. Asigne el volumen o archivo ZFS a un dominio huésped.

Cuando se inicia el dominio huésped, el volumen o archivo ZFS aparece como un disco virtual en el que puede instalarse el SO de Solaris de Oracle.

▼ Cree una imagen de disco usando un volumen ZFS

- Por ejemplo, cree una imagen de disco de 20 gigabytes en un volumen ZFS.

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

▼ Cree una imagen de disco usando un archivo ZFS

- Por ejemplo, cree una imagen de disco de 20 gigabytes en un volumen ZFS.

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

▼ Exporte el volumen ZFS

- Exporte el volumen ZFS como disco virtual.

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

▼ Exporte el archivo XFS

- Exporte el archivo ZFS como disco virtual.

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

▼ Exportación del volumen o archivo ZFS a un dominio huésped

- Asigne el volumen o archivo ZFS a un dominio huésped, en este ejemplo, ldg1.

```
primary# ldm add-vdisk disk0 ldg1_disk0@primary-vds0 ldg1
```

Creación de una instantánea de la imagen del disco

Cuando la imagen del disco se almacena en un volumen ZFS o un archivo ZFS, puede crear instantáneas de esta imagen de disco usando el comando de instantánea de ZFS.

Antes de crear una instantánea de la imagen del disco, asegúrese de que el disco no está actualmente en uso en el dominio huésped para asegurarse de que los datos actualmente almacenados en la imagen del disco son coherentes. Hay varias maneras de asegurarse de que un disco no está en uso en el dominio huésped. Puede realizar una de estas acciones:

- Parar y desenlazar el dominio huésped. Esta es la solución más segura, y es la única solución disponible si desea crear una instantánea de una imagen de disco usada como disco de inicio para un dominio huésped.
- Como alternativa, puede desmontar cualquier segmento de disco del que desee realizar una instantánea en el dominio huésped, y asegurarse de que no hay ningún segmento en uso en el dominio huésped.

En este ejemplo, debido al diseño del ZFS, el comando para crear una instantánea del disco es el mismo si la imagen se almacena en un volumen ZFS o en un archivo ZFS.

▼ Creación de una instantánea de la imagen del disco

- Cree una instantánea de la imagen del disco que se ha creado para el dominio ldg1, por ejemplo.

```
primary# zfs snapshot ldmpool/ldg1/disk0@version_1
```

Uso de un clon para ofrecer un dominio nuevo

Una vez ha creado la instantánea de la imagen del disco, puede duplicar esta imagen del disco usando un comando de clonación de ZFS. Entonces la imagen clonada puede asignarse a otro dominio. La clonación de la imagen de un disco de inicio rápidamente crea un disco de inicio para un nuevo dominio huésped sin tener que realizar todo el proceso de instalación del SO de Solaris de Oracle.

Por ejemplo, si el `disk0` creado era el disco de inicio del dominio `ldg1`, realice los siguientes pasos para clonar ese disco para crear un disco de inicio para el dominio `ldg2`.

```
primary# zfs create ldmpool/ldg2
primary# zfs clone ldmpool/ldg1/disk0@version_1 ldmpool/ldg2/disk0
```

Entonces `ldmpool/ldg2/disk0` puede exportarse como disco virtual y asignarse al nuevo dominio `ldg2`. El dominio `ldg2` puede iniciar directamente desde esa imagen de disco sin tener que pasar por el proceso de instalación del SO.

Clonación de la imagen de un disco de inicio

Cuando se clona una imagen de disco de inicio, la nueva imagen es exactamente igual que el disco de inicio original, y contiene cualquier información que se haya guardado en el disco de inicio antes de que la imagen sea clonada, como el nombre del host, dirección IP, la tabla del sistema de archivos montados, o cualquier configuración o ajuste del sistema.

Dado que la tabla del sistema de archivo montado es la misma en la imagen del disco de inicio original y en la imagen del disco clonado, la imagen del disco clonado debe asignarse al nuevo dominio en el mismo orden que estaba en el dominio original. Por ejemplo, si la imagen del disco de inicio estaba asignada como el primer disco del dominio original, entonces la imagen del disco clonado tiene que ser asignado como el primer disco del nuevo dominio. En caso contrario, el nuevo dominio no puede iniciarse.

Si el dominio original estaba configurado con una dirección IP estática, entonces el nuevo dominio usa la imagen clonada con la misma dirección IP. En ese caso, puede cambiar la configuración de red de nuevo dominio usando el comando `sys-unconfig(1M)`. Para evitar ese problema también puede crear una instantánea de una imagen de disco de un sistema no configurado.

Si el dominio original estaba configurado con el protocolo de configuración dinámica de host (DHCP), entonces el nuevo dominio que usa la imagen clonada también usa DHCP. En este caso, si no necesita cambiar la configuración de red del nuevo dominio porque recibe automáticamente una dirección IP y la configuración de red cuando inicia.

Nota – El id e host de un dominio no se guarda en el disco de inicio, sino que es asignado por el Administrador de Dominios lógicos cuando crea un dominio. Por lo tanto, cuando clona una imagen de disco, el nuevo dominio no guarda el id de host del dominio original.

▼ **Creación de una instantánea de una imagen de disco de un sistema no configurado**

- 1 Enlace e inicie el dominio original.
- 2 Ejecute el comando `sys-unconfig`.
- 3 Después de haber completado el comando `sys-unconfig`, el dominio para.
- 4 Pare y desenchace el dominio; *no* lo reinicie.
- 5 Tome una instantánea de la imagen del disco de inicio del dominio.

Por ejemplo:

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

En este momento tiene la instantánea de la imagen del disco de inicio de un sistema no configurado.

- 6 Clone esta imagen para crear un nuevo dominio que, cuando se inicia por primera vez, solicita la configuración del sistema.

Uso de los administradores de volumen en el entorno Dominios lógicos

Esta sección describe el uso de los administradores de volumen en un entorno Dominios lógicos.

Uso de discos virtuales para administradores de volúmenes

Cualquier volumen Zettabyte File System (ZFS), Solaris Volume Manager o Veritas Volume Manager (VxVM) puede ser exportado desde un dominio de servicio a un dominio huésped como disco virtual. Un volumen puede ser exportado con disco de segmento único (si la opción `slice` se especifica con el comando `ldm add -vdsdev`) o como disco completo.

Nota – El resto de esta sección usa un volumen Solaris Volume Manager como ejemplo. Sin embargo, la discusión también afecta a los volúmenes ZFS y VxVM.

Los siguientes ejemplos muestran como exportar un volumen como disco de segmento único.

El disco virtual en el dominio huésped (por ejemplo, `/dev/dsk/c0d2s0`) se asigna directamente al volumen asociado (por ejemplo, `/dev/md/dsk/d0`) y los datos que se guardan en el disco virtual del dominio huésped son directamente guardados en el volumen asociado sin metadatos adicionales. Así pues, también puede accederse directamente a los datos guardados en el disco virtual del dominio huésped desde el dominio de servicio a través del volumen asociado.

Ejemplos

- Si el volumen Solaris Volume Manager `d0` se exporta del dominio `primary` a `domain1`, entonces la configuración de `domain1` requiere algunos pasos adicionales.

```
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
```

- Después de que `domain1` esté enlazado e iniciado, el volumen exportado aparece como `/dev/dsk/c0d2s0`, por ejemplo, y puede usarlo.

```
domain1# newfs /dev/rdisk/c0d2s0
domain1# mount /dev/dsk/c0d2s0 /mnt
domain1# echo test-domain1 > /mnt/file
```

- Después de haber parado y desenlazado `domain1`, puede accederse directamente a los datos guardados en el disco virtual de `domain1` a través del volumen de Solaris Volume Manager `d0`.

```
primary# mount /dev/md/dsk/d0 /mnt
primary# cat /mnt/file
test-domain1
```

Uso de discos virtuales para Solaris Volume Manager

Cuando se usa un volumen RAID o Solaris Volume Manager reflejado como disco virtual por otro dominio, tiene que ser exportando sin fijar la opción (`excl`) exclusiva. En caso contrario, si hay un fallo de los componentes del volumen Solaris Volume Manager, la recuperación del volumen Solaris Volume Manager que usa el comando `metareplace` o usa un disco preparado y conectado no comienza. El comando `metastat` ve el volumen como resincronizando, pero esta operación no está en curso.

Por ejemplo, `/dev/md/dsk/d0` es un volumen RAID Solaris Volume Manager exportado como disco virtual con la opción `excl` a otro dominio, y `d0` se configura con algunos dispositivos preparados y conectados. Si un componente de `d0` falla, Solaris Volume Manager sustituye el volumen que ha fallado con uno preparado y conectado y vuelve a sincronizar el volumen Solaris Volume Manager. Sin embargo, no comienza la resincronización. El volumen se indica como resincronizando, pero la resincronización no avanza.

```
# 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/c10t600C0FF000000000015153295A4B100d0s1 330      No    Resyncing Yes
```

En esta situación, el dominio que usa el volumen Solaris Volume Manager como disco virtual debe pararse y desenlazarse para completar la resincronización. Entonces el volumen Solaris Volume Manager puede resincronizarse usando el comando `metasync`.

```
# metasync d0
```

Uso de discos virtual cuando VxVM está instalado

Cuando Veritas Volume Manager (VxVM) está instalado en su sistema, y si Veritas Dynamic Multipathing (DMP) está habilitado en un disco físico o partición que desee exportar como disco virtual, tiene que exportar ese disco o partición sin fijar la opción (no determinada) `excl`. En caso contrario, recibe un error en `/var/adm/messages` mientras enlaza un dominio que usa dicho disco.

```
vd_setup_vd(): ldi_open_by_name(/dev/dsk/c4t12d0s2) = errno 16
vds_add_vd(): Failed to add vdisk ID 0
```

Puede comprobar si Veritas DMP está habilitado comprobando la información de ruta múltiple en la salida del comando `vxdisk list`; por ejemplo:

```
# 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
```

Alternativamente, si Veritas DMP está habilitado en un disco o en un segmento que desee exportar como disco virtual con la opción `excl` fijada, puede deshabilitar DMP usando el comando `vxdatapadm`. Por ejemplo:

```
# vxdatapadm -f disable path=/dev/dsk/c4t12d0s2
```

Uso de administradores de volumen para discos virtuales

Esta sección describe el uso de administradores de volumen para discos virtuales.

Uso de ZFS para discos virtuales

Cualquier disco virtual puede usarse con ZFS. Un grupo de almacenamiento ZFS (`zpool`) puede ser importado en cualquier dominio que crea todos los dispositivos de almacenamiento que forman parte de este `zpool`, sin que importe si el dominio ve todos esos dispositivos como dispositivos virtuales o reales.

Uso de Solaris Volume Manager para disco virtuales

Cualquier disco virtual puede usarse en el conjunto del disco local del Solaris Volume Manager. Por ejemplo, un disco virtual puede usarse para el almacenamiento de la base de datos de estado de metadispositivo Solaris Volume Manager, `metadb(1M)`, del conjunto de disco local o para crear volúmenes de Solaris Volume Manager en el conjunto de disco local.

Cualquier disco virtual cuyo componente posterior es un disco SCSI puede usarse en un conjunto de disco compartido de Solaris Volume Manager, `metaset(1M)`. Los discos virtuales cuyos componentes posteriores no son discos SCSI no pueden agregarse a un conjunto de discos compartido Solaris Volume Manager. Si se intenta agregar un disco virtual cuyo componente posterior no es un disco SCSI a un conjunto de disco compartido Solaris Volume Manager falla con un error parecido al siguiente.

```
# metaset -s test -a c2d2
metaset: domain1: test: failed to reserve any drives
```

Uso de VxVM para discos virtuales

Para asistencia técnica de VxVM en dominios huésped, consulte la documentación VxVM de Symantec.

Uso de las redes virtuales

Este capítulo describe cómo usar una red virtual con el software del Oracle VM Server for SPARC y trata sobre los siguientes temas:

- “Introducción a una red virtual” en la página 111
- “Conmutador virtual” en la página 112
- “Dispositivo de red virtual” en la página 112
- “Administración de conmutador virtual” en la página 114
- “Administración del dispositivo de red virtual” en la página 116
- “Identificador del dispositivo virtual y nombre de interfaz de red” en la página 118
- “Asignación de direcciones MAC automática o manualmente” en la página 120
- “Uso de adaptadores de red con Dominios lógicos” en la página 123
- “Configuración del conmutador virtual y dominio de servicio para NAT y enrutamiento” en la página 124
- “Configuración de IPMP en un entorno Dominios lógicos” en la página 126
- “Uso de etiquetado VLAN” en la página 133
- “Uso de E/S híbridas de NIU” en la página 136
- “Uso de la agregación de enlaces con un conmutador virtual” en la página 139
- “Configuración de marcos Jumbo” en la página 141

Introducción a una red virtual

Una red virtual permite que los dominios se comuniquen unos con otros sin usar las redes físicas exteriores. Una red virtual también puede permitir el uso de la misma interfaz de red física para acceder a la red física y comunicarse con sistemas remotos. Una red virtual se crea con un conmutador virtual al que se pueden conectar dispositivos de red virtual.

Conmutador virtual

Un conmutador virtual (*vsw*) es un componente que se ejecuta en un dominio de servicio y está gestionado por un controlador de conmutador virtual. El conmutador virtual puede conectarse a algunos dominios huésped para habilitar las comunicaciones de red entre estos dominios. Además, si el conmutador virtual también está asociado con una interfaz de red física, esto permite la comunicación de red entre dominios huésped y la red física sobre la interfaz de red física. Un conmutador virtual también tiene una interfaz de red, *vsw*n**, que permite que el dominio de servicio se comuniquen con los otros dominios conectados a dicho conmutador virtual. Puede usarse como cualquier interfaz de red normal y configurarse con el comando `ifconfig(1M)`.

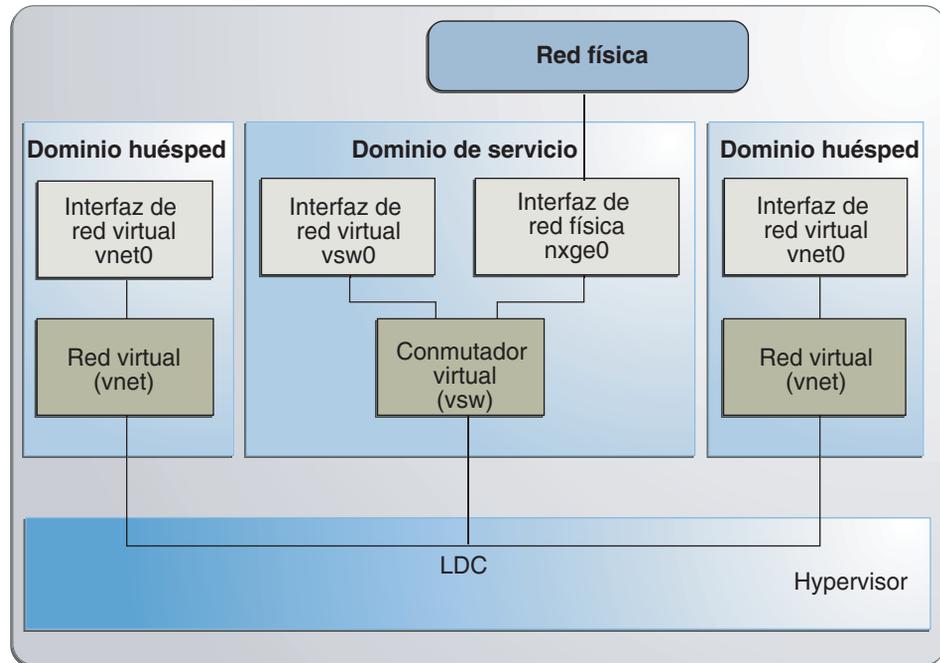
Nota – Cuando se agrega un conmutador virtual a un dominio de servicio, la interfaz de red no se instala. Así que de manera predeterminada el dominio de servicio no puede comunicarse con los dominios huéspedes conectados al conmutador virtual. Para habilitar las comunicaciones de red entre dominios huésped y el dominio de servicio, la interfaz de red del conmutador virtual asociado debe estar instalada y configurada en el dominio de servicio. Véase “[Habilitación de las funciones de red entre dominio de control/servicio y otros dominios](#)” en la página 53 para conocer las instrucciones.

Dispositivo de red virtual

Un dispositivo de red virtual (*vnet*) es un dispositivo virtual que está definido en un dominio conectado a un conmutador virtual. Un dispositivo de red virtual está administrado por el controlador de red virtual, y está conectado a una red virtual a través del hypervisor usando canales de dominio lógico (LDC).

Un dispositivo de red virtual puede usarse como interfaz de red con el nombre *vnet*n**, que puede usarse como cualquier interfaz de red normal y configurarse con el comando `ifconfig(1M)`.

FIGURA 8-1 Configuración de una red virtual



A continuación se incluye una explicación para el ejemplo en la [Figura 8-1](#).

- El conmutador virtual en el dominio de servicio está conectado a los dominios huéspedes. Esto permite que los dominios huéspedes se comuniquen unos con otros.
- El conmutador virtual también está conectado a la interfaz de red física `nxge0`. Esto permite que los dominios huéspedes se comuniquen con la red física.
- La interfaz de red del conmutador virtual `vsw0` está instalada en el dominio de servicio, esto permite la comunicación con el dominio de servicio.
- La interfaz de red del conmutador virtual `vsw0` en el dominio de servicio puede configurarse usando el comando `ifconfig(1M)`.
- Las interfaces de redes virtuales `vnet0` en los dominios huéspedes pueden configurarse usando el comando `ifconfig(1M)`.

Básicamente el conmutador virtual se comporta como un conmutador de red física normal e intercambia paquetes de red entre diferentes sistemas, como dominios huéspedes, dominios de servicio y redes físicas a los que está conectado.

Administración de conmutador virtual

Esta sección describe el agregado de un conmutador virtual a un dominio, la configuración de opciones para un conmutador virtual y la eliminación de un conmutador virtual.

▼ Agregación de conmutador virtual

● Use la siguiente sintaxis de comando para agregar un conmutador virtual.

```
# ldm add-vsw [default-vlan-id=vlan-id] [pvid=[port-vlan-id]] [vid=vlan-id1,vlan-id2,...]
  [linkprop=phys-state] [mac-addr=num] [net-dev=device] [mode=sc] [mtu=size]
  [id=switch-id] vswitch-name ldom
```

Donde:

- `default-vlan-id=vlan-id` especifica la red de área local virtual predeterminada (VLAN) a la que un conmutador virtual y sus dispositivos de redes virtuales asociados pertenecen implícitamente, en modo sin etiquetas. Sirve como id de VLAN de puerto predeterminado (`pvid`) del conmutador virtual y los dispositivos de red virtual. Sin esta opción, el valor predeterminado de esta propiedad es 1. Normalmente no es necesario utilizar esta opción. Se ofrece sólo como manera de cambiar el valor predeterminado de 1. Véase [“Uso de etiquetado VLAN” en la página 133](#) para más información.
- `pvid=port-vlan-id` especifica la VLAN de la cual el conmutador virtual necesita ser miembro, en modo sin etiquetas. Véase [“Uso de etiquetado VLAN” en la página 133](#) para más información.
- `vid=vlan-id` especifica una o varias VLANs de las que el conmutador virtual necesita ser un miembro, en modo etiquetado. Véase [“Uso de etiquetado VLAN” en la página 133](#) para más información.
- `linkprop=phys-state` especifica si el dispositivo virtual informa de su estado de enlace basándose en el dispositivo de red físico subyacente. Cuando se especifica `linkprop=phys-state` en la línea de comando, el estado del enlace del dispositivo virtual refleja el estado del enlace físico. De manera predeterminada, el estado del enlace del dispositivo virtual no refleja el estado del enlace físico.

Especifique esta opción para usar IPMP basados en enlaces. Véase [“Uso de IPMP basado en enlace en funciones de redes virtuales de Dominios lógicos” en la página 128](#).

- `mac-addr=num` es la dirección MAC que será usada por este conmutador. Este número debe estar en notación de octetos estándar; por ejemplo, 80:00:33:55:22:66. Si no especifica una dirección MAC, se asigna automáticamente al conmutador una dirección del rango de direcciones MAC públicas asignadas al Administrador de Dominios lógicos. Véase [“Asignación de direcciones MAC automática o manualmente” en la página 120](#) para más información.
- `net-dev=device` es la ruta al dispositivo de red en el que opera este conmutador.

- `mode=sc` habilita la asistencia de funciones de redes virtuales para el procesamiento prioritario de los paquetes de respuesta de Clúster de Solaris de Oracle en un entorno Dominios lógicos. Las aplicaciones como Clúster de Solaris de Oracle deben asegurarse de que los paquetes de respuesta de prioridad alta no se caen por redes virtuales y dispositivos de conmutador congestionados. Esta opción da prioridad a los marcos de respuesta de Clúster de Solaris de Oracle y se asegura de que son transmitidos de manera fiable.
 Debe fijar esta opción cuando ejecute Clúster de Solaris de Oracle en un entorno de Dominios lógicos y use los dominios huésped como nodos de Clúster de Solaris de Oracle. No fije esta opción cuando no esté ejecutando el software del Clúster de Solaris de Oracle en dominios huésped, ya que podría tener un impacto en el rendimiento de la red.
- `mtu=tamaño` especifica la unidad de transmisión máxima (MTU) de un dispositivo de conmutador virtual. Los valores válidos se sitúan en el rango de 1500-16000.
- `id=switch-id` es el id de un dispositivo de conmutador virtual nuevo. De manera predeterminada, estos valores de ID se generan automáticamente, así que debe configurar esta propiedad si necesita que coincida con un nombre de un dispositivo existente en el SO. Véase [“Identificador del dispositivo virtual y nombre de interfaz de red” en la página 118](#)
- `vswitch-name` es el nombre único del conmutador que se exporta como servicio. Los clientes (red) se pueden adjuntar a este servicio.
- `ldom` especifica el dominio lógico al que se agrega un conmutador virtual.

▼ Configuración de opciones para un conmutador virtual existente

- Use la siguiente sintaxis de comando para fijar las opciones de un conmutador virtual que ya existe.

```
# ldm set-vsw [pvid=[port-vlan-id]] [vid=[vlan-id1,vlan-id2,...]] [mac-addr=num]
  [linkprop=[phys-state]] [net-dev=[device]] [mode=[sc]] [mtu=[size]] vswitch-name
```

Donde:

- `mode=` (en blanco) para el procesamiento especial de los paquetes de respuesta del Clúster de Solaris de Oracle.
- En caso contrario, los argumentos del comando son los mismos que los descritos en [“Agregación de conmutador virtual” en la página 114](#).

▼ Eliminación de un interruptor virtual

- Use la siguiente sintaxis de comando para eliminar un conmutador virtual.

```
# ldm rm-vsw [-f] vswitch-name
```

Donde:

- -f intenta forzar la eliminación de un conmutador virtual. La eliminación puede fallar.
- *vswitch-name* es el nombre del conmutador que se elimina como servicio.

Administración del dispositivo de red virtual

Esta sección describe la agregación de un dispositivo de red virtual a un dominio, la configuración de opciones para un dispositivo de red virtual existente y la eliminación de un dispositivo de red virtual.

▼ Agregado de un dispositivo de red virtual

- Use la siguiente sintaxis de comando para agregar un dispositivo de red virtual.

```
# ldm add-vnet [mac-addr=num] [mode=hybrid] [pvid=[port-vlan-id]]
  [linkprop=phys-state] [vid=vlan-id1,vlan-id2,...] [mtu=size] [id=network-id]
  if-name vswitch-name ldom
```

Donde:

- *mac-addr=num* es la dirección MAC para este dispositivo de red. Este número debe estar en notación de octetos estándar; por ejemplo, `80:00:33:55:22:66`. Véase [“Asignación de direcciones MAC automática o manualmente” en la página 120](#) para más información.
- *mode=hybrid* para solicitar que el sistema usa las E/S híbridas de NIU en este vnet si es posible. Si no es posible, el sistema vuelve a E/S virtual. Este modo híbrido se considera una reconfiguración retrasada si se fija en un vnet activo. Véase [“Uso de E/S híbridas de NIU” en la página 136](#) para más información.
- *pvid=port-vlan-id* especifica la VLAN de la cual el dispositivo de red virtual necesita ser miembro, en modo sin etiquetas. Véase [“Uso de etiquetado VLAN” en la página 133](#) para más información.

- *linkprop=phys-state* especifica si el dispositivo de red virtual informa de su estado de enlace basándose en el dispositivo de red físico subyacente. Cuando se especifica *linkprop=phys-state* en la línea de comando, el estado de enlace del dispositivo de red virtual refleja el estado del enlace físico. De manera predeterminada, el estado del enlace del dispositivo de red virtual no refleja el estado del enlace físico.

Especifique esta opción para usar IPMP basados en enlaces. Véase [“Uso de IPMP basado en enlace en funciones de redes virtuales de Dominios lógicos” en la página 128](#).

- *vid=vlan-id* especifica una o varias VLANs de las que el dispositivo de red virtual necesita ser un miembro, en modo etiquetado. Véase [“Uso de etiquetado VLAN” en la página 133](#) para más información.
- *mtu=tamaño* especifica la unidad de transmisión máxima (MTU) de un dispositivo de red virtual. Los valores válidos se sitúan en el rango de 1500-16000.

- `id=network-id` es el id de un nuevo dispositivo de red virtual. De manera predeterminada, estos valores de ID se generan automáticamente, así que debe configurar esta propiedad si necesita que coincida con un nombre de un dispositivo existente en el SO. Véase [“Identificador del dispositivo virtual y nombre de interfaz de red” en la página 118](#)
- `if-name`, nombre de la interfaz, es un nombre único en el dominio lógico, asignado a la instancia del dispositivo de red virtual como referencia en subsiguientes comandos `ldm set-vnet` o `ldm rm-vnet`.
- `vswitch-name` es el nombre de un servicio de red existente (conmutador virtual) al que conectarse.
- `ldom` especifica el dominio lógico al que agregar el dispositivo de red virtual.

▼ Opciones configuradas para un dispositivo de red virtual existente

- Use la siguiente sintaxis de comando para fijar las opciones de un dispositivo de red virtual que ya existe.

```
# ldm set-vnet [mac-addr=num] [vswitch=vswitch-name] [mode=[hybrid]]
  [pvid=[port-vlan-id]] [linkprop=[phys-state]] [vid=[vlan-id1,vlan-id2,...]]
  [mtu=[size]] if-name ldom
```

Donde:

- `mode=` (en blanco) deshabilita la E/S híbrida de NIU.
- `if-name`, nombre de interfaz, es el nombre único asignado al dispositivo de red virtual que desea configurar.
- `ldom` especifica el dominio lógico del que eliminar el dispositivo de red virtual.
- En caso contrario, los argumentos del comando son los mismos que los descritos en [“Agregado de un dispositivo de red virtual” en la página 116](#).

▼ Eliminación del dispositivo de red virtual

- Use la siguiente sintaxis de comando para eliminar un dispositivo de red virtual.

```
# ldm rm-vnet [-f] if-name ldom
```

Donde:

- `-f` intenta forzar la eliminación de un dispositivo de red virtual de un dominio lógico. La eliminación puede fallar.
- `if-name`, nombre de interfaz, es el nombre único asignado al dispositivo de red virtual que desea eliminar.

- *ldom* especifica el dominio lógico del que eliminar el dispositivo de red virtual.

Identificador del dispositivo virtual y nombre de interfaz de red

Cuando agrega un conmutador virtual o dispositivo de red virtual a un dominio, puede especificar el número de dispositivo configurando la propiedad *id*.

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

Cada conmutador virtual y dispositivo de red virtual tiene un número de dispositivo único que se asigna cuando el dominio está enlazado. Si un conmutador virtual o dispositivo de red virtual se ha agregado con un número de dispositivo explícito (configurando la propiedad *id*), se usa el número de dispositivo especificado. En caso contrario, el sistema asignará automáticamente el número de dispositivo más bajo posible. En este caso, el número de dispositivo asignado depende de cómo han sido agregados al sistema el conmutador virtual o los dispositivos de red virtual. El número de dispositivo eventualmente asignado a un conmutador virtual o dispositivo de red virtual se puede ver en la salida del comando `ldm list-bindings` cuando un dominio está enlazado.

El siguiente ejemplo muestra que el dominio `primary` tiene un conmutador virtual, `primary-vsw0`. El conmutador virtual tiene el número de dispositivo 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
...
```

El siguiente ejemplo muestra que el dominio `ldg1` tiene dos dispositivos de red virtual: `vnet` y `vnet1`. El dispositivo `vnet` tiene un número de dispositivo de `0` (`network@0`) y el dispositivo `vnet1` tiene un número de dispositivo 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
...
```

Cuando un dominio con un conmutador virtual está ejecutando el SO de Solaris de Oracle, el conmutador virtual tiene una interfaz de red, `vswN`. En cualquier caso, el número de la interfaz de red del conmutador virtual, *N*, no es necesariamente el mismo que el número del dispositivo del conmutador virtual, *n*.

De manera parecida, cuando un dominio de red virtual está ejecutando el SO de Solaris de Oracle, el dispositivo de red virtual tiene una interfaz de red, `vnetN`. En cualquier caso, el número de la interfaz de red del dispositivo de red virtual, `N`, no es necesariamente el mismo que el número del dispositivo del dispositivo de red virtual, `n`.



Precaución – El SO de Solaris de Oracle conserva la asignación entre el nombre de la interfaz de red y un conmutador virtual o red virtual basada en el número del dispositivo. Si un número de dispositivo no se asigna explícitamente a un conmutador virtual o a un dispositivo de red virtual, el número de dispositivo puede cambiar cuando el dominio se desenlaza y se vuelve a enlazar posteriormente. En este caso, el nombre de la interfaz de red asignado por el SO en ejecución en el dominio también puede cambiar e interrumpir la configuración existente en el sistema. Esto puede suceder cuando un conmutador virtual o interfaz de red virtual se eliminan de la configuración del dominio.

No puede usar los comandos `ldm list -*` para determinar directamente el nombre de la interfaz de red de SO de Solaris de Oracle que corresponde a un conmutador virtual o dispositivo de red virtual. En cualquier caso, puede obtener esta información usando una combinación de la salida del comando `ldm list -l` y de las entradas bajo `/devices` en el SO de Solaris de Oracle.

▼ Búsqueda del nombre de la interfaz de red de SO de Solaris de Oracle

En este procedimiento de ejemplo, el dominio huésped `ldg1` contiene dos dispositivos de red virtual, `net - a` y `net - c`. Para encontrar el nombre de la interfaz de red de SO de Solaris de Oracle en `ldg1` que corresponde a `net - c`, haga lo siguiente. Este ejemplo también muestra las diferencias si está buscando el nombre de la interfaz de red de un conmutador virtual en vez de un dispositivo de red virtual.

1 Use el comando `ldm` para saber el número de dispositivo de red virtual para `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
...
```

El número de dispositivo de red virtual para `net - c` es 2 (`network@2`).

Para determinar el nombre de la interfaz de red de un conmutador virtual, busque el número del dispositivo del conmutador virtual, `n` como `switch@n`.

- 2 **Para encontrar la interfaz de red correspondiente en ldg1, inicie sesión en ldg1 y busque la entrada para este número de dispositivo bajo /devices.**

```
# uname -n
ldg1
# find /devices/virtual-devices@100 -type c -name network@2\*
/devices/virtual-devices@100/channel-devices@200/network@2:vnet1
```

El nombre de la interfaz de red es la parte de la entrada después del punto y coma, esto es, vnet1.

Para determinar el nombre de la interfaz de red de un conmutador virtual, sustituya el argumento en la opción `-name` con `virtual-network-switch@n*`. Después, busque la interfaz de red con el nombre `vswN`.

- 3 **Instale vnet1 para comprobar que tiene la dirección MAC 00:14:4f:f8:dd:68 tal y como se muestra en la salida `ldm list -l` para `net-c` en el paso 1.**

```
# 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
```

Asignación de direcciones MAC automática o manualmente

Debe tener suficientes direcciones de control de acceso a medios (MAC) para asignar al número de dominios lógicos, conmutadores virtuales y redes virtuales que va a usar. Puede hacer que el Administrador de Dominios lógicos asigne automáticamente las direcciones MAC a un dominio lógico, una red virtual (vnet) y un conmutador virtual (vsw), o puede asignar manualmente las direcciones MAC del grupo de direcciones MAC asignadas. Los subcomandos `ldm` que fijan las direcciones MAC son `add-domain`, `add-vsw`, `set-vsw`, `add-vnet` y `set-vnet`. Si no especifica una dirección MAC en estos subcomandos, el Administrador de Dominios lógicos asigna una automáticamente.

La ventaja de que el Administrador de Dominios lógicos asigne las direcciones MAC es que utiliza el bloque de direcciones MAC específico para el uso con dominios lógicos. Asimismo, el Administrador de Dominios lógicos detecta y evita colisiones de direcciones MAC con otras instancias de Administrador de Dominios lógicos en la misma subred. Esto le evita tener que administrar manualmente el grupo de direcciones MAC.

La asignación de direcciones MAC se efectúa en cuando se crea un dominio lógico o se configura un dispositivo de red en un dominio. Además, la asignación es persistente hasta que el dispositivo, o el mismo dominio lógico, se elimina.

Rango de las direcciones MAC asignadas a Dominios lógicos

Dominios lógicos se ha asignado el siguiente bloque de direcciones MAC de 512K:

00:14:4F:F8:00:00 ~ 00:14:4F:FF:FF:FF

Las direcciones de 256K inferiores son usadas por el Administrador de Dominios lógicos para la *asignación automática de direcciones MAC* y *no puede* solicitar manualmente una dirección en este rango:

00:14:4F:F8:00:00 - 00:14:4F:FB:FF:FF

Puede usar la mitad superior de este rango para la *asignación de direcciones MAC manual*:

00:14:4F:FC:00:00 - 00:14:4F:FF:FF:FF

Algoritmo de asignación automática

Si no especifica una dirección MAC al crear un dominio lógico y dispositivo de red, el Administrador de Dominios lógicos otorga automáticamente y asigna una dirección MAC a ese dominio lógico o dispositivo de red. Para obtener esta dirección MAC, el Administrador de Dominios lógicos de manera repetitiva intenta seleccionar una dirección y después comprueba las posibles colisiones.

Antes de seleccionar una dirección potencia, el Administrador de Dominios lógicos primero controla si tiene una dirección recientemente liberada automáticamente asignada guardada en la base de datos para este objetivo (véase [“Direcciones MAC liberadas” en la página 122](#)). Si es así, el Administrador de Dominios lógicos selecciona la dirección del candidato de la base de datos.

Si no están disponibles direcciones recientemente liberadas, la dirección MAC se selecciona aleatoriamente del rango de 256K de direcciones guardadas con este propósito. La dirección MAC se selecciona aleatoriamente para minimizar la posibilidad de una dirección MAC duplicada seleccionada como candidata.

Después, el Administrador de Dominios lógicos comprueba la dirección seleccionada para evitar la asignación de direcciones MAC duplicadas. El algoritmo empleado se describe en [“Detección de duplicación de direcciones MAC” en la página 122](#). Si la dirección ya está asignada, el Administrador de Dominios lógicos repite este proceso, eligiendo otra dirección y vuelve a comprobar si se producen colisiones. Este proceso continua hasta que se encuentra una dirección MAC que aun no está asignada, o se supera un intervalo de tiempo de 30 segundos. Si se alcanza el límite de tiempo, falla la creación del dispositivo y se muestra un mensaje de error parecido al siguiente.

```
Automatic MAC allocation failed. Please set the vnet MAC address manually.
```

Detección de duplicación de direcciones MAC

Para evitar la asignación de una misma dirección MAC a diferentes dispositivos, un Administrador de Dominios lógicos comprueba con otros Administrador de Dominios lógicos en otros sistemas enviando un mensaje de multidifusión por la interfaz de red predeterminada del dominio de control, incluyendo la dirección que el Administrador de Dominios lógicos desea asignar al dispositivo. El Administrador de Dominios lógicos que está intentando asignar la dirección MAC espera durante un segundo la respuesta. Si se ha asignado esa dirección MAC a un dispositivo diferente en otro sistema habilitado por Dominios lógicos, el Administrador de Dominios lógicos en dicho sistema envía una respuesta que contiene la dirección MAC en cuestión. Si la solicitud del Administrador de Dominios lógicos recibe una respuesta, éste sabe que la dirección MAC elegida ya ha sido asignada, elige otra y repite el proceso.

De manera predeterminada, los mensajes de multidifusión se envían sólo a otros administrados en la misma subred; la duración predeterminada es (TTL) 1. La TTL puede configurarse usando la propiedad del Service Management Facilities (SMF) `ldmd/hops`.

Cada Administrador de Dominios lógicos es responsable de:

- Escucha de mensajes multidifusión
- Seguimiento de las direcciones MAC asignadas a los dominios
- Búsqueda de duplicados
- Respuesta para que no se produzcan duplicados

Si el Administrador de Dominios lógicos de un sistema se apaga por cualquier motivo, se pueden producir direcciones MAC duplicadas mientras el Administrador de Dominios lógicos está apagado.

La asignación MAC automática se realiza en el momento que el dominio lógico o el dispositivo de red se crean y persisten hasta que se eliminan el dispositivo o el dominio lógico.

Nota – Se realiza una comprobación de detección para direcciones MAC duplicadas cuando el dominio lógico o el dispositivo de red se crea, y se pone en marcha el dominio lógico.

Direcciones MAC liberadas

Cuando un dominio lógico o un dispositivo asociado con una dirección MAC automática se eliminan, esa dirección MAC se guarda en una base de datos de direcciones MAC recientemente liberadas para usarla posteriormente en ese sistema. Esas direcciones MAC se guardan para evitar que se gasten las direcciones de protocolo de internet (IP) de un servidor de protocolo de configuración dinámica de host (DHCP). Cuando los servidores DHCP asignan direcciones IP, lo hacen durante un periodo de tiempo (el tiempo de asignación). La duración de la asignación a menudo se configura para que sea bastante larga, normalmente horas o días. Si se crean y eliminan dispositivos de red a una velocidad alta sin que el Administrador de

Dominios lógicos reutilice automáticamente las direcciones MAC asignadas, el número de direcciones MAC asignadas podría superar rápidamente un servidor DHCP de configuración típica.

Cuando se solicita al Administrador de Dominios lógicos que obtenga automáticamente una dirección MAC para un dominio lógico o un dispositivo de red, primero comprueba la base de datos de las direcciones MAC libres para ver si puede reutilizar una dirección MAC previamente asignada. Si en la base de datos hay una dirección MAC disponible, se ejecuta el algoritmo de detección de direcciones MAC duplicadas. Si la dirección MAC no ha sido asignada a nadie más desde que ha sido liberada, será reutilizada y eliminada de la base de datos. Si se detecta una colisión, la dirección simplemente se elimina de la base de datos. Entonces el Administrador de Dominios lógicos o bien intenta la siguiente dirección en la base de datos o si ninguna está disponible, o bien selecciona aleatoriamente una nueva dirección MAC.

Uso de adaptadores de red con Dominios lógicos

En un entorno de dominios lógicos, el servicio de conmutador virtual en ejecución en un dominio de servicio puede interactuar directamente con adaptadores de red compatibles con GLDv3. A pesar de que pueden usar adaptadores de red no compatibles con GLDv3 en estos sistemas, el conmutador virtual no puede conectarse directamente con ellos mediante interfaz. Véase [“Configuración del conmutador virtual y dominio de servicio para NAT y enrutamiento” en la página 124](#) para más información sobre cómo usar adaptadores de red no compatibles con GLDv3.

Para más información sobre cómo usar las agregaciones de enlaces, véase [“Uso de la agregación de enlaces con un conmutador virtual” en la página 139](#).

▼ Determine si el adaptador de una red es compatible con GLDv3

- 1 Use el comando del SO de Solaris de Oracle `dladm(1M)`, donde, por ejemplo, `bge0` es el nombre del dispositivo de red.

```
# dladm show-link bge0
bge0          type: non-vlan  mtu: 1500      device: bge0
```

- 2 Mire la opción `type`: en la salida:

- Los controladores compatibles con GLDv3 tendrán un tipo de `non-vlan` o `vlan`.
- Los controladores no compatibles con GLDv3 tendrá un tipo de `legacy`.

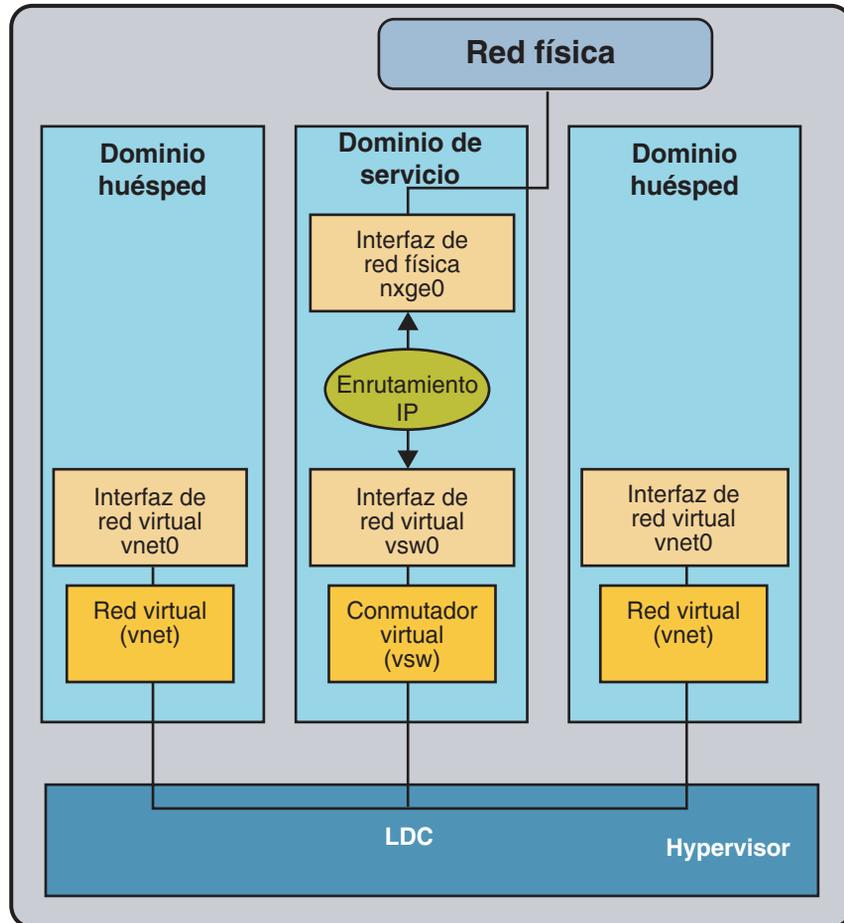
Configuración del conmutador virtual y dominio de servicio para NAT y enrutamiento

El conmutador virtual (vsw) es un conmutador de 2 capas que también pueden usarse como dispositivo de red en el dominio de servicio. El conmutador virtual puede configurarse para actuar sólo como conmutador entre los dispositivos de la red virtual (vnet) en los diferentes dominios lógicos pero sin conectividad a una red fuera del cuadro a través de un dispositivo físico. De este modo, la instalación de vsw como un dispositivo de red y la habilitación del enrutamiento de IP en el dominio de servicio habilita la comunicación entre redes virtuales fuera del cuadro, usando el dominio de servicio como enrutador. Este modo de operación es esencial para ofrecer conectividad exterior a dominios cuando el adaptador de red físico no es compatible con GLDv3.

Las ventajas de esta configuración son:

- El conmutador virtual no debe usar un dispositivo físico directamente y puede ofrecer conectividad exterior incluso cuando el dispositivo subyacente no es compatible con GLDv3.
- La configuración puede aprovechar el enrutamiento de IP y las capacidades de filtrado del SO de Solaris de Oracle.

FIGURA 8-2 Enrutamiento de red virtual



▼ Configuración del conmutador virtual para ofrecer conectividad externa a los dominios

- 1 Cree un conmutador virtual sin dispositivo físico asociado.

Si está asignando una dirección, asegúrese de que el conmutador virtual tiene una dirección MAC única.

```
primary# ldm add-vsw [mac-addr=xx:xx:xx:xx:xx:xx] primary-vsw0 primary
```

- 2 Instala el conmutador virtual como dispositivo de red además del dispositivo de red físico que está siendo usado por el dominio.**

Véase [“Configuración del conmutador virtual como interfaz primaria.”](#) en la página 53 para más información sobre cómo instalar el conmutador virtual.

- 3 Configure el dispositivo de conmutador virtual para DHCP, si es necesario.**

Véase [“Configuración del conmutador virtual como interfaz primaria.”](#) en la página 53 para más información sobre cómo configurar el conmutador virtual para DHCP.

- 4 Cree el archivo `/etc/dhcp.vsw`, si es necesario.**

- 5 Configure el enrutamiento de IP en el dominio de servicio, y configure las tablas de enrutamiento necesarias en todos los dominios.**

Para más información sobre cómo realizar esta operación, consulte [“Reenvío de paquetes y rutas en redes IPv4”](#) de *Guía de administración del sistema: servicios IP*.

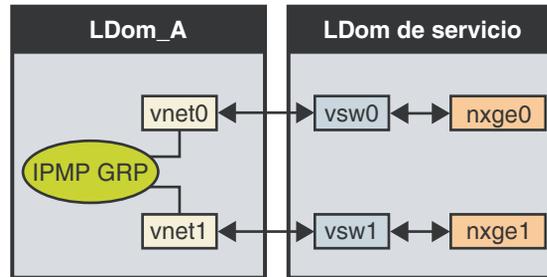
Configuración de IPMP en un entorno Dominios lógicos

La versión 1.3 de Dominios lógicos introdujo la asistencia técnica para IPMP basada en enlace con dispositivos de redes virtuales. Cuando se configura un grupo IPMP con dispositivos de red virtual, configure el grupo para que use detección basada en enlaces. Si usa versiones anteriores del programa de Oracle VM Server for SPARC (Dominios lógicos), sólo puede configurar la detección basada en sondas con dispositivos de red virtual.

Configuración de dispositivos de red virtual en un grupo IPMP en un dominio

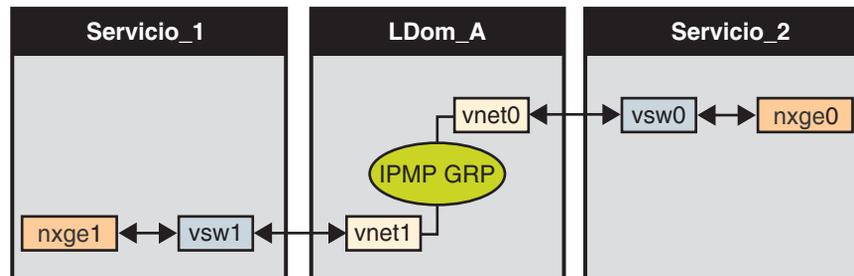
El siguiente diagrama muestra dos redes virtuales (vnet1 y vnet2) conectadas a instancias de conmutador virtual separadas (vsw0 y vsw1) en el dominio de servicio, que, a su vez, usando dos interfaces físicas diferentes (nxge0 y nxge1). En caso de fallo de enlace físico en el dominio de servicio, el dispositivo de conmutador virtual que está enlazado a ese dispositivo físico detecta el fallo del enlace. Entonces, el dispositivo de conmutador virtual propaga el fallo al dispositivo de red virtual correspondiente que está enlazado con este conmutador virtual. El dispositivo de red virtual envía una notificación de este evento en el enlace a la capa IP en el huésped LDom_A, que provoca una conmutación por error en el otro dispositivo de red virtual en el grupo IPMP.

FIGURA 8-3 Dos redes virtuales conectadas a instancias de conmutador virtual separadas



Puede conseguirse una mayor fiabilidad en el dominio lógico conectando cada dispositivo de red virtual (vnet0 y vnet1) a instancias de conmutador virtual en diferentes dominios de servicio (como se muestra en el siguiente diagrama). En este caso, además del fallo de la red física, LDom_A puede detectar un fallo en la red virtual y accionar una conmutación por error después de un bloqueo o un paro del dominio de servicio.

FIGURA 8-4 Cada dispositivo de red virtual conectado a diferentes dominios de servicio

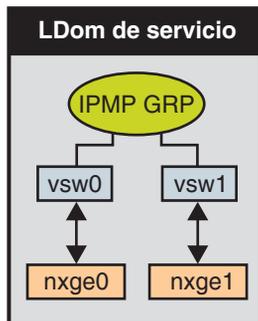


Consulte el Solaris de Oracle 10 *Guía de administración del sistema: servicios IP* para más información sobre cómo configurar y usar los grupos IPMP.

Configuración y uso de IPMP en el dominio de servicio

La IPMP puede configurarse en el dominio de servicio configurando las interfaces del conmutador virtual en un grupo. El siguiente diagrama muestra dos instancias de conmutador virtual (vsw0 y vsw1) que están enlazadas a dos dispositivos físicos diferentes. Las dos interfaces de conmutador virtual puede instalarse y configurarse en un grupo IPMP. En caso de fallo de enlace físico, el dispositivo de conmutador virtual que está enlazado con ese dispositivo físico detecta el fallo del enlace. Entonces, el dispositivo de conmutador virtual envía una notificación de este evento de enlace a la capa de IP en el dominio de servicio, que provoca una conmutación por error en el otro dispositivo de conmutador virtual en el grupo IPMP.

FIGURA 8-5 Dos interfaces de conmutador virtual configuradas como parte de un grupo IPMP



Uso de IPMP basado en enlace en funciones de redes virtuales de Dominios lógicos

Desde la versión 1.3 de Dominios lógicos, los dispositivos de red virtual y de conmutador virtual admiten actualizaciones del estado del enlace en la pila de red. De manera predeterminada, un dispositivo de red virtual informa del estado del enlace virtual (el LDC al conmutador virtual). Esta configuración está habilitada de manera predeterminada y no es necesario que realice ningún paso de configuración adicional.

A veces puede ser necesario detectar cambios de estado del enlace de red física. Por ejemplo, si se ha asignado un conmutador virtual a un dispositivo físico, incluso si el enlace del dispositivo de red virtual está funcionando, el enlace de la red física desde el dominio de servicio a la red externa puede estar apagado. En este caso, puede ser necesario obtener e informar del estado de enlace físico al dispositivo de red virtual y la pila.

La opción `linkprop=phys-state` puede usarse para configurar el estado del enlace físico para los dispositivos de red virtual como dispositivos de redes virtuales. Cuando esta opción está habilitada, el dispositivo virtual (red virtual o conmutador virtual) indica el estado de enlace basado en el estado del enlace físico mientras está instalado como una interfaz en el dominio. Puede usar los comandos de administración de red estándar de Solaris de Oracle como `dladm` y `ifconfig` para comprobar el estado del enlace. Véanse las páginas del comando `man dladm(1M)` y `ifconfig(1M)`. Además, el estado del enlace también se registra en el archivo `/var/adm/messages`.

Nota – Puede ejecutar ambos controladores, el de `link-state-unaware` y `link-state-aware` `vnet` y `vsw` de manera simultánea en un sistema Dominios lógicos. En cualquier caso, si intenta configurar un IPMP basado en enlace, debe instalar el controlador preparado para el estado de enlace. Si desea habilitar las actualizaciones de estado de enlace físico, actualice ambos controladores `vnet` y `vsw` al SO 10 9/10 Solaris de Oracle y ejecute como mínimo la versión 1.3 del Administrador de Dominios lógicos.

▼ Configurar las actualizaciones del estado del enlace físico

Este procedimiento muestra como habilitar las actualizaciones de estado del enlace físico para los dispositivos de redes virtuales.

Puede habilitar las actualizaciones del estado del enlace físico para el dispositivo de conmutador virtual siguiendo unos pasos parecidos y especificando la opción `linkprop=phys-state` en los comandos `ldm add-vsw` y `ldm set-vsw`.

Nota – Necesitará usar la opción `linkprop=phys-state` sólo si el dispositivo de conmutador virtual mismo está instalado como una interfaz. Si se especifica `linkprop=phys-state` y el enlace físico está apagado, el dispositivo de red virtual indica su estado como apagado, incluso si la conexión del conmutador virtual está encendida. Esta situación se produce porque el SO de Solaris de Oracle actualmente no ofrece interfaces para indicar dos estados de enlace distintos, como estado de enlace virtual y estado de enlace físico.

1 Conviértase en un superusuario o asuma una función equivalente.

Las funciones contienen autorizaciones y comandos con privilegios. Para más información sobre las funciones, véase “[Configuring RBAC \(Task Map\)](#)” de *System Administration Guide: Security Services*.

2 Habilite las actualizaciones de estado de enlace físico para el dispositivo virtual.

Puede habilitar las actualizaciones del estado de enlace físico para un dispositivo de red virtual de las siguientes maneras:

- Cree un dispositivo de red virtual especificando `linkprop=phys-state` cuando se ejecuta el comando `ldm add-vnet`.
Si se especifica la opción `linkprop=phys-state` se configura el dispositivo de red virtual para que obtenga las actualizaciones del estado de enlace físico y las agregue al informe de la pila.

Nota – Si se especifica `linkprop=phys-state` y el enlace físico está apagado (incluso si la conexión al conmutador virtual está encendida), el dispositivo de red virtual indica como estado de enlace apagado. Esta situación se produce porque el SO de Solaris de Oracle actualmente no ofrece interfaces para indicar dos estados de enlace distintos, como estado de enlace virtual y estado de enlace físico.

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

El siguiente ejemplo habilita las actualizaciones de enlace físico para `vnet0` conectado a `primary-vsw0` en el dominio lógico `ldom1`:

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

- Modifique un dispositivo de red virtual especificando `linkprop=phys-state` cuando se ejecuta el comando `ldm set-vnet`.

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

El siguiente ejemplo habilita las actualizaciones de estado de enlace físico para `vnet0` en el dominio lógico `ldom1`:

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

Para deshabilitar las actualizaciones de estado de enlace físico, especifique `linkprop=` ejecutando el comando `ldm set-vnet`.

El siguiente ejemplo deshabilita las actualizaciones de estado de enlace físico para `vnet0` en el dominio lógico `ldom1`:

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

Ejemplo 8-1 Configuración de IPMP basado en enlaces

El siguiente ejemplo muestra cómo configurar un IPMP basado en enlace con y sin habilitar las actualizaciones del estado del enlace físico:

- El siguiente ejemplo configura dos dispositivos de redes virtuales en un dominio. Cada dispositivo de red virtual está conectado a un dispositivo de conmutador virtual separado en el dominio de servicio para usar el IPMP basado en enlace.

Nota – Pruebe las direcciones que no están configuradas en los dispositivos de redes virtuales. Asimismo, no es necesario realizar una configuración adicional cuando usa el comando `ldm add-vnet` para crear estos dispositivos de red virtual.

Los siguientes comandos agregan dispositivos de redes virtuales al dominio. Tenga en cuenta que dado que no se especifica `linkprop=phys-state`, sólo se efectúa un seguimiento de los cambios de estado del enlace del conmutador virtual.

```
# ldm add-vnet vnet0 primary-vsw0 ldom1
# ldm add-vnet vnet1 primary-vsw1 ldom1
```

Los siguientes comandos configuran los dispositivos de red virtual en el dominio huésped y les asigna a un grupo IPMP. Tenga en cuenta que las direcciones de prueba no están configuradas en estos dispositivos de red virtual porque se está usando la detección de fallos basados en enlace.

```
# 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
```

- El siguiente ejemplo configura dos dispositivos de redes virtuales en un dominio. Cada dominio está conectado a un dispositivo de conmutador virtual separado en el dominio de servicio para usar el IPMP basado en enlace. Los dispositivos de red virtual también se configuran para obtener actualizaciones de estado de enlace físico.

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

Nota – El conmutador virtual debe tener un dispositivo de red física asignado para enlazar correctamente. Si el dominio ya está enlazado y el conmutador virtual no tiene un dispositivo de red física asignado, el comando `ldm add-vnet` fallará.

Los siguientes comandos instalan los dispositivos de red virtual y los asignan a un grupo IPMP:

```
# 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
```

Configuración y uso de IPMP en versiones anteriores a Dominios lógicos 1.3

En las versiones de Dominios lógicos anteriores a la 1.3, el conmutador virtual y los dispositivos de red virtual no podían realizar la detección de fallos de enlace. En estas versiones, la detección de fallos de red y recuperación puede configurarse usando un IPMP basado en sonda.

Configuración de IPMP en el dominio huésped

Los dispositivos de red virtual en el dominio huésped puede configurarse en un grupo IPMP tal y como se muestra en la [Figura 8–3](#) y en la [Figura 8–4](#). La única diferente es que la detección de

fallos basada en sonda se usa configurando direcciones de prueba en los dispositivos de red virtual. Véase *Guía de administración del sistema: servicios IP* para más información sobre la configuración de los IPMP basados en sondas.

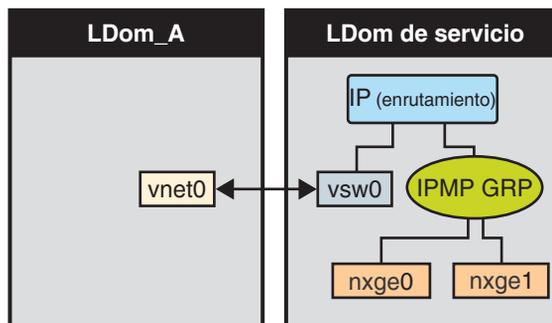
Configuración de IPMP en el dominio de servicio

En versiones de Dominios lógicos anteriores a la 1.3, el dispositivo de conmutador virtual no podía detectar el fallo del enlace físico. En estos casos, la detección y recuperación de fallos de red puede configurarse fijando las opciones de las interfaces físicas en el dominio de servicio en un grupo IPMP. Para ello, configure el conmutador virtual en el dominio de servicio sin asignar un dispositivo de red física a él. Esto es, no especifique un valor para la propiedad `net-dev` (`net-dev=`) mientras usa el comando `ldm add-vswitch` para crear el conmutador virtual. Instale la interfaz de conmutador virtual en el dominio de servicio y configure el dominio de servicio para actuar como enrutador de IP. Consulte el Solaris de Oracle 10 *Guía de administración del sistema: servicios IP* para más información sobre la configuración del enrutamiento IP.

Una vez configurado, el conmutador virtual envía todos los paquetes originales de las redes virtuales (y con destino a un equipo externo) a su capa de IP, en vez de mandar los paquetes directamente por el dispositivo físico. En caso de un fallo de interfaz física, la capa IP detecta el fallo y reconfigura la ruta automáticamente de los paquetes a través de la interfaz secundaria.

Dado que las interfaces físicas son configuradas directamente en un grupo IPMP, el grupo puede configurarse para una detección basada en enlace o en sonda. El siguiente diagrama muestra dos interfaces de red (`nxge0` y `nxge1`) configuradas como parte de un grupo IPMP. La instancia de conmutador virtual (`vsw0`) se ha instalado como dispositivo de red para enviar paquetes a la capa IP.

FIGURA 8-6 Dos interfaces de red configuradas como parte de un grupo IPMP



▼ Configuración de la ruta del host para un IPMP basado en sonda

Nota – Este procedimiento sólo es aplicable a los dominios huésped anteriores a la 1.3, en los que sólo se admitía un IPMP basado en sonda.

Si no se ha configurado una ruta explícita en la red correspondiente a las interfaces IPMP, entonces una o varias rutas de host explícitas a los equipos de destino deben configurarse para que la detección basada en sondas de IPMP funcione como se espera. En caso contrario, la detección de sonda puede no detectar los fallos de la red.

- **Configure una ruta de host.**

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

Por ejemplo:

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

Consulte “[Configuración de sistemas de destino](#)” de *Guía de administración del sistema: servicios IP* para más información.

Uso de etiquetado VLAN

Desde la versión 10 10/08 del SO de Solaris de Oracle y el software 1.1 del Dominios lógicos, se admite el etiquetado 802.1Q VLAN en la infraestructura de red de Dominios lógicos.

Nota – Las VLANs etiquetadas no se admiten en las versiones anteriores de Dominios lógicos para los componentes de las funciones de red.

Los dispositivos del conmutador virtual (vsw) y la red virtual (vnet) admiten el intercambio de paquetes Ethernet basados en el identificador (ID) de la red de área local virtual (VLAN) y pueden administrar el etiquetado o desetiquetado necesario de los marcos de Ethernet.

Puede crear múltiples interfaces VLAN en un dispositivo vnet en un dominio huésped. Puede usar el comando del SO de Solaris de Oracle `ifconfig(1M)` para crear una interfaz VLAN sobre un dispositivo de red virtual, la misma manera se usa para configurar una interfaz VLAN sobre cualquier otro dispositivo de red físico. Los requisitos adicionales en los entornos Dominios lógicos es que debe asignar el vnet a las VLAN correspondientes usando los comandos de CLI del Administrador de Dominios lógicos. Consulte `ldm(1M)` para una información completa sobre los comandos de la CLI del Administrador de Dominios lógicos.

De manera parecida, puede configurar las interfaces de VLAN sobre un dispositivo de conmutador virtual en el dominio de servicio. Los id de VLAN IDs de 2 a 4094 son válidos; el id 1 de VLAN está reservado como `default-vlan-id`.

Cuando crea un dispositivo `vnet` en un dominio huésped, debe asignarlo a las VLANs necesarias especificando un `id` de VLAN de puerto y un `ID` de VLAN de cero o más para este `vnet`, usando los argumentos `pvid=` y `vid=` en el comando `ldm add-vnet`. Esto configura el conmutador virtual para admitir múltiples VLAN en la red Dominios lógicos e intercambia paquetes usando direcciones MAC y ID de VLAN en la red.

De manera parecida, cualquier VLAN a la que debe pertenecer un dispositivo `vsw`, cuando se instala como una interfaz de red, debe estar configurado en el dispositivo `vsw` usando los argumentos `pvid=` y `vid=` en el comando `ldm add-vsw`.

Puede cambiar el VLAN al que pertenece un dispositivo usando el comando `ldm set -vnet` o `ldm set -vsw`.

Puerto VLAN ID (PVID)

El PVID indica la VLAN de la cual el dispositivo de red virtual necesita ser miembro, en modo sin etiquetas. En este caso, el dispositivo `vsw` ofrece el etiquetado o desetiquetado de marcos necesarios para el dispositivo `vnet` en la VLAN especificada por el PVID. Cualquier marco saliente de la red virtual que están desetiquetados se etiquetan con su PVID por el conmutador virtual. Los marcos entrantes etiquetados con este PVID son desetiquetados por el conmutador virtual, antes de enviarlos al dispositivo `vnet`. Por lo tanto, la asignación de un PVID a un `vnet` implícitamente significa que el puerto de la red correspondiente en el conmutador virtual se marca como desetiquetado para la VLAN especificada por el PVID. Sólo puede tener un PVID para un dispositivo `vnet`.

La interfaz de red virtual correspondiente, cuando se configura usando el comando `ifconfig(1M)` sin un `id` de VLAN y usando sólo la instancia de dispositivo, hace que la interfaz sea implícitamente asignada a la VLAN especificada por el PVID de la red virtual.

Por ejemplo, si debe instalar la instancia `vnet 0`, usando el siguiente comando y el argumento `pvid=` para `vnet` se ha especificado como `10`, la interfaz `vnet0` se asignaría implícitamente para que pertenezca a VLAN 10.

```
# ifconfig vnet0 plumb
```

VLAN ID (VID)

El VID indica la VLAN a la que un dispositivo de red virtual o conmutador virtual debe ser miembro, en modo con etiquetas. El dispositivo de red virtual envía y recibe marcos etiquetados sobre las VLAN especificadas por la VID. El conmutador virtual pasa cualquier marco que está etiquetado con el VID específico entre el dispositivo de red virtual y la red externa.

▼ Asignación de una VLAN a un conmutador virtual y dispositivo de red virtual

1 Asigna el conmutador virtual (vsw) a dos VLAN.

Por ejemplo, configura VLAN 21 como no etiquetada y VLAN 20 como etiquetada. Asigna la red virtual (vnet) a tres VLANs. Configure VLAN 20 como no etiquetada y VLAN 21 y 22 como etiquetadas.

```
# 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 Instalación de interfaces VLAN.

Este ejemplo asume que el número de la instancia de estos dispositivos es 0 en los dominios y las VLAN están asignadas a estas subredes:

VLAN	Subred
20	192.168.1.0 (máscara de red: 255.255.255.0)
21	192.168.2.0 (máscara de red: 255.255.255.0)
22	192.168.3.0 (máscara de red: 255.255.255.0)

a. Instale la interfaz de VLAN en el dominio de servicio (primary).

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

b. Instale la interfaz de VLAN en el dominio huésped (ldom1).

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

Para más información sobre como configurar las interfaces de VLAN en el SO de Solaris de Oracle, consulte “[Administración de redes de área local virtuales](#)” de *Guía de administración del sistema: servicios IP*.

▼ Instalación de un dominio huésped cuando el servidor de instalación es una VLAN

Tenga cuidado cuando instala un dominio huésped sobre red (JumpStart) y el servidor de instalación está en una VLAN. Especifique que el id de VLAN que está asociado con el servidor

de instalación como el PVID del dispositivo de red virtual y no configure VLAN etiquetadas (vid) para ese dispositivo de red virtual. Debe hacer esto porque OBP no está preparado para VLAN y no puede administrar paquetes de red etiquetados con VLAN. El conmutador virtual maneja el etiquetado y desetiquetado de los paquetes hacia y desde el dominio huésped durante la instalación de la red. Después de haber completado la instalación y el inicio del SO de Solaris de Oracle, puede configurar el dispositivo de red virtual para que esté etiquetado en esa VLAN. Entonces puede agregar un dispositivo de red virtual a las VLAN adicionales en modo con etiquetas.

Para más información sobre el uso de JumpStart para instalar un dominio huésped, véase [“Realización de una operación JumpStart en un dominio huésped” en la página 61.](#)

1 Inicialmente configure el dispositivo de red en modo sin etiquetas.

Por ejemplo, si el servidor de instalación está en VLAN 21, configure la red virtual inicialmente de la siguiente manera:

```
primary# ldm add-vnet pvid=21 vnet01 primary-vsw0 ldom1
```

2 Después de haber completado la instalación y efectuado el inicio del SO de Solaris de Oracle, configure la red virtual en modo con etiquetas.

```
primary# ldm set-vnet pvid= vid=21, 22, 23 vnet01 primary-vsw0 ldom1
```

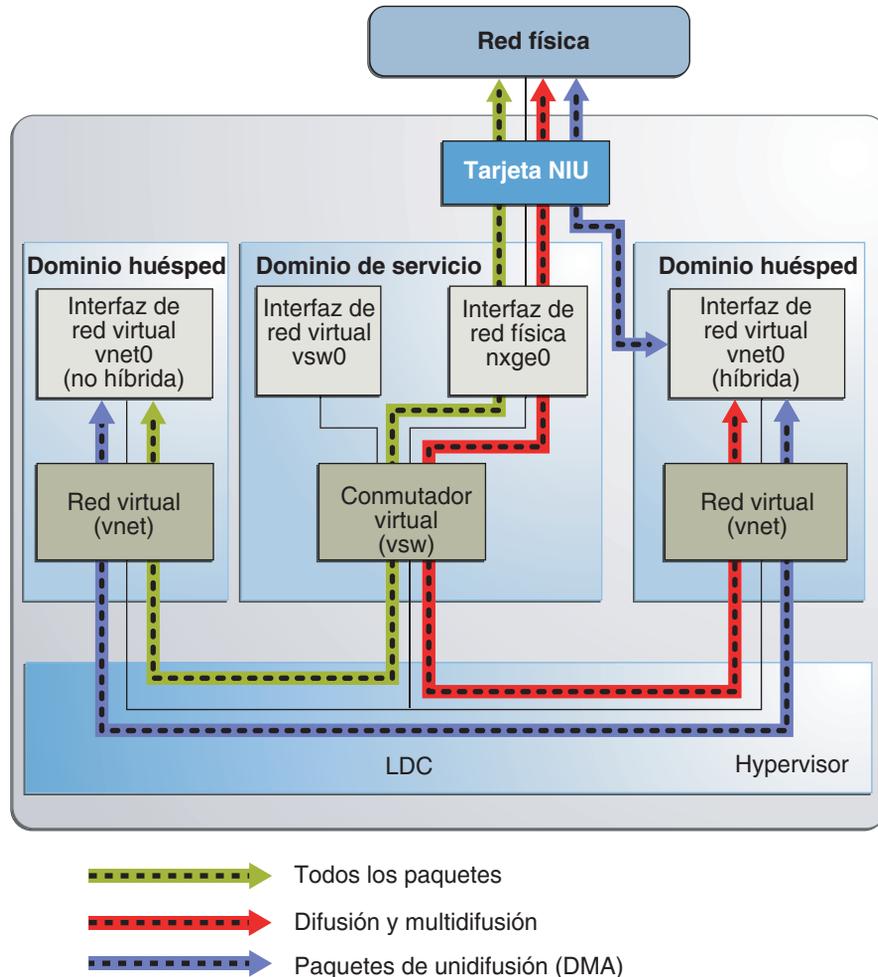
Uso de E/S híbridas de NIU

La estructura de E/S virtual implementa un modelo de E/S *híbrida* que mejora la funcionalidad y el rendimiento. El modelo de E/S híbrida combina la E/S virtualizada para permitir una implementación flexible de los recursos de E/S en los equipos virtuales. Es especialmente útil cuando la E/S directa no ofrece una capacidad total para el equipo virtual, o la E/S directa no está permanente o consistentemente disponible para el equipo virtual. Esto puede ser debido a la disponibilidad de recursos o migración del equipo virtual. La arquitectura de E/S híbrida es conveniente para la unidad de interfaz de red (NIU) en las plataformas UltraSPARC T2 y SPARC T3 de Sun. Una NIU es una interfaz de E/S de red que está integrada en chip. Esta arquitectura le permite la asignación dinámica de los recursos de acceso directo a memoria (DMA) para los dispositivos de las funciones de red virtual y por lo tanto, le ofrece un rendimiento coherente con las aplicaciones en el dominio.

La E/S híbrida de NIU está disponible para las plataformas UltraSPARC T2 y SPARC T3 de Sun. Esta característica está habilitada por un modo híbrido opcional para un dispositivo de red virtual (vnet) donde los recursos de hardware de DMA son prestados a un dispositivo vnet en un dominio huésped para mejorar el rendimiento. En el modo híbrido, un dispositivo vnet en un dominio huésped puede enviar y recibir tráfico de unidifusión desde una red externa directamente en el dominio huésped usando los recursos de hardware DMA. El tráfico de difusión o multidifusión y el tráfico de unidifusión a los otros dominios huésped en el mismo sistema continua siendo enviado usando el mecanismo de comunicación de E/S virtual.

Nota – La E/S híbrida de NIU no está disponible en las plataformas UltraSPARC T2.

FIGURA 8-7 Funciones de redes virtuales híbridas



El modo híbrido se aplica sólo para los dispositivos vnet que están asociados con un conmutador virtual (vsw) configurado para usar un dispositivo de red NIU. Dado que los recursos hardware DMA que se pueden compartir son limitados, sólo tres dispositivos vnet por vsw pueden tener recursos hardware DMA asignados en un determinado momento. Si más de tres dispositivos vnet tienen el modo híbrido habilitado, la asignación se realiza en orden de

llegada. Dado que hay dos dispositivos de red NIU en un sistema puede haber un total de seis dispositivos vnet en dos conmutadores virtuales diferente con los recursos hardware DMA asignados.

A continuación se incluyen los puntos que es necesario tener en cuenta cuando se usa esta característica:

- La opción de modo híbrido para un dispositivo vnet se trata sólo como una sugerencia. Esto significa que los recursos DMA se asignan sólo cuando están disponibles y el dispositivo puede usarlos.
- Los comandos de CLI del Administrador de Dominios lógicos no validan la opción del modo híbrido, esto es, es posible configurar el modo híbrido sólo en cualquier vnet o cualquier número de dispositivos vnet.
- Los dominios huésped y el dominio de servicio debe ejecutar como mínimo el SO 10 10/08 de Solaris de Oracle.
- Hasta un máximo de sólo tres dispositivos vnet por vsw pueden tener los recursos hardware DMA prestados en un determinado momento. Dado que hay dos dispositivos de red NIU, puede haber un total de seis dispositivos vnet con los recursos de hardware DMA prestados.

Nota – Configure el modo híbrido sólo para tres dispositivos vnet por vsw de manera que se garantice que tienen los recursos de hardware DMA asignados.

- De manera predeterminada el modo híbrido está deshabilitado para un dispositivo vnet. Tiene que ser habilitado explícitamente con los comandos de CLI del Administrador de Dominios lógicos. Véase [“Habilitación del modo híbrido” en la página 139](#). (Consulte la página de comandos `ldm(1M)` para más detalles).
- La opción de modo híbrido no puede cambiarse dinámicamente mientras el dominio huésped está activo.
- Los recursos de hardware DMA se asignan sólo cuando un dispositivo vnet está activo y está instalado en el dominio huésped.
- El controlador Ethernet de 10-gigabyte de NIU (nxge) se usa para la tarjeta NIU. También se encuentra el mismo controlador para otras tarjetas de red de 10-gigabyte. En cualquier caso, la característica de E/S híbrida de NIU está disponible sólo para los dispositivos de red NIU.

▼ Configuración de un conmutador virtual con un dispositivo de red NIU

1 Determine un dispositivo de red NIU.

El siguiente ejemplo muestra la salida en un servidor UltraSPARC T2:

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

El siguiente ejemplo muestra la salida en un servidor SPARC T3-1:

```
# grep nxge /etc/path_to_inst
"/niu@480/network@0" 0 "nxge"
"/niu@480/network@1" 1 "nxge"
```

2 Configure un conmutador virtual.

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

▼ Habilitación del modo híbrido

- Por ejemplo, habilite un modo híbrido para un dispositivo vnet mientras está siendo creado.

```
# ldm add-vnet mode=hybrid vnet01 primary-vsw0 ldom01
```

▼ Deshabilitación del modo híbrido

- Por ejemplo, deshabilite el modo híbrido para un dispositivo vnet.

```
# ldm set-vnet mode= vnet01 ldom01
```

Uso de la agregación de enlaces con un conmutador virtual

Desde la versión SO 10 10/08 de Solaris de Oracle y el software 1.1 del Dominios lógicos, el conmutador virtual puede configurarse para usar una agregación de enlace. Una agregación de enlace se usa como el dispositivo de red del conmutador virtual para conectar a la red física. Esta configuración permite que el conmutador virtual aprovechar las características ofrecidas por el estándar de agregación de enlace IEEE 802.3ad. Estas características incluyen ancho de banda aumentado, equilibrio de carga y conmutación por error. Para más información sobre como configurar la agregación de enlaces, véase la *Guía de administración del sistema: servicios IP*.

Después de haber creado una agregación de enlace, puede asignarla al conmutador virtual. La realización de esta asignación es parecida a la asignación de un dispositivo de red física a un conmutador virtual. Use el comando `ldm add-vswitch` o `ldm set-vswitch` para fijar la propiedad `net - dev`.

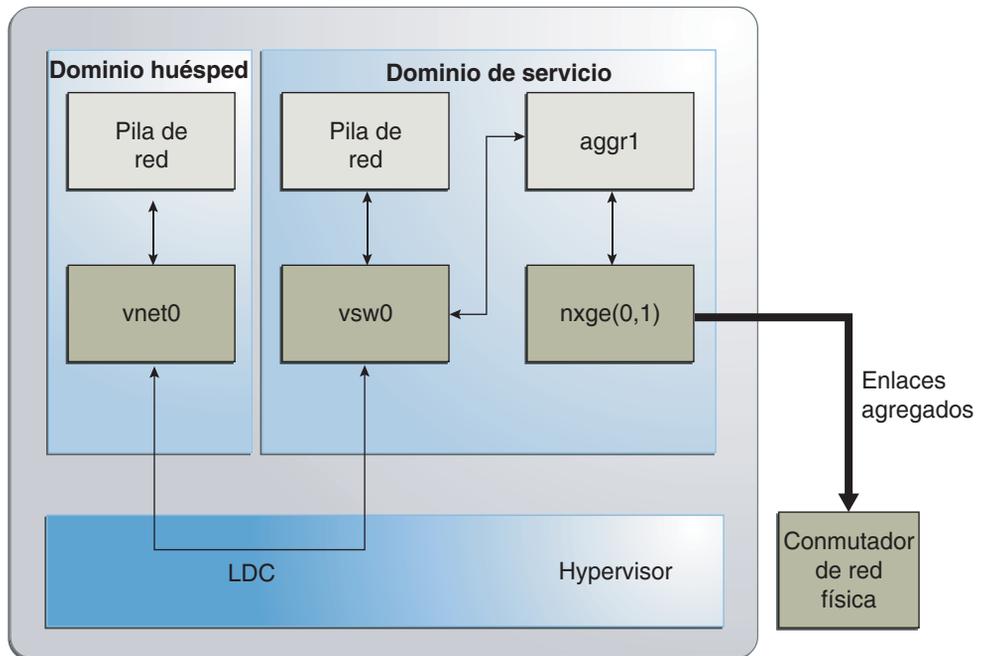
Cuando la agregación de enlace se asigna al conmutador virtual, el tráfico de y hacia la red física pasa por la agregación. Cualquier equilibrio de carga o conmutación por error necesarios se manejan de manera transparente por la estructura de agregación subyacente. La agregación de enlaces es completamente transparente a los dispositivos de red virtual (vnet) que están en los dominios huésped y que están enlazados a un conmutador virtual que usa una agregación.

Nota – No puede agrupar dispositivos de redes virtuales (vnet y vsw) en una agregación de enlace.

Puede instalar y usar el conmutador virtual que está configurado para usar la agregación de enlace en el dominio de servicio. Véase [“Configuración del conmutador virtual como interfaz primaria.” en la página 53.](#)

La siguiente figura ilustra un conmutador virtual configurado para usar una agregación, aggr1, sobre interfaces físicas nxge0 y nxge1.

FIGURA 8-8 Configuración de un conmutador virtual para usar una agregación de enlace



Configuración de marcos Jumbo

El conmutador virtual del Dominios lógicos (vsw) y los dispositivos de la red virtual (vnet) ahora pueden admitir marcos Ethernet con un tamaño de datos útiles superior a 1500 bytes. Este cambio hace que los controladores puedan aumentar la capacidad de tratamiento de la red.

▼ Configuración de la red virtual y de los dispositivos de conmutador virtual para el uso de marcos Jumbo

Puede habilitar los marcos jumbo especificando la unidad de transmisión máxima (MTU) para el dispositivo de conmutador virtual. En estos casos, el dispositivo de conmutador virtual y todos los dispositivos de la red virtual que están enlazados al dispositivo de conmutador virtual usan el valor de MTU especificado.

En algunas circunstancias, se puede especificar un valor de MTU directamente en un dispositivo de red virtual. Puede hacer esto si el valor de MTU necesario para el dispositivo de red virtual debe ser inferior al admitido por el conmutador virtual.

Nota – En el SO 5/09 10 de Solaris de Oracle, la MTU de un dispositivo físico debe estar configurado para coincidir con la MTU del conmutador virtual. Para más información sobre la configuración de controladores especiales, véase la página de comando `man` que corresponde a dicho controlador en la sección 7D del manual de referencia de Solaris de Oracle. Por ejemplo, para obtener información sobre el controlador `nxge`, véase la página de comando `man nxge(7D)`.

1 Inicie la sesión en el dominio de control

2 Conviértase en un superusuario o asuma una función equivalente.

Las funciones contienen autorizaciones y comandos con privilegios. Para más información sobre las funciones, véase “[Configuring RBAC \(Task Map\)](#)” de *System Administration Guide: Security Services*.

3 Determine el valor de MTU que desea usar para la red virtual.

Puede especificar un valor de MTU de 1500 a 16000 bytes. La MTU especificada debe coincidir con la MTU del dispositivo de red física que se ha asignado al conmutador virtual.

4 Especifique el valor de MTU de un dispositivo de conmutador virtual o un dispositivo de red virtual.

Elija una de estas posibilidades:

- Habilite los marcos jumbo en un dispositivo de conmutador virtual nuevo en el dominio de servicio especificando la MTU como un valor de la propiedad `mtu`.

```
# ldm add-vsw mtu=value vswitch-name ldom
```

Además de configurar el conmutador virtual, este comando actualiza el valor de MTU de cada dispositivo de red virtual que se enlazará con este conmutador virtual.

- Habilite los marcos jumbo en un dispositivo de conmutador virtual existente en el dominio de servicio especificando la MTU como un valor de la propiedad `mtu`.

```
# ldm set-vsw mtu=value vswitch-name
```

Además de configurar el conmutador virtual, este comando actualiza el valor de MTU de cada dispositivo de red virtual que se enlazará con este conmutador virtual.

En circunstancias raras, puede necesitar usar el comando `ldm add-vnet` o `ldm set-vnet` para especificar un valor de MTU para un dispositivo de red virtual que difiere del valor de MTU del conmutador virtual. Por ejemplo, puede cambiar el valor de MTU del dispositivo de red virtual si configura las VLAN en un dispositivo de red virtual y la MTU de VLAN más grandes es inferior al valor de MTU en el conmutador virtual. Un controlador `vnet` que admite marcos jumbo puede no ser necesario para dominios en los que sólo se usa el valor de MTU predeterminado. En cualquier caso, si los dominios tienen dispositivos de red virtual enlazados a un conmutador virtual que usa marcos jumbo, asegúrese de que el controlador `vnet` admite los marcos jumbo.

Si usa el comando `ldm set-vnet` para especificar un valor de `mtu` en un dispositivo de red virtual, las actualizaciones futuras del valor de MTU del dispositivo de conmutador virtual no se propagan al dispositivo de red virtual. Para volver a habilitar el dispositivo de red virtual para obtener un valor de MTU del dispositivo de conmutador virtual, ejecute el siguiente comando:

```
# ldm set-vnet mtu= vnet-name ldom
```

Tenga en cuenta que la habilitación de los marcos jumbo para un dispositivo de red virtual habilita automáticamente los marcos jumbo para cualquier recurso híbrido de E/S que está asignado a ese dispositivo de red virtual.

En el dominio de control, el Administrador de Dominios lógicos actualiza los valores de MTU que son iniciados por los comandos `ldm set-vsw` y `ldm set-vnet` como operaciones de reconfiguración retrasada. Para realizar las actualizaciones de MTU a dominios que no sean los dominios de control, debe parar un dominio antes de ejecutar el comando `ldm set-vsw` o `ldm set-vnet` para modificar el valor de MTU.

Ejemplo 8-2 Configuración de marcos Jumbo en conmutador virtual y dispositivos de red virtual

- El siguiente ejemplo muestra como agregar un nuevo dispositivo de conmutador virtual que usa un valor de MTU de 9000. Este valor de MTU se propaga desde el dispositivo de conmutador virtual a todos los dispositivos de red virtual de cliente.

Primero, el comando `ldm add-vsw` crea el dispositivo de conmutador virtual, `primary-vsw0`, con un valor de MTU de 9000. Tenga en cuenta que la instancia 0 del dispositivo de red `nxge0` se especifica como un valor de la propiedad `net-dev`.

```
# ldm add-vsw net-dev=nxge0 mtu=9000 primary-vsw0 primary
```

Después, el comando `ldm add-vnet` agrega un dispositivo de red virtual cliente a este conmutador virtual, `primary-vsw0`. Tenga en cuenta que la MTU del dispositivo de red virtual se asigna implícitamente por el conmutador virtual al que está enlazado. Como resultado, el comando `ldm add-vnet` no necesita que se especifique un valor para la propiedad `mtu`.

```
# ldm add-vnet vnet01 primary-vsw0 ldom1
```

El comando `ifconfig` instala la interfaz de conmutador virtual en el dominio de servicio, `primary`. La salida del comando `ifconfig vsw0` muestra que el valor de la propiedad `mtu` es 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 ffffffff broadcast 192.168.1.255
      ether 0:14:4f:fa:0:99
```

El comando `ifconfig` instala la interfaz de red virtual en el dominio huésped, `ldom1`. La salida del comando `ifconfig vnet0` muestra que el valor de la propiedad `mtu` es 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 ffffffff broadcast 192.168.1.255
      ether 0:14:4f:f9:c4:13
```

- El siguiente ejemplo muestra cómo usar el comando `ifconfig` para cambiar la MTU de la interfaz a 4000.

Tenga en cuenta que la MTU de un dispositivo sólo puede cambiarse a un valor que sea inferior a la MTU del dispositivo que es asignado por el Administrador de Dominios lógicos. Este método es útil cuando las VLANs están configuradas y cada interfaz de VLAN necesita una MTU diferente.

```
# 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 ffffffff broadcast 192.168.1.255
      ether 0:14:4f:f9:c4:13
```

Compatibilidad con versiones anteriores (no preparadas para Jumbo) de los controladores vnet y VSW

Los controladores que admiten los marcos jumbo puede operar con controladores que no admiten marcos jumbo en el mismo sistema. Esta interoperatividad es posible porque la asistencia técnica de marcos jumbo no está habilitada cuando se crea el conmutador virtual.

Nota – No fije la propiedad `mtu` si cualquier dominio huésped o de servicio que están asociados con el conmutador virtual no usan los controladores de Dominios lógicos que admiten los marcos jumbo.

Los marcos Jumbo pueden habilitarse cambiando la propiedad `mtu` de un conmutador virtual del valor predeterminado de 1500. En esta instancia, las versiones anterior ignoran la configuración `mtu` y siguen usando el valor predeterminado. Tenga en cuenta que la salida `ldm list` mostrará en valor de MTU que ha especificado y no el valor predeterminado. Cualquier marco superior a la MTU predeterminada no se envía a esos dispositivos y son liberados por los nuevos controladores. Esta situación puede provocar un comportamiento de la red inconstante con los huéspedes que aun usan controladores antiguos. Esto es aplicable tanto a los dominios huéspedes de cliente como a los dominios de servicio.

Así pues, mientras los marcos jumbo están habilitados, asegúrese de que todos los dispositivos virtuales en la red de Dominios lógicos están actualizados para usar los nuevos controladores que admiten marcos jumbo. Asimismo, asegúrese de que actualiza al menos hasta la versión 1.2 de Dominios lógicos para que pueda configurar los marcos jumbo.

Migración de dominios

Este capítulo describe como migrar los dominios de un equipo host a otro equipo host.

Este capítulo trata sobre los siguientes temas:

- “Introducción a la migración de dominios” en la página 145
- “Información general sobre la operación de migración” en la página 146
- “Compatibilidad de software” en la página 146
- “Autenticación para las operaciones de migración” en la página 147
- “Migración de un dominio” en la página 147
- “Migración de un dominio activo” en la página 148
- “Migración de dominios enlazados o inactivos” en la página 153
- “Realización de una simulación” en la página 147
- “Seguimiento de una migración en curso” en la página 154
- “Cancelación de una migración en curso” en la página 155
- “Recuperación de una migración fallida” en la página 155
- “Realización de migraciones no interactivas” en la página 147
- “Ejemplos de migración” en la página 156

Introducción a la migración de dominios

La migración de dominios ofrece la posibilidad de migrar un dominio lógico de un equipo host a otro. El host donde se inicia la migración se llama equipo de origen, y el host a donde migra el dominio se llama equipo de destino. De manera parecida, una vez ha comenzado una migración, el dominio que se debe migrar se llama dominio de origen y el comando de intérprete del dominio creado en el equipo de destino se llama dominio de destino mientras la migración está en curso.

Información general sobre la operación de migración

El Administrador de Dominios lógicos en el equipo de destino acepta la solicitud de migración de un dominio y establece una conexión de red segura con el Administrador de Dominios lógicos en ejecución en el equipo de destino. Una vez se ha establecido esta conexión, se produce la migración. La misma migración puede dividirse en diferentes fases.

Fase 1: Después de conectar con el Administrador de Dominios lógicos que se ejecuta en el host de destino, se transfiere la información sobre el equipo y dominio de origen al host de destino. Esta información se usa para realizar una serie de comprobaciones para determinar si una migración es posible. Las comprobaciones difieren dependiendo del estado del dominio de origen. Por ejemplo, si el dominio de origen está activo, se realizan un conjunto diferente de comprobaciones que si el dominio está enlazado o inactivo.

Fase 2: Cuando se han realizado todas las comprobaciones en la fase 1, los equipos de origen y de destino se preparan para la migración y el dominio de origen se suspende. En el equipo de destino se crea un dominio para recibir el dominio de origen.

Fase 3: Para un dominio activo, la siguiente fase es transferir toda la información sobre el estado en tiempo de ejecución para el dominio en el destino. Esta información se obtiene del hypervisor. En el destino, la información de estado se instala en el hypervisor.

Fase 4: Transferencia. Después de haber transferido toda la información de estado, la transferencia se produce cuando el dominio de destino reinicia la ejecución (si el origen estaba activo) y se destruye el dominio de origen. A partir de este momento, el dominio de destino es la única versión de dominio en ejecución.

Compatibilidad de software

Para que se produzca una migración, los equipos de origen y de destino deben ejecutar software compatible, de la siguiente manera:

- El hipervisor en el equipo de origen y de destino debe tener una versión compatible de firmware instalada: Véase “Required Software to Enable Oracle VM Server for SPARC 2.0 Features” en *Notas sobre la versión del servidor Oracle VM para SPARC 2.0*.
- En ambos equipos debe haber una versión compatible de Administrador de Dominios lógicos.

Nota – La característica de migración fue lanzada por primera vez con el software Dominios lógicos 1.1 y el correspondiente firmware. Para más información sobre la última versión de firmware disponible para la plataforma, véase *Notas sobre la versión del servidor Oracle VM para SPARC 2.0*.

Autenticación para las operaciones de migración

Dado que la operación de migración se ejecuta en dos equipos, debe autenticarse un usuario en el host de origen y en el de destino. En especial, un usuario que no sea el superusuario debe poseer las autorizaciones `solaris.ldoms.read` y `solaris.ldoms.write`.

La interfaz de línea de comandos `ldm` para la migración permite que el usuario especifique un nombre de usuario alternativo opcional para la autenticación en el host de destino. Si no se especifica, se utiliza el nombre de usuario del usuario que ejecuta el comando de migración. En ambos casos, se envía un mensaje al usuario para que introduzca la contraseña del sistema de destino, a menos que se use la opción `-p` para iniciar una migración no interactiva.

Migración de un dominio

Puede usar el comando `ldm migrate-domain` para iniciar la migración de un dominio de un sistema a otro sistema.

Para más información sobre las opciones y operandos de la migración, véase la página de comandos `man ldm(1M)`.

Realización de una simulación

Se proporciona la opción `-n` en el subcomando `migrate-domain`, se realizan las comprobaciones, pero no se migra el dominio de origen. Cualquier requisito que no se cumpla se devuelve como error. Esto le permite corregir los errores de configuración antes de intentar una migración real.

Nota – Debido a la naturaleza dinámica de los dominios lógicos, puede efectuarse correctamente un simulación y fallar una migración y viceversa.

Realización de migraciones no interactivas

Hasta la versión 1.3 del software de Dominios lógicos, las migraciones eran operaciones interactivas. Cuando comenzaba la migración, se solicitaba la contraseña que se debía usar en el equipo de destino. Desde la versión 1.3 de Dominios lógicos, puede usar el comando `ldm migrate-domain -p filename` para iniciar una operación de migración no interactiva.

El nombre de archivo que especifica como argumento para la opción `-p` debe tener las siguientes propiedades:

- La primera línea del archivo debe contener la contraseña
- La contraseña debe ser texto sin formato.
- La longitud de la contraseña no debe superar los 256 caracteres

Se ignoran un carácter de nueva línea al final de la contraseña y todas las líneas que siguen a la primera línea.

El archivo en el que guarda la contraseña del equipo de destino debe estar protegido adecuadamente. Si desea almacenar contraseñas de esta manera, asegúrese de que los permisos de archivos están fijados para que el propietario de raíz, o un usuario con privilegios, puedan leer o escribir el archivo (400 o 600).

Migración de un dominio activo

Para que la migración de un dominio activo se realice con el software Oracle VM Server for SPARC 2.0, existe un determinado conjunto de requisitos y restricciones impuestos en el dominio lógico de origen, el equipo de origen y el equipo de destino. Las siguientes secciones describen estos requisitos y restricciones para cada uno de los tipos de recursos.

Nota – La operación de migración se acelera cuando el dominio `primary` en los sistemas de origen y de destino tienen unidades criptográficas asignadas. Desde la versión 1.3 de Dominios lógicos, puede acelerar la migración agregando más CPU virtuales a los dominios `primary` de los dos sistemas, el de origen y el de destino.

Migración de CPUs en un dominio activo

A continuación se indican los requisitos y restricciones en una CPU cuando se realiza una migración:

- Los equipos de origen y de destino deben tener el mismo tipo de procesador en ejecución a la misma frecuencia.
- El equipo de destino debe tener suficientes cables libres para acomodar el número de cables en uso por el dominio.

Son aplicables requisitos y restricciones adicionales en cualquiera de las siguientes condiciones:

- El sistema de destino no ejecuta al menos la versión 2.0 del Administrador de Dominios lógicos. En este caso, puede ver el siguiente mensaje durante la migración:

```
The target machine is running an older version of the domain
manager that does not support the latest migration functionality.
```

- El sistema de origen no ejecuta al menos la versión 2.0 del Administrador de Dominios lógicos. Debido al Administrador de Dominios lógicos de herencia en el dominio de origen no se puede detectar un error de coincidencia en el software, la migración prosigue sin generar un mensaje.
- El dominio de origen se ejecuta sobre una versión SO de Solaris de Oracle anterior al SO 10 9/10 Solaris de Oracle. En este caso, puede ver el siguiente mensaje durante la migración:

```
Domain ldom is not running an operating system that is
compatible with the latest migration functionality.
```

Si se cumplen cualquiera de estas condiciones, se aplican los siguientes requisitos y restricciones en la CPU:

- Deben asignarse núcleos completos para el dominio migrado. Si el número de cables en el dominio de origen es inferior a un núcleo completo, los cables extra no están disponibles para ningún dominio hasta después del reinicio del dominio migrado.
- Después de una migración, la reconfiguración dinámica de CPU (DR) se deshabilita para el dominio de destino hasta que se ha reiniciado. Después del reinicio, la DR de la CPU está disponible para ese dominio.
- El sistema de destino debe tener suficientes núcleos completos que sean totalmente libres para ofrecer el número de cables necesarios para el dominio migrado. Después de la migración, si un dominio migrado usa sólo parcialmente un núcleo completo, los cables extra no están disponibles para otros dominios hasta que no se haya reiniciado el dominio migrado.

Migración de memoria en un dominio activo

Debe haber suficiente memoria libre en el equipo de destino para acomodar la migración del dominio de origen. Además, a continuación se incluyen algunas propiedades que deben mantenerse a lo largo de la migración:

- Se deben crear el mismo número de bloques de memoria con una tamaño idéntico.
- No es necesario que coincidan las direcciones físicas de los bloques de memoria, pero deben mantenerse las mismas direcciones reales a lo largo de la migración.

El equipo de destino debe tener suficiente memoria libre para acomodar la migración del dominio de origen. Además, el diseño de la memoria disponible en el equipo de destino debe ser compatible con el diseño de memoria en el dominio de origen o fallará la migración.

En especial, si la memoria en el equipo de destino está fragmentada en múltiples rangos de direcciones pequeñas, pero el dominio de origen requiere un rango de dirección larga única, fallará la migración. El siguiente ejemplo ilustra este escenario. El dominio de destino tiene dos Gbytes de memoria libres en dos bloques de memoria:

```
# ldm list-devices memory
MEMORY
  PA          SIZE
  0x108000000 1G
  0x188000000 1G
```

El dominio de origen, `ldg-src`, también tiene dos Gbytes de memoria libre, pero está aparece como un solo bloque de memoria:

```
# ldm list -o memory ldg-src
NAME
ldg-src

MEMORY
  RA          PA          SIZE
  0x80000000  0x208000000  2G
```

En esta situación de diseño de la memoria, falla la migración:

```
# ldm migrate-domain ldg-src dt212-239
Target Password:
Unable to bind 2G memory region at real address 0x80000000
Domain Migration of LDom ldg-src failed
```

Nota – Después de la migración, la reconfiguración dinámica de memoria (DR) está deshabilitada para el dominio de destino hasta que se reinicia. Después de haber completado el reinicio, la DR de memoria se vuelve a habilitar para el dominio.

Migración de dispositivos de E/S físicos en un dominio activo

Los dispositivos virtuales que están asociado con dispositivos físicos pueden migrar. En cualquier caso, los dominios que tienen acceso directo a los dispositivos físicos no pueden migrar. Por ejemplo, no se pueden migrar dominios E/S.

Migración de dispositivos virtuales de E/S en un dominio activo

Todos los servicios virtuales de E/S (VIO) usados por el dominio de origen deben estar disponibles en el equipo de destino. En otras palabras, deben producirse las siguientes condiciones:

- Cada volumen lógico usado en el dominio lógico de origen debe estar disponible en el host de destino y debe referirse al mismo almacenamiento.



Precaución – Si el volumen lógico usado por el origen como dispositivo de inicio existe en el destino pero no se refiere al mismo almacenamiento, la migración parece haber sido efectuada correctamente, pero no se puede utilizar el equipo ya que no puede acceder al dispositivo de inicio. Debe pararse el dominio, corregirse el problema de configuración y después reiniciarse el dominio. En caso contrario, el dominio puede quedar en un estado de falta de coherencia.

- Para cada dispositivo de red virtual en el dominio de origen, debe existir un conmutador de red virtual en el host de destino, con el mismo nombre que el conmutador de red virtual al que está acoplado el dispositivo en el host de origen.

Por ejemplo, si `vnet0` en el dominio de origen está asociado al nombre de servicio del conmutador virtual `switch-y`, entonces debe haber un dominio lógico en el host de destino que ofrezca un servicio de conmutador virtual llamado `switch-y`.

Nota – Los conmutadores no tienen que estar conectados a la misma red para que se produzca la migración, a pesar de que el dominio migrado puede experimentar problemas de red si los conmutadores no están conectados a la misma red.

Las direcciones MAC usadas por el dominio de origen que están automáticamente en el rango asignado deben estar disponibles para su uso en el host de destino.

- Debe existir un servicio de concentrador de consola virtual (`vcc`) en el host de destino y tener al menos un puerto libre. Durante la migración se ignoran las restricciones de consola explícitas. La consola para el dominio de destino se crea usando el nombre del dominio de destino como grupo de consola y usando cualquier puerto disponible en el primer dispositivo `vcc` en el dominio de control. Si se produce un conflicto con un nombre de grupo predeterminado, falla la migración.

Migración de una entrada/salida híbrida de NIU en un dominio activo.

Puede efectuarse la migración de un dominio que usa recursos híbridos de E/S de NIU. Una restricción que especifique los recursos híbridos de E/S de NIU no es un requisito estricto de un dominio lógico. Si dicho dominio migra a un equipo que no tiene disponibles recursos de NIU, se conserva la restricción, pero no se ejecuta.

Migración de unidades criptográficas en un dominio activo

Desde la versión 1.3 de Dominios lógicos, puede migrar un dominio huésped que ha enlazado unidades criptográficas si ejecuta un sistema operativo que admite la reconfiguración dinámica de las unidades criptográficas (DR).

Las siguientes versiones de SO de Solaris de Oracle admiten una DR de unidad criptográfica:

- Como mínimo SO 10 10/09 de Solaris
- Al menos SO 10 5/08 de Solaris de Oracle más parche ID 142245-01

El principio de la migración, el Administrador de Dominios lógicos determina si el dominio de origen admite la DR de unidad criptográfica. Si se admite, el Administrador de Dominios lógicos intenta eliminar cualquier unidad criptográfica del dominio. Después de haber completado la migración, las unidades criptográfica se vuelven a agregar al dominio migrado.

Nota – Si no se pueden cumplir las restricciones para las unidades criptográficas en el equipo de destino, la operación de migración podría aún completarse satisfactoriamente. En este caso, el dominio puede acabar con menos unidades criptográficas de las que tenía antes de la operación de migración.

Reconfiguración retrasada en un dominio activo

Cualquier operación de reconfiguración retrasada activa en los host de origen o de destino evita una migración desde el principio. Las operaciones de reconfiguración retrasada se bloquean mientras una migración está en curso.

Migración mientras un dominio activo está en modo elástico.

Las migraciones de dominio no son admitidas para un equipo de origen o de destino en modo elástico. Si se está realizando una migración mientras el dominio está en modo de rendimiento y la normativa de administración de energía (PM) se ha fijado en modo elástico, el conmutador de normativa se aplaza hasta que se ha completado la migración. El comando de migración genera un error si el equipo de origen o de destino está en modo elástico y se intenta una migración de dominio.

Operaciones en otros dominios

Mientras una migración está en curso en un equipo, cualquier operación que pueda provocar una modificación en el estado o configuración del dominio que se está migrando se bloquea. Todas las operaciones en este mismo dominio así como las operaciones que enlazan o paran otros dominios en el equipo se bloquean.

Migración de dominios enlazados o inactivos

Dado que un dominio enlazado o inactivo no está en ejecución en el momento de la migración, hay menos restricciones que cuando migra un dominio activo.

La migración de un dominio enlazado requiere que el destino pueda cumplir las restricciones de CPU, memoria y E/S en el dominio de origen. En caso contrario, fallará la migración. La migración de un dominio inactivo no presenta estos requisitos. En cualquier caso, el destino debe cumplir las restricciones del dominio cuando posteriormente se intente un enlace. En caso contrario, fallará el enlace del dominio.

Migración de CPU en dominio enlazado o inactivo

Puede migrar un dominio enlazado o inactivo entre equipos que ejecuten diferentes tipos de procesador y equipos que estén en ejecución a diferentes frecuencias.

La imagen de SO de Solaris de Oracle en el huésped debe poder admitir el tipo de procesador en el equipo de destino.

Migración de entrada/salida virtual en un dominio enlazado o inactivo

Para un dominio inactivo, no se realizan comprobaciones contra las restricciones de entrada/salida virtuales (VIO). Así pues, no es necesario que hayan servidores VIO para que la migración se realice correctamente. Como sucede con cualquier dominio inactivo, los servidores VIO son necesarios y tienen que estar disponibles en el momento que el dominio se enlaza.

Migración de dispositivos de punto final PCIe en dominios enlazados o inactivos

No puede realizar una migración de dominio en un dominio E/S que está configurado con dispositivos de punto final PCIe.

Para más información sobre la características de E/S directa (DIO), véase [“Asignación de dispositivos de punto final PCIe” en la página 69.](#)

Seguimiento de una migración en curso

Cuando una migración está en curso, los dominios de origen y de destino se muestran de manera diferente en la salida de estado. La salida del comando `ldm list` indica el estado del dominio que está migrando.

La sexta columna en el campo `FLAGS` muestra uno de los siguientes valores:

- El dominio de origen muestra una `s` para indicar que es el origen de la migración.
- El dominio de destino muestra una `t` para indicar que es el destino de la migración.
- Si se produce un error que requiere la intervención del usuario, se muestra una `e`.

A continuación se muestra que `ldg-src` es el dominio de origen de la migración:

```
# ldm list ldg-src
NAME      STATE      FLAGS  CONS  VCPU  MEMORY  UTIL  UPTIME
ldg-src   suspended -n---s    1    1G      0.0%  2h 7m
```

A continuación se muestra que `ldg-tgt` es el dominio de destino de la migración:

```
# ldm list ldg-tgt
NAME      STATE      FLAGS  CONS  VCPU  MEMORY  UTIL  UPTIME
ldg-tgt   bound      -----t 5000    1    1G
```

En la forma larga de la salida de estado, se muestra información adicional sobre la migración. En el origen, el porcentaje de operación completada se muestra junto con el host de destino y el nombre de dominio. De manera parecida, en el destino, el porcentaje de operación completada se muestra junto con el host de origen y el nombre de dominio.

EJEMPLO 9-1 Seguimiento de una migración en curso

```
# ldm list -o status ldg-src
NAME
ldg-src

STATUS
  OPERATION  PROGRESS  TARGET
migration   17%      t5440-sys-2
```

Cancelación de una migración en curso

Una vez comienza la migración, si se interrumpe el comando `ldm` con una señal `KILL`, la migración finaliza. Se destruye el dominio de destino, y se reinicia el dominio de origen si estaba activo. Si se pierde el comando de intérprete de control del comando `ldm`, la migración continúa en segundo plano.

Una operación de migración también se puede cancelar desde el exterior usando el comando `ldm cancel -operation`. Esto finaliza la migración en curso, y el dominio de origen reanuda su función de dominio activo. El comando `ldm cancel -operation` debe ser inicializado desde el sistema de origen. En un determinado sistema, cualquier comando relacionado con la migración tiene un impacto en la operación de migración que ha sido iniciada por dicho sistema. Un sistema no puede controlar una operación de migración cuando está en el sistema de de destino.

Nota – Cuando se ha iniciado una migración, la suspensión del proceso `ldm(1M)` no pausa la operación, ya que es el daemon del Administrador de Dominios lógicos (`ldmd`) en los equipos de origen y de destino el que está efectuando la migración. El proceso `ldm` espera una señal del `ldmd` que indique que se ha completado la migración antes de volver.

Recuperación de una migración fallida

Si la conexión de red se pierde después de que el origen haya completado en envío de la información sobre el estado en tiempo de ejecución, pero antes de que el destino pueda confirmar que se ha reiniciado el dominio, también termina la operación de migración, y se pone el origen en estado de error. Esto indica que es necesaria la interacción del usuario para determinar si la migración se ha completado correctamente o no. En esta situación, debe seguir los siguientes pasos.

- Determinar si el dominio de destino se ha reanudado correctamente. El dominio de destino estará en uno de estos dos estados:
 - Si la migración se ha realizado correctamente, el dominio de destino estará en estado normal.
 - Si se ha producido un error en la migración, el destino limpia y destruye el dominio de destino.
- Si se reinicia el destino, es seguro destruir el dominio de origen en estado de error. Si el destino no está presente, el dominio de origen es aun la versión maestra del dominio, y debe recuperarse. Para ello, ejecute el comando de cancelación en el equipo de origen. Esto borra el estado de error y restablece el dominio de origen a las condiciones originales.

Ejemplos de migración

[Ejemplo 9-2](#) muestra un dominio, llamado `ldg1`, que puede migrar a un equipo llamado `t5440-sys-2`.

EJEMPLO 9-2 Migración de un dominio huésped

```
# ldm migrate-domain ldg1 t5440-sys-2
Target Password:
```

Para realizar esta migración sin que se solicite la contraseña, use el siguiente comando:

```
# ldm migrate-domain -p pfile ldg1 t5440-sys-2
```

La opción `-p` toma un nombre de archivo como argumento. El archivo especificado contiene la contraseña de superusuario para el destino. En este ejemplo, `pfile` contiene la contraseña para el destino, `t5440-sys-2`.

[Ejemplo 9-3](#) muestra que puede cambiarse el nombre de un dominio como parte de la migración. En este ejemplo, `ldg-src` es el dominio de origen, y se cambia su nombre a `ldg-tgt` en el equipo de destino (`t5440-sys-2`) como parte de la migración. Además, el nombre de usuario (`root`) en el equipo de destino se especifica de manera explícita.

EJEMPLO 9-3 Migración y cambio de nombre de un dominio huésped

```
# ldm migrate ldg-src root@t5440-sys-2:ldg-tgt
Target Password:
```

[Ejemplo 9-4](#) muestra el mensaje de fallo de muestra si el dominio de destino no tiene asistencia técnica de migración, esto es, si está ejecutando una versión de Dominios lógicos anterior a la 1.1.

EJEMPLO 9-4 Mensaje de error en la migración

```
# ldm migrate ldg1 t5440-sys-2
Target Password:
Failed to establish connection with ldmd(1m) on target: t5440-sys-2
Check that the 'ldmd' service is enabled on the target machine and
that the version supports Domain Migration. Check that the 'xmpp_enabled'
and 'incoming_migration_enabled' properties of the 'ldmd' service on
the target machine are set to 'true' using svccfg(1M).
```

[Ejemplo 9-5](#) muestra cómo obtener el estado de un dominio de destino mientras la migración está en curso. En este ejemplo, el equipo de origen es `t5440-sys-1`.

EJEMPLO 9-5 Obtención del estado del dominio de destino

```
# ldm list -o status ldg-tgt
NAME
ldg-tgt
```

EJEMPLO 9-5 Obtención del estado del dominio de destino (Continuación)

```
STATUS
OPERATION   PROGRESS   SOURCE
migration   55%        t5440-sys-1
```

[Ejemplo 9-6](#) muestra cómo obtener el estado analizable en el dominio de origen mientras la migración está en curso. En este ejemplo, el equipo de destino es t5440-sys-2.

EJEMPLO 9-6 Obtención del estado analizable del dominio de origen

```
# ldm list -o status -p ldg-src
VERSION 1.3
DOMAIN |name=ldg-src|
STATUS
|op=migration|progress=42|error=no|target=t5440-sys-2
```


Administración de recursos

Este capítulo contiene información sobre cómo realizar la administración de recursos en los sistemas Oracle VM Server for SPARC.

Este capítulo trata sobre los siguientes temas:

- “Reconfiguración de recursos” en la página 159
- “Asignación de recursos” en la página 161
- “Asignación de CPU” en la página 161
- “Uso de la reconfiguración dinámica de memoria” en la página 166
- “Uso de la administración de energía” en la página 174
- “Uso de la administración de recursos dinámicos” en la página 177
- “Enumeración de recursos de dominios” en la página 180

Reconfiguración de recursos

Un sistema que ejecuta el software del Oracle VM Server for SPARC puede configurar recursos, como CPU virtuales, dispositivos virtuales de E/S, unidades criptográficas y memoria. Algunos recursos pueden configurarse dinámicamente en un dominio en ejecución mientras que otros deben configurarse en un dominio parado. Si no puede configurarse dinámicamente un recurso en el dominio de control, primero debe iniciar una reconfiguración retrasada. La reconfiguración retrasada pospone las actividades de configuración hasta después del reinicio del dominio de control.

Reconfiguración dinámica

La reconfiguración dinámica (DR) habilita el agregado o la eliminación de recursos mientras el sistema operativo (SO) está en ejecución. La capacidad de realizar una DR de un determinado recurso depende de si lo admite el SO en ejecución en el dominio lógico.

Se admite la reconfiguración dinámica para los siguientes recursos:

- **CPU virtuales** – Admitidas en todas las versiones del SO 10 de Solaris de Oracle
- **Dispositivos virtuales de E/S** – Admitidos en al menos el SO 10 10/08 Solaris
- **Unidades criptográficas** – Admitidas en al menos el SO 10 9/10 Solaris de Oracle
- **Memoria** – Se admite desde la versión 2.0 del Oracle VM Server for SPARC (véase “[Uso de la reconfiguración dinámica de memoria](#)” en la página 166)
- **Dispositivos de E/S físicos** – No admitidos

Para usar la capacidad de DR, el daemon de Dominios lógicos DR, `drd`, debe estar en ejecución en el dominio que desea cambiar. Véase la página de comando `man drd(1M)` sobre el daemon de reconfiguración dinámica `drd(1M)`.

Reconfiguración retrasada

A diferencia de las operaciones de DR que se efectúan inmediatamente, las operaciones de reconfiguración retrasada se efectúan en las siguientes circunstancias:

- Después del siguiente reinicio del SO
- Después de un paro y una puesta en marcha de un dominio lógico

Desde la versión 1.2 del software de Dominios lógicos, las operaciones de reconfiguración retrasada se restringen al dominio de control. Para todos los otros dominios, debe parar el dominio para modificar la configuración a menos que el recurso puede reconfigurarse dinámicamente.

Desde la versión 2.0 del software de Oracle VM Server for SPARC, debe iniciar primero una reconfiguración retrasada en el dominio de control antes de realizar las operaciones de configuración de recursos. Puede iniciar la reconfiguración retrasada usando el comando `ldm start-reconf primary`.

Cuando está en curso una reconfiguración retrasada en el dominio de control, se aplazan otras solicitudes de reconfiguración para el dominio de control hasta que se reinicia, o se para e inicia de nuevo. Asimismo, cuando una reconfiguración retrasada está pendiente para el dominio de control, las solicitudes de reconfiguración para otros dominios lógicos están muy restringidas y fallarán con un mensaje de error específico.

El comando `ldm cancel-operation reconf` cancela las operaciones de reconfiguración retrasada en el dominio de control. Puede enumerar las operaciones de reconfiguración retrasada usando el subcomando `ldm list-domain`. Para más información sobre cómo usar la característica de reconfiguración retrasada, véase la página de comando `man ldm(1M)`.

Nota – Puede usar el comando `ldm cancel-operation reconf` si otros comandos `ldm remove-*` ya han realizado la operación de reconfiguración retrasada en los dispositivos virtuales de E/S. El comando `ldm cancel-operation reconf` falla en estas circunstancias.

Asignación de recursos

Desde la versión 2.0 del Oracle VM Server for SPARC, el mecanismo de asignación de recursos usa las restricciones y sugerencias de asignación de recursos para asignar los recursos a un dominio en el tiempo de enlace.

Una *restricción de asignación de recursos* es un requisito estricto que el sistema *debe* cumplir cuando asigna un recurso a un dominio. Si no se puede cumplir la restricción, fallan la asignación de recursos y el enlazado del dominio.

Una *sugerencia de asignación de recursos* es un requisito laxo que el sistema *intenta* cumplir cuando asigna un recurso a un dominio. La asignación de un recurso puede tener éxito y el dominio puede enlazarse incluso si se no puede cumplir totalmente la sugerencia. Esta situación puede producirse si el sistema puede asignar el recurso de una manera que no cumpla necesariamente el requisito.

Asignación de CPU

El mecanismo de asignación de CPU usa las siguientes restricciones y sugerencias para los recursos de CPU:

- **Restricción de núcleo completo.** Esta restricción especifica que las CPU virtuales se asignan a un dominio basándose en un número específico de núcleos de CPU. El sistema debe poder asignar el número específico de núcleos y debe poder asignar también todas las CPU virtuales de los núcleos asignados al dominio. Si el sistema no puede asignar el número específico de núcleos, el dominio no puede enlazar.
- **Número máximo de restricción de núcleos.** Esta restricción especifica el número máximo de núcleos que pueden ser asignados a un dominio enlazado o activo. La restricción se habilita automáticamente cuando se fija una restricción de núcleo completo en un dominio. En este caso, el número máximo de núcleos se fija automáticamente en el número de núcleos configurados cuando el dominio está inactivo. Actualmente, esta restricción no puede habilitarse de manera independiente de la restricción de núcleo completo, y el número máximo de núcleos no puede fijarse manualmente.
- **Sugerencia de afinidad de núcleos.** Esta sugerencia solicita que las CPU virtuales asignadas a un dominio provengan de los mismos núcleos de la CPU o del menor número posible de núcleos de la CPU. El sistema realiza todos los esfuerzos posibles para cumplir esta solicitud. El dominio no puede enlazar *sólo* si está disponible en el sistema un número no suficiente de CPU virtuales libres.

La sugerencia de afinidad de núcleo está habilitada de manera predeterminada y no puede deshabilitarse.

Nota – La restricción de núcleo completo y la sugerencia de afinidad de núcleo sólo dirigen la ubicación de las CPU virtuales en los núcleos. No dirigen la ubicación de un núcleo o de un chip en puntos de conexión.

Habilitación de la restricción de núcleo completo

La restricción de núcleo completo se habilita automáticamente cuando se especifican el número de núcleos que se deben asignar a un dominio. De manera predeterminada, se especifica una CPU virtual para asignar a un dominio. Sólo puede habilitar la restricción de núcleo completo en un dominio inactivo, pero no en un dominio que está enlazado o activo. Antes de habilitar la restricción de núcleo completo en el dominio de control, primero debe iniciar una reconfiguración retrasada.

Use el comando `ldm add-vcpu -c number`, `ldm set-vcpu -c number` o `ldm remove-vcpu -c number` para asignar o eliminar núcleos de CPU de o a un dominio. *number* especifica el número de núcleos de CPU y habilita la restricción de núcleo completo. Para más información, véase la página de comando `man ldm(1M)`.

Puede usar el comando `ldm add-vcpu -c number` o `ldm remove-vcpu -c number` en un dominio que se había configurado previamente con CPU virtuales. En este caso, el número existente de CPU virtuales se convierte automáticamente en el número correspondiente de núcleos. Esta conversión es posible sólo si el número existente de CPU virtuales es un múltiplo del número de CPU virtuales por núcleo. Si no es así, no puede realizarse la conversión, y falla el comando.

Nota – Si usa estos comandos para habilitar la restricción de núcleo completo en un dominio inactivo o en el dominio de control en modo de reconfiguración retrasada, también se fija el número máximo de núcleos. El número máximo de núcleos no se ve afectado si usa estos comandos en un dominio enlazado o activo.

Por ejemplo, un núcleo se compone de ocho CPU virtuales. Si un dominio tiene siete CPU virtuales asignadas, un comando `ldm add-vcpu -c` o `ldm remove-vcpu -c` no puede cumplir la restricción de núcleo completo. En cambio, puede usar el comando `set-vcpu -c` para especificar el número de núcleos y habilitar la restricción de núcleo completo.

El siguiente ejemplo habilita la restricción de núcleo completo en el dominio inactivo `ldg1`. El comando `ldm list` comprueba que la restricción de núcleo completo está habilitada.

```
primary# ldm add-vcpu -c 1 ldg1
primary# ldm list -o resgmt ldg1
NAME
```

```
ldg1
CONSTRAINT
  whole-core
  max-cores=1
```

Nota – Cuando se habilita la restricción de núcleo completo en un dominio, las unidades criptográficas que están asociadas con esos núcleos no se ven afectadas por las adiciones de núcleos. Así pues, el sistema no agrega o elimina automáticamente las unidades criptográficas asociadas a o del dominio. Asimismo, no puede eliminar núcleos si las unidades criptográficas correspondientes están asignadas al dominio.

Deshabilitación de la restricción de núcleo completo

Cuando se asignan a un dominio CPU virtuales en vez de núcleos, se deshabilita la restricción de núcleo completo. Sólo puede deshabilitar la restricción de núcleo completo en un dominio inactivo, no en un dominio que está enlazado o activo. Antes de deshabilitar la restricción de núcleo completo en el dominio de control, primero debe iniciar una reconfiguración retrasada.

Use el comando `ldm add-vcpu number`, `ldm set-vcpu number` o `ldm remove-vcpu number` para asignar o eliminar CPU virtuales de un dominio. *number* especifica el número de CPU virtuales y deshabilita la restricción de núcleo completo. Para más información, véase la página de comando [man ldm\(1M\)](#).

Puede usar el comando `ldm add-vcpu number` o `ldm rm-vcpu number` en un dominio que anteriormente estaba configurado con núcleos CPU. En este caso, el número existente de núcleos de CPU se convierten automáticamente en el número correspondiente de CPU virtuales.

Nota – Cuando deshabilita la restricción de núcleo completo, la restricción de núcleo máximo también se deshabilita de forma automática.

El siguiente ejemplo deshabilita la restricción de núcleo completo en el dominio inactivo `ldg1`.

```
primary# ldm set-vcpu 1 ldg1
```

Asignación de CPU al dominio de control

Para habilitar la restricción de núcleo completo en el dominio de control, el dominio de control debe estar en modo de reconfiguración retrasada. La habilitación de la restricción de núcleo completo en el dominio de control sólo tiene un resultado satisfactorio si hay suficientes núcleos de CPU disponibles para cumplir la restricción solicitada. Esto es, deben estar disponibles los núcleos no usados, los núcleos que ya son usados por el dominio de control, o

los núcleos que son parcialmente usados por el dominio de control. En caso contrario, la asignación de CPU en el dominio de control no cambia.

Nota – Cuando un dominio de control está en modo de reconfiguración retrasada, la restricción de núcleo completo y la configuración del número de núcleos también especifica el número máximo de núcleos.

El siguiente ejemplo habilita la restricción de núcleo completo en el dominio de control (primary). Primero, inicie una reconfiguración retrasada en el dominio de control. Después, asigne un número completo al dominio de control, y después reinicie el dominio para que se efectúen los cambios.

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

primary# ldm add-vcpu -c 1 primary
primary# reboot
```

Interacciones entre la restricción de núcleo completo y otras características del dominio

Esta sección describe las interacciones entre la restricción de núcleo completo y las siguientes características:

- “Reconfiguración dinámica de CPU” en la página 164
- “Administración de los recursos dinámicos” en la página 165
- “Migración de dominio” en la página 165
- “Administración de energía” en la página 165

Reconfiguración dinámica de CPU

La restricción de núcleo completo es totalmente compatible con la reconfiguración dinámica de CPU (DR). Cuando un dominio se define con la restricción de núcleo completo, puede usar el comando `ldm add-vcpu -c`, `ldm set-vcpu -c` o `remove-vcpu -c` para cambiar el número de núcleos en un dominio activo.

En cualquier caso, si un dominio enlazado o activo no está en modo de reconfiguración retrasada, el número de núcleos de éste no puede superar el número máximo de núcleos. El máximo se fija con la restricción de núcleos máximos, que se habilita automáticamente cuando el habilita la restricción de núcleo completo. Falla cualquier operación de la DR de CPU que no cumple la restricción de núcleo máximo.

Administración de los recursos dinámicos

La restricción de núcleo completo no es compatible con la administración de recursos dinámica (DRM). Cuando está habilitada una normativa de DRM en un dominio que usa la restricción de núcleo completo, la normativa se deshabilita automáticamente. La restricción de núcleo completo permanece habilitada.

A pesar de que la normativa de DRM no puede habilitarse cuando está en efecto la restricción de núcleo completo, aun puede definir una normativa de DRM para el dominio. Tenga en cuenta que cuando una normativa se deshabilita automáticamente, sigue estando activa. La normativa se vuelve a habilitar automáticamente cuando el dominio se reinicia sin la restricción de núcleo completo.

A continuación se indican las interacciones que pueden esperarse entre la restricción de núcleo completo y la DRM:

- Si se fija la restricción de núcleo completo en un dominio, se genera un mensaje de advertencia cuando intenta habilitar la normativa de DRM en ese dominio.
- Si está en efecto una normativa de DRM en un dominio inactivo, se le permite habilitar la restricción de núcleo completo en el dominio. Cuando el dominio se convierte en activo y se habilita la normativa, el sistema deshabilita automáticamente la normativa de DRM para el dominio.
- Si una normativa de DRM se habilita en un dominio activo o enlazado, no se le permite habilitar la restricción de núcleo completo.

Migración de dominio

La configuración de núcleo completo de CPU es incompatible con la migración del dominio. En cualquier caso, puede migrar un dominio que está configurado con núcleos completos de CPU. Para restablecer la restricción de núcleo completo después de dicha migración, pare el dominio y reconfigúrelo para la asignación de núcleos completos.

Administración de energía

La restricción de núcleo completo es totalmente compatible con los modos de rendimiento y elástico de la administración de energía (PM). Cuando está habilitado el modo elástico, el subsistema de la PM puede agregar o eliminar núcleos de CPU en dominios que están configurados con la restricción de núcleo completo. En este caso, la restricción de núcleo completo continúa cumpliéndose, y los dominios que usan dicha restricción continúan configurados sólo con núcleos completos.

Uso de la reconfiguración dinámica de memoria

La versión 2.0 del Oracle VM Server for SPARC introduce la reconfiguración dinámica de memoria (DR). Esta característica se basa en la capacidad y le permite agregar a o eliminar del dominio lógico activo una cantidad arbitraria de memoria.

A continuación se indican los requisitos y restricciones para el uso de la característica de DR de memoria:

- Puede realizar operaciones de DR de memoria en cualquier dominio. En cualquier caso, sólo una operación de DR de memoria individual puede estar en progreso en un dominio en un determinado momento.
- La característica de DR de memoria refuerza la alineación de 256 Mbytes en las direcciones y el tamaño de la memoria implicada en una determinada operación. Véase [“Alineación de memoria” en la página 168](#).
- La memoria no alineada en el grupo de memoria libre *no puede* ser asignada a un dominio usando la característica de DR de memoria. Véase [“Agregación de memoria no alineada” en la página 169](#).

Si la memoria de un dominio no puede reconfigurarse usando una operación de DR de memoria, el dominio debe pararse antes de que se reconfigure la memoria. Si el dominio es el dominio de control, debe comenzar una reconfiguración retrasada.

Agregación de memoria

Si un dominio está activo, puede usar el comando `ldm add-memory` para agregar dinámicamente memoria al dominio. El comando `ldm set-memory` también puede agregar dinámicamente memoria si el tamaño de memoria especificado es superior al tamaño de memoria actual del dominio.

Eliminación de memoria

Si un dominio está activo, puede usar el comando `ldm remove-memory` para eliminar dinámicamente la memoria del dominio. El comando `ldm set-memory` también puede eliminar dinámicamente memoria si el tamaño de memoria especificado es inferior al tamaño de memoria actual del dominio.

La eliminación de memoria puede ser una operación larga. Puede efectuar un seguimiento del progreso de una operación o cancelar una solicitud de DR de memoria en curso.

Seguimiento del progreso de la solicitud de DR de memoria

Puede efectuar un seguimiento de un comando `ldm remove-memory` ejecutando el comando `ldm list -l` para el dominio especificado.

Cancelación de una solicitud de DR de memoria

Puede cancelar una solicitud de eliminar que está en progreso interrumpiendo el comando `ldm remove-memory` (pulsando Control-C) o generando el comando `ldm cancel-operation memdr`. Si cancela una solicitud de memoria, sólo la parte que sobra de la solicitud de eliminación se ve afectada, esto es, la cantidad de memoria que aun debe ser eliminada del dominio.

Solicitudes parciales de DR de memoria

Se rechaza una solicitud de adición de memoria si no hay suficiente memoria libre para cumplir toda la solicitud. En cualquier caso, una solicitud de adición de memoria puede cumplirse parcialmente si el dominio de destino no puede agregar parte de la memoria solicitada por el Administrador de Dominios lógicos.

Se rechaza una solicitud de eliminación de memoria si la memoria en el dominio es insuficiente para cumplir toda la solicitud. En cualquier caso, una solicitud de eliminación de memoria puede cumplirse parcialmente si el dominio de destino no puede eliminar parte de la memoria solicitada por el Administrador de Dominios lógicos.

Nota – La memoria se borra después de haber sido eliminada de un dominio y antes de ser agregada a otro dominio.

Reconfiguración de memoria del dominio de control

Esta característica de DR de memoria puede usarse para reconfigurar la memoria del dominio de control. Si no puede realizarse una solicitud de DR de memoria en el dominio de control, primero debe iniciar una reconfiguración retrasada.

El uso de una DR de memoria puede no ser adecuado para la eliminación de grandes cantidades de memoria de un dominio activo ya que las operaciones de DR de memoria pueden ser largas. En especial, durante la configuración inicial del sistema, debe usar la reconfiguración retrasada para disminuir la memoria en el dominio de control.

Disminución de la memoria del dominio de control

Use una reconfiguración retrasada en vez de una DR de memoria para disminuir la memoria del dominio de control de la configuración predeterminada de fábrica inicial. En este caso, el dominio de control posee toda la memoria del sistema host. La característica de DR de memoria no es adecuada para este objetivo ya que no se garantiza que un dominio activo agregue o más típicamente dé toda la memoria solicitada. Más bien, la ejecución del sistema operativo en ese dominio hace todo lo posible para cumplir la solicitud. Además, la eliminación de la memoria

puede ser una operación larga. Estos temas se amplían cuando están implicadas operaciones de memoria de gran tamaño, como en el caso para la disminución inicial de la memoria del dominio de control.

Por estas razones, use una reconfiguración retrasada usando los siguientes pasos:

1. Use el comando `ldm start -reconf primary` para poner el dominio de control en modo de reconfiguración retrasada.
2. Efectúe la partición de los recursos del sistema host que pertenecen al dominio de control, si es necesario.
3. Use el comando `ldm cancel -reconf` para deshacer las operaciones del paso 2, si es necesario, y volver a empezar.
4. Reinicie el dominio de control para hacer que se realicen los cambios de la configuración.

Reconfiguración dinámica y retrasada

Si está pendiente una reconfiguración retrasada en el dominio de control, se rechaza una solicitud de reconfiguración de memoria para cualquier otro dominio. Si no está pendiente una reconfiguración retrasada en el dominio de control, se rechaza una reconfiguración de memoria para cualquier dominio que no admita la DR de memoria. Una solicitud de reconfiguración de memoria en un dominio de control que no admite la DR de memoria se convierte en una solicitud de reconfiguración retrasada.

Alineación de memoria

Las solicitudes de reconfiguración de memoria tienen diferentes requisitos de alineación que dependen del estado del dominio al que se aplica la solicitud.

Alineación de memoria para dominios activos

- **Adición y eliminación dinámica.** La dirección y el tamaño del bloque de memoria son de 256 Mbytes alineados para la adición dinámica y eliminación dinámica. El tamaño de funcionamiento mínimo es de 256 Mbytes.

Se rechaza una solicitud no alineada o una solicitud de eliminación que es superior al tamaño enlazado.

Use los siguientes comandos para ajustar las asignaciones de memoria:

- `ldm add-memory`. Si especifica la opción `--auto-adj` con este comando, la cantidad de memoria que se debe agregar está alineada a 256 Mbytes, lo que puede aumentar la cantidad de memoria realmente agregada al dominio.
- `ldm remove-memory`. Si especifica la opción `--auto-adj` con este comando, la cantidad de memoria que se debe eliminar está alineado a 256 Mbytes, lo que puede disminuir la cantidad de memoria realmente eliminada del dominio.
- `ldm set-memory`. Este comando se trata como una operación de adición o eliminación. Si especifica la opción `--auto-adj`, la cantidad de memoria agregada o eliminada está alineada a 256 Mbytes como anteriormente descrito. Tenga en cuenta que esta alineación puede aumentar el tamaño de la memoria resultante del dominio.
- **Reconfiguración retrasada.** La dirección y el tamaño de un bloque de memoria están alineadas a 4 Mbytes. Si realiza una solicitud de no alineación, la solicitud se redondea para que esté alineada a 4 Mbytes.

Alineación de memoria para dominios enlazados

La dirección y el tamaño del bloque de memoria están alineadas a 4 Mbytes para dominios enlazados. Si realiza una solicitud de no alineación, la solicitud se redondea para que esté alineada a 4 Mbytes. Esto significa que el tamaño de la memoria del dominio resultante puede ser un poco más de lo que se ha especificado.

Para los comandos `ldm add-memory`, `ldm set-memory` y `ldm remove-memory`, la opción `--auto-adj` redondea el tamaño de la memoria resultante para que esté alineada a 256 Mbytes. Esto significa que el tamaño de la memoria resultante puede ser ligeramente superior a lo que se ha especificado.

Alineación de memoria para dominios inactivos

Para los comandos `ldm add-memory`, `ldm set-memory` y `ldm remove-memory`, la opción `--auto-adj` redondea el tamaño de la memoria resultante para que esté alineada a 256 Mbytes. No existe un requisito de alineación para un dominio inactivo. Las restricciones descritas en [“Alineación de memoria para dominios enlazados” en la página 169](#) se efectúan después de que se haya enlazado dicho dominio.

Agregación de memoria no alineada

La característica de DR de memoria aplica la alineación de memoria de 256 Mbytes en la dirección y el tamaño de la memoria que se agrega o elimina dinámicamente de un dominio activo. Esto significa que una memoria no alineada en un dominio activo no puede ser eliminada usando la DR de memoria.

Esto también significa que una memoria no alineada en el grupo de memoria libre no puede agregarse a un dominio activo usando la DR de memoria.

Después de la asignación de la memoria alineada, puede usar el comando `ldm add-memory` para agregar la memoria no alineada restante a un dominio enlazado o inactivo. También puede usar este comando para agregar la memoria no alineada restante al dominio de control con la operación de reconfiguración retrasada.

El siguiente ejemplo muestra cómo agregar los dos bloques de memoria de 128-Mbyte restantes a los dominios `primary` y `ldom1`. El dominio `ldom1` está en estado enlazado. Los siguientes comandos agregan los dos bloques de memoria restantes. El primero comando comienza una operación de reconfiguración retrasada en el dominio de control. El segundo comando agrega uno de los bloques de memoria de 128-Mbyte al dominio de control. El quinto comando agrega el otro bloque de memoria de 128-Mbyte al dominio `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
```

Ejemplos de DR de memoria

Los siguientes ejemplos muestran cómo realizar operaciones DR de memoria. Para más información sobre los comandos de la CLI relacionados, véase la página de comando [man ldm\(1M\)](#).

EJEMPLO 10-1 Operaciones DR de memoria en dominios activos

Este ejemplo muestra cómo agregar memoria dinámicamente y eliminarla de un dominio activo, `ldom1`.

EJEMPLO 10-1 Operaciones DR de memoria en dominios activos (Continuación)

La salida `ldm list` muestra la memoria para cada dominio en el campo Memoria. El primer comando `ldm add-mem` sale con un error porque debe especificar memoria en múltiplos de 256 Mbytes. El siguiente comando `ldm add-mem` usa la opción `--auto-adj` de manera que incluso si especifica 200M como la cantidad de memoria que se debe agregar, la cantidad se redondea hasta 256 Mbytes.

El comando `ldm rm-mem` sale con un error ya que debe especificar la memoria en múltiplos de 256 Mbytes. Cuando agrega la opción `--auto-adj` al mismo comando, se produce satisfactoriamente la eliminación de memoria, y la cantidad de memoria se redondea al siguiente límite de 256 Mbytes.

```
# 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
```

EJEMPLO 10-2 Operaciones DR de memoria en dominios enlazados

Este ejemplo muestra cómo agregar y eliminar memoria de un dominio activo, `ldom2`.

La salida `ldm list` muestra la memoria para cada dominio en el campo Memoria. El primer comando `ldm add-mem` agrega 100 Mbytes de memoria al dominio `ldom2`. El siguiente comando `ldm add-mem` especifica la opción `--auto-adj`, lo que provoca la agregación de 112 Mbytes de memoria adicionales que se agregan dinámicamente a `ldom2`.

EJEMPLO 10-2 Operaciones DR de memoria en dominios enlazados *(Continuación)*

El comando `ldm rm-mem` elimina dinámicamente 100 Mbytes del dominio `ldom2`. Si especifica la opción `--auto-adj` en el mismo comando para eliminar 300 Mbytes de memoria, la cantidad de memoria se redondea al límite de los siguientes 256 Mbytes.

```
# 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
```

EJEMPLO 10-3 Configuración de los tamaños de memoria del dominio

Este ejemplo muestra cómo usar el comando `ldm set -memory` para agregar memoria y para quitarla de un dominio.

La salida `ldm list` muestra la memoria para cada dominio en el campo Memoria. El primer comando `ldm set -mem` intenta fijar el tamaño del dominio `primary` a 3400 Mbytes. El error

EJEMPLO 10-3 Configuración de los tamaños de memoria del dominio (Continuación)

resultante indica que el valor especificado no está en el límite de 256 Mbytes. Si se agrega la opción `--auto-adj` al mismo comando le permite eliminar correctamente alguna memoria y permanecer dentro del límite de 256 Mbytes. Este comando también envía un aviso para indicar que no se ha podido eliminar toda la memoria solicitada ya que el dominio ha usado esa memoria.

El siguiente comando `ldm set -mem` fija el tamaño de la memoria del dominio `ldom2`, que está en estado enlazado, a 690 Mbytes. Si agrega la opción `--auto-adj` al mismo comando, se agregan dinámicamente 78 Mbytes de memoria a `ldom2` para mantenerse en el límite de 256 Mbytes.

```
# 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
```

Uso de la administración de energía

Para usar la administración de energía (PM), primero necesita fijar el modo PM en el firmware Integrated Lights Out Manager (ILOM) 3.0 de Oracle. Esta sección resume la información que necesita para poder usar la PM con el software del Oracle VM Server for SPARC.

Para más información acerca del ILOM, véase el siguiente párrafo:

- “Monitoring Power Consumption” en *Sun Integrated Lights Out Manager (ILOM) 3.0 CLI Procedures Guide* (<http://d1c.sun.com/pdf/820-6412-12/820-6412-12.pdf>)
- *Oracle Integrated Lights Out Manager (ILOM) 3.0 Feature Updates and Release Notes* (<http://d1c.sun.com/pdf/820-7329-17/820-7329-17.pdf>)

El modo de energía es la configuración que gobierna el uso de energía del sistema en cualquier momento. Desde la versión 1.3 de Dominios lógicos, se admiten los siguientes modos de energía, asumiendo que la plataforma subyacente ha implementado las características de la PM:

- **Modo de rendimiento.** El sistema puede usar toda la energía que está disponible.
- **Modo elástico:** El uso de energía del sistema se adapta al nivel de utilización actual. Por ejemplo, el estado de energía de los recursos se reduce a medida que disminuye la utilización.

A continuación se indican las características de la PM:

- **Deshabilitación automática de núcleo de CPU.** La PM deshabilita automáticamente un núcleo de CPU cuando todos los cables en dicho núcleo se han deshabilitado.
- **Omisión de ciclo de reloj de CPU.** Desde la versión 2.0 del Oracle VM Server for SPARC, la PM puede ajustar automáticamente la omisión del ciclo de reloj de CPU en las plataformas SPARC T3. El ajuste puede aumentar o disminuir el número de ciclos de reloj que se omiten para mantener todos los dominios dentro de los umbrales de utilización de energía. La PM determina si realizar estos ajustes basándose en la utilización de la CPU. Cuando el sistema entra en modo de rendimiento, el número de ciclos de reloj que se saltan se ajusta automáticamente a ninguno.
- **Operaciones de memoria en modo de reposo profundo.** Desde la versión 2.0 del Oracle VM Server for SPARC, cuando las plataformas SPARC T3 están en modo elástico, se configura automáticamente la memoria infrautilizada para funcionar en modo de reposo profundo para ahorrar energía.
- **Límite de energía.** Desde la versión del Oracle VM Server for SPARC 2.0, puede fijar un *límite de energía* en las plataformas SPARC T3 para restringir el consumo de energía de un sistema. Si el consumo de energía es superior al límite de energía, las técnicas de la PM se usan para reducir la energía. Puede usar el procesador de servicios (SP) de ILOM para fijar el límite de energía.

Véanse los siguientes documentos:

- *Sun Integrated Lights Out Manager (ILOM) 3.0 CLI Procedures Guide* (<http://dlc.sun.com/pdf/820-6412-12/820-6412-12.pdf>)
- *Oracle Integrated Lights Out Manager (ILOM) 3.0 Feature Updates and Release Notes* (<http://dlc.sun.com/pdf/820-7329-17/820-7329-17.pdf>)

Puede usar la interfaz ILOM para fijar un límite de energía, un periodo de gracia y una acción de violación. Si se supera el límite de energía durante un intervalo superior al periodo de gracia, se realiza la acción de violación.

Si el consumo de energía actual supera el límite de energía, se intenta reducir el estado de energía de los recursos que pueden ser administrados por energía. Si el consumo de energía baja por debajo del límite de energía, se permite un aumento del estado de energía de dicho recursos. Si el sistema está en modo elástico, un aumento en el estado de energía de los recursos está controlado por el nivel de utilización.

Cuando el sistema está en modo elástico, algunas modificaciones en la configuración del dominio son validadas primero para confirmar que no se supera el límite de energía. Si se supera el límite de energía, sólo pueden modificarse o agregarse algunos de los recursos según solicitado. Si el límite de energía aumenta posteriormente, entonces puede agregar cualquier recurso que no se habían modificado correctamente.

Si la carga de un dominio hace que se consuma más energía, sólo se encienden correctamente los recursos que mantienen el consumo de energía bajo el límite de energía.

Para las instrucciones sobre cómo configurar el modo de energía usando la CLI del firmware 3.0 de ILOM, véase “Monitoring Power Consumption” en *Sun Integrated Lights Out Manager (ILOM) 3.0 CLI Procedures Guide* (<http://dlc.sun.com/pdf/820-6412-12/820-6412-12.pdf>).

Enumeración de cables con administración de energía de CPU

Esta sección muestra cómo enumerar los cables administrados por energía y los CPU virtuales.

▼ Lista de cables con administración de energía de CPU

- Enumera los cables administrados por energía usando uno de los siguientes comandos:

a. Use el subcomando `list -l`.

En la salida, un guión (---) en la columna UTIL para la CPU significa que el cable está administrado por energía.

```
# ldm list -l primary
NAME          STATE  FLAGS  CONS  VCPU  MEMORY  UTIL  UPTIME
primary      active -n-cv  SP    8     4G      4.3%  7d 19h 43m

SOFTSTATE
Solaris running

MAC
    00:14:4f:fa:ed:88

HOSTID
    0x84faed88

CONTROL
    failure-policy=ignore

DEPENDENCY
    master=

VCPU
    VID    PID    UTIL  STRAND
    0      0      0.0%  100%
    1      1      ---   100%
    2      2      ---   100%
    3      3      ---   100%
    4      4      ---   100%
    5      5      ---   100%
    6      6      ---   100%
    7      7      ---   100%
    ....
```

b. Use la opción analizable (-p) en el subcomando `list -l`.

En la salida, un espacio en blanco después de `util=` significa que el cable está administrado por energía.

```
# 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
```

▼ Enumeración de CPU administradas por energía

- Enumera las CPU administradas por energía usando uno de los siguientes comandos:

a. Use el comando `list-devices -a cpu`.

En la columna PM de la salida, un yes significa que la CPU está administrada por energía, un no significa que la CPU está encendida. Se considera que las CPU al 100% libres son administradas por energía de manera predeterminada, por lo tanto se incluye el guión (---) debajo de la columna PM para éstas.

```
# 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. Use la opción analizable (-p) en el subcomando `list-devices -a cpu`.

En el campo pm= en la salida, un yes significa que la CPU está administrada por energía y un no significa que la CPU está encendida. Se considera que las CPU libres al 100% están administradas por energía de manera predeterminada, de ahí el espacio en blanco en este campo para éstas.

```
# ldm list-devices -a -p cpu
VERSION 1.4
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=
```

Uso de la administración de recursos dinámicos

Desde la versión 1.3 del programa Dominios lógicos, puede usar normativas para determinar cómo realizar automáticamente actividades de DR. En este momento, *sólo* puede crear normativas para gobernar la administración de recursos dinámicos de las CPU virtuales.



Precaución – Los siguientes temas afectan la administración de recursos dinámicos (DRM):

- Cuando la PM está en modo elástico, la DRM no puede habilitarse.
- Cualquier cambio del modo de rendimiento al modo elástico se retrasa mientras la DRM está habilitada.
- Asegúrese de que deshabilita la DRM de la CPU antes de realizar la operación de migración del dominio.
- Las normativas DRM no se aplican a dominios que están configurados con la restricción de núcleo completo.

Una *normativa de administración de recursos* especifica bajo qué condiciones las CPU virtuales pueden agregarse y eliminarse automáticamente de un dominio lógico. Una normativa se administra usando los comandos `ldm add-policy`, `ldm set-policy` y `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
```

Para más información sobre estos comandos y sobre la creación de normativas de administración de recursos, véase la página de comando `man ldm(1M)`.

Una normativa está en efecto durante los intervalos de tiempo especificados en las propiedades `tod-begin` y `tod-end`. La normativa usa el valor de propiedad `priority` para determinar qué normativa usar si más de una normativa está en efecto simultáneamente.

La normativa usa los valores de propiedad `util-high` y `util-low` para especificar los umbrales alto y bajo para la utilización de la CPU. Si la utilización supera el valor de `util-high`, se agregan CPU virtuales hasta que el número está entre los valores `vcpu-min` y `vcpu-max`. Si la utilización baja por debajo del valor `util-low`, se eliminan las CPU virtuales del dominio hasta que el número se sitúa entre los valores `vcpu-min` y `vcpu-max`. Si se alcanza `vcpu-min`, no pueden eliminarse dinámicamente más CPU virtuales. Si se alcanza `vcpu-max`, no pueden agregarse dinámicamente más CPU virtuales.

EJEMPLO 10-4 Agregado de normativas de administración de recursos

Por ejemplo, después de observar la utilización típica de los sistemas a lo largo de varias semanas, puede configurar normativas para optimizar el uso de los recursos. El uso más alto es diariamente de 9:00 a.m. a 6:00 p.m. hora local, y el uso más bajo es diariamente de 6:00 p.m. a 9:00 a.m. hora local.

EJEMPLO 10-4 Agregado de normativas de administración de recursos (Continuación)

Basándose en la observación de esta utilización del sistema, decide crear las siguientes normativas altas y bajas basándose en la utilización general del sistema:

- **Alta:** Diariamente de 9:00 a.m. a 6:00 p.m. hora local
- **Baja:** Diariamente de 6:00 p.m. a 9:00 a.m. hora local

El siguiente comando `ldm add-policy` crea la normativa de `high-usage` que debe usarse durante el periodo de más utilización en el periodo `ldom1`.

La siguiente normativa `high-usage` realiza los siguientes pasos:

- Especifica que las horas de comienzo y final son 9:00 a.m. y 6:00 p.m. configurando las propiedades `tod-begin` y `tod-end`, respectivamente.
- Especifica que los límites inferior y superior en los que realizar los análisis de normativa son 25 por ciento y 75 por ciento configurando las propiedades `util-lower` y `util-upper`, respectivamente.
- Especifica que el número mínimo y máximo de CPU virtuales es 2 y 16 configurando las propiedades `vcpu-min` y `vcpu-max`, respectivamente.
- Especifica que el número máximo de CPU virtuales que pueden agregarse durante un ciclo de control de recursos es 1 fijando la propiedad `attack`.
- Especifica que el número máximo de CPU virtuales que pueden eliminarse durante un ciclo de control de recursos es 1 fijando la propiedad `decay`.
- Especifica que la prioridad de esta normativa es 1 configurando la propiedad `priority`. Una prioridad de 1 significa que esta normativa se aplicará incluso si otra normativa puede efectuarse.
- Especifica que el nombre del archivo de la normativa es `high-usage` configurando la propiedad del `name`.
- Usa los valores predeterminados para las propiedades que no se especifican, como `enable` y `sample-rate`. Véase la página de comando `man 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
```

El siguiente comando `ldm add-policy` crea una normativa `med-usage` que se debe usar durante el periodo de baja utilización en el dominio `ldom1`.

La siguiente normativa `med-usage` realiza las siguientes acciones:

- Especifica que las horas de comienzo y final son 6:00 p.m. y 9:00 a.m. configurando las propiedades `tod-begin` y `tod-end`, respectivamente.
- Especifica que los límites inferior y superior en los que realizar los análisis de normativa son 10 por ciento y 50 por ciento configurando las propiedades `util-lower` y `util-upper`, respectivamente.

EJEMPLO 10-4 Agregado de normativas de administración de recursos (Continuación)

- Especifica que el número mínimo y máximo de CPU virtuales es 2 y 16 configurando las propiedades `vcpu-min` y `vcpu-max`, respectivamente.
- Especifica que el número máximo de CPU virtuales que pueden agregarse durante un ciclo de control de recursos es 1 fijando la propiedad `attack`.
- Especifica que el número máximo de CPU virtuales que pueden eliminarse durante un ciclo de control de recursos es 1 fijando la propiedad `decay`.
- Especifica que la prioridad de esta normativa es 1 configurando la propiedad `priority`. Una prioridad de 1 significa que esta normativa se aplicará incluso si otra normativa puede efectuarse.
- Especifica que el nombre del archivo de la normativa es `high-usage` configurando la propiedad del `name`.
- Usa los valores predeterminados para las propiedades que no se especifican, como `enable` y `sample-rate`. Véase la página de comando `man 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
```

Enumeración de recursos de dominios

Esta sección muestra que el uso de la sintaxis para los subcomandos `ldm`, define algunos términos de salida, como los indicadores y las estadísticas de utilización y ofrece ejemplos que son parecidos a los que realmente ve como salida.

Salida informatizada

Si está creando secuencias de comandos que usan la salida de comando `ldm list`, use *siempre* la opción `-p` para obtener una forma informatizada de la salida. Véase “[Generar una lista analizable informatizada \(-p\)](#)” en la página 182 para más información.

▼ Mostrar el uso de la sintaxis para los subcomandos `ldm`

- Mire el uso de la sintaxis para todos los subcomandos `ldm`.

```
primary# ldm --help
```

Para más información sobre los subcomandos `ldm`, véase la página de comando `man ldm(1M)`.

Definiciones de marcadores

Los siguientes indicadores pueden mostrarse en la salida para un dominio (`ldm list`). Si utiliza opciones largas analizables (`-l -p`) para el comando, los indicadores se escriben con el nombre completo, por ejemplo, `flags=normal, control, vio-service`. Si no es así, se muestra la abreviación de la letra, por ejemplo `-n -cv-`. Los valores de la etiqueta de la lista dependen de la posición. A continuación se incluyen los valores que pueden aparecer en cada una de las seis columnas de izquierda a derecha.

Columna 1

- `s` inicio o paro
- `-` marcador de posición

Columna 2

- `n` normal
- `t` transición

Columna 3

- `d` reconfiguración retrasada
- `r` reconfiguración de memoria dinámica (DR)
- `-` marcador de posición

Columna 4

- `c` dominio de control
- `-` marcador de posición

Columna 5

- `v` dominio de servicios E/S virtual
- `-` marcador de posición

Columna 6

- `s` dominio de origen en una migración
- `t` dominio de destino en una migración
- `e` error ocurrido durante una migración
- `-` marcador de posición

Utilización de la definición estadística

La estadística de utilización de CPU virtual (UTIL) se muestra en la opción larga del comando (`-l`) `ldm list`. La estadística es el porcentaje de tiempo que la CPU ha gastado ejecutando en nombre del sistema operativo huésped. Se considera que una CPU virtual está en ejecución en nombre del sistema operativo huésped excepto cuando ha sido proporcionada al hypervisor. Si

el sistema operativo huésped no proporciona las CPU virtuales al hypervisor, la utilización de las CPU en el sistema operativo huésped siempre se mostrará como 100%.

Las estadísticas de utilización indicadas para un dominio lógico es la media de las utilizaciones de las CPU virtuales en el dominio. Un guión (- - -) en la columna UTIL significa que el cable está administrado por energía.

Ver varias listas

▼ **Mostrar versiones de software (-V)**

- Ver las versiones de software actualmente instaladas.

```
primary# ldm -V
```

▼ **Generar una lista corta**

- Genera una lista corta para todos los dominios.

```
primary# ldm list
```

▼ **Generar una lista larga (-l)**

- Genera una lista larga para todos los dominios.

```
primary# ldm list -l
```

▼ **Generar lista extendida (-e)**

- Genera una lista extendida de todos los dominios.

```
primary# ldm list -e
```

▼ **Generar una lista analizable informatizada (-p)**

- Genera una lista analizable informatizada de todos los dominios.

```
primary# ldm list -p
```

▼ **Generar un subconjunto de lista larga (-o *format*)**

- Genera una salida de un subconjunto de recursos introduciendo una o varias de las siguientes opciones *format*. Si especifica más de un formato, delimite los elementos con una coma sin espacios.

```
primary# ldm list -o resource[,resource...] ldom
```

- `console` – La salida contiene consola virtual (`vcons`) y un servicio de concentrador de consola virtual (`vcc`)
- `core` – La salida contiene información sobre los dominios que tienen núcleos completos asignados
- `cpu` – La salida contiene información sobre las CPU virtuales (`vcpu`), CPU físicas (`pcpu`) e id de núcleo
- `crypto` – La salida de la unidad criptográfica contiene una unidad aritmética modular (`mau`) y otras unidades criptográficas admitidas por LDoms, como el Control Word Queue (`CWQ`)
- `disk` – La salida contiene disco virtual (`vdisk`) y servidor de disco virtual (`vds`)
- `domain` – La salida contiene variables (`var`), id del host (`hostid`), estado del dominio, indicadores, UUID y estado del software
- `memory` – La salida contiene `memory`
- `network` – La salida contiene direcciones de control de acceso a los medios (`mac`), conmutador de red virtual (`vsw`) y dispositivo de red virtual (`vnet`)
- `physio` – La entrada/salida física contiene interconexiones con los componentes periféricos (`pci`) y unidad de interfaz de red (`niu`)
- `resgmt` – La administración contiene la información sobre la normativa de administración de recursos dinámicos (DRM), indica qué normativa se está ejecutando en ese momento y enumera las restricciones relacionadas con la configuración de núcleo completo
- `serial` – La salida contiene un servicio de canal de dominio lógico virtual (`vldc`), un cliente de canal de dominio lógico virtual (`vldcc`), un cliente de canal plano de datos virtuales (`vdpsc`), un servicio de canal plano de datos virtuales (`vdpcs`)
- `stats` – La salida contiene estadísticas que están relacionadas con las normativas de administración de recursos
- `status` – La salida contiene estados sobre la migración de dominio en curso

Los siguientes ejemplos muestran varios subconjuntos de salida que puede especificar:

- Enumerar la información de la CPU para el dominio de control


```
# ldm list -o cpu primary
```
- Enumerar la información del dominio para un dominio huésped


```
# ldm list -o domain ldm2
```
- Enumerar la información de la memoria y red para un dominio huésped


```
# ldm list -o network,memory ldm1
```
- Enumerar la información de la normativa de DRM para un dominio huésped


```
# ldm list -o resgmt,stats ldm1
```

▼ Enumerar una variable

- Muestra una variable y el valor de ésta para un dominio.

```
primary# ldm list-variable variable-name ldom
```

Por ejemplo, el siguiente comando muestra el valor para la variable `boot-device` en el dominio `ldg1`:

```
primary# ldm list-variable boot-device ldg1
boot-device=/virtual-devices@100/channel-devices@200/disk@0:a
```

▼ Enumerar enlaces

- Enumera los recursos enlazados a un dominio.

```
primary# ldm list-bindings ldom
```

▼ Enumerar configuraciones

- Enumera las configuraciones de un dominio lógico que se han guardado en el SP.

Ejemplo 10-5 Lista de configuraciones

El comando `ldm list-config` enumera las configuraciones del dominio lógico que están almacenadas en el procesador de servicios. Cuando se usa con la opción `-r`, este comando enumera las condiciones que existe en los archivos de autoguardado en el dominio de control.

Para más información sobre las configuraciones, véase [“Administración de las configuraciones Dominios lógicos” en la página 190](#). Para más información, véase la página de comando `man ldm(1M)`.

```
primary# ldm list-config
factory-default
3guests
foo [next poweron]
primary
reconfig-primary
```

Más información Significado de las etiquetas

Las etiquetas en la parte derecha del nombre de la configuración significan:

- `[current]` – Última configuración iniciada, sólo si coincide con la configuración actualmente en ejecución, esto es, hasta que se inicia una reconfiguración. Después de la reconfiguración, la anotación cambia a `[next poweron]`.
- `[next poweron]` – Configuración que se usará en el siguiente apagado y encendido.

▼ Enumerar dispositivos

- Enumera todos los recursos del servidor, enlazados y desenlazados.

```
primary# ldm list-devices -a
```

▼ Enumerar la memoria disponible

- Enumera la cantidad de memoria disponible para ser asignada.

```
primary# ldm list-devices mem
MEMORY
  PA                SIZE
  0x14e000000      2848M
```

▼ Enumerar los servicios

- Enumera los servicios que están disponibles.

```
primary# ldm list-services
```

Enumeración de restricciones

Para el Administrador de Dominios lógicos, las restricciones son uno o varios recursos que desea asignar a un dominio específico. O bien recibe todos los recursos que ha solicitado que se agreguen a un dominio o ninguno de éstos, dependiendo de los recursos disponibles. El subcomando `list-constraints` enumera los recursos que ha solicitado que se asignen al dominio.

▼ Enumerar restricciones para un dominio

- Enumera las restricciones para un dominio.

```
primary# ldm list-constraints ldom
```

▼ Enumerar restricciones en formato XML

- Enumera las restricciones en formato XML para un determinado dominio.

```
primary# ldm list-constraints -x ldom
```

▼ Enumera las restricciones en un formato informatizado

- Enumera las restricciones para todos los dominios en formato analizable.

```
primary# ldm list-constraints -p
```


Administración de las configuraciones

Este capítulo contiene información sobre la administración de las configuraciones de dominio.

Este capítulo trata sobre los siguientes temas:

- [“Operación para guardar las configuraciones del dominio para una reconstrucción en el futuro” en la página 187](#)
- [“Administración de las configuraciones Dominios lógicos” en la página 190](#)

Operación para guardar las configuraciones del dominio para una reconstrucción en el futuro

Este proceso básico es guardar la información sobre las restricciones de recursos para cada dominio en un archivo XML, que después puede volver a ser emitido por el Administrador de Dominios lógicos, por ejemplo, después de un fallo de hardware, para construir la configuración deseada.

[“Restablecimiento de la configuración de un dominio desde un archivo XML \(`ldm add-domain`\)” en la página 188](#) funciona para los dominios huésped, pero no para el dominio de control (`primary`). Puede guardar las restricciones del dominio `primary` en un archivo XML, pero no puede volver a enviar el archivo al comando `ldm add-domain -i`. En cualquier caso, puede usar el comando `ldm init-system` y las restricciones de recursos del archivo XML para reconfigurar el dominio `primary`. También puede usar el comando `ldm init-system` para reconfigurar otros dominios que se describen en el archivo XML, pero estos dominios quedan inactivos cuando se completa la configuración.

El siguiente método no conserva los enlaces actuales, sólo las restricciones usadas para crear dichos enlaces. Esto significa que, después de este procedimiento, los dominios tendrán los mismos recursos virtuales, pero no estarán necesariamente enlazados a los mismos recursos físicos.

▼ Operación para guardar las configuraciones del dominio

Este procedimiento muestra cómo guardar la configuración de un dominio para un solo dominio o para todos los dominios en un sistema.

- **Guardar la configuración de un dominio para uno o varios dominios.**

- **Para guardar la configuración de un solo dominio, cree un archivo XML que contenga todas las restricciones del dominio.**

```
# ldm list-constraints -x ldom >ldom.xml
```

El siguiente ejemplo muestra cómo crear un archivo XML, `ldg1.xml`, que contiene las restricciones del dominio `ldg1`:

```
# ldm list-constraints -x ldg1 >ldg1.xml
```

- **Para guardar las configuraciones para todos los dominios en un sistema, cree un archivo XML que contenga todas las restricciones de todos los dominios.**

```
# ldm list-constraints -x >file.xml
```

El siguiente ejemplo muestra cómo crear un archivo XML, `config.xml`, que contiene las restricciones para todos los dominios de un sistema:

```
# ldm list-constraints -x >config.xml
```

▼ Restablecimiento de la configuración de un dominio desde un archivo XML (`ldm add-domain`)

En vez de este procedimiento, puede usar el comando `ldm init-system` para restaurar las configuraciones de un dominio desde un archivo XML. Véase [“Restauración de la configuración de un dominio desde un archivo XML \(`ldm init-system`\)”](#) en la página 189.

- 1 **Cree el dominio usando el archivo XML que ha creado como entrada.**

```
# ldm add-domain -i ldom.xml
```

- 2 **Enlace el dominio.**

```
# ldm bind-domain ldom
```

- 3 **Inicie el dominio.**

```
# ldm start-domain ldom
```

Ejemplo 11-1 Restablecimiento de un solo dominio desde un archivo XML

El siguiente ejemplo muestra cómo restaurar un solo dominio. Primero, restaure el dominio ldg1 desde el archivo XML. Después, enlace y reinicie el dominio ldg1 que ha restaurado.

```
# ldm add-domain -i ldg1.xml
# ldm bind ldg1
# ldm start ldg1
```

▼ Restauración de la configuración de un dominio desde un archivo XML (ldm init-system)

Este procedimiento explica cómo usar el comando `ldm init-system` con un archivo XML para volver a crear una configuración anteriormente guardada. El archivo XML describe una o varias configuraciones de dominio. El archivo XML puede crearse ejecutando el comando `ldm ls-constraints -x`. Se espera que el comando `ldm init-system` sea ejecutado en la configuración `factory-default`, pero puede restaurar cualquier configuración desde un archivo XML. El dominio `primary` se reconfigura tal y como se especifica en el archivo, y cualquier dominio que no es `primary` que tiene configuración en el archivo XML es reconfigurado pero se deja inactivo.

En vez de este procedimiento, puede usar el comando `ldm add-domain` para restaurar la configuración de un solo dominio desde un archivo XML. Véase [“Restablecimiento de la configuración de un dominio desde un archivo XML \(ldm add-domain\)”](#) en la página 188.

- 1 **Inicie la sesión en el dominio `primary`.**
- 2 **Compruebe que el sistema esté en la configuración `factory-default`.**

```
primary# ldm list-config | grep "factory-default"
factory-default [current]
```

Si el sistema no está en la configuración `factory-default`, véase [“Restablezca la configuración predeterminada de fábrica.”](#) en la página 38.

- 3 **Conviértase en un superusuario o asuma una función equivalente.**

Las funciones contienen autorizaciones y comandos con privilegios. Para más información sobre las funciones, véase [“Configuring RBAC \(Task Map\)”](#) de *System Administration Guide: Security Services*.

- 4 **Restablezca la configuración del dominio o las configuraciones desde el archivo XML.**

```
# ldm init-system [-rs] -i filename.xml
```

La opción `-r` reinicia el dominio `primary` después de la configuración. Si no especifica la opción `-r`, debe realizar el reinicio manualmente. La opción `-s` restablece sólo la configuración de los servicios virtuales (`vds`, `vcc` y `vsw`) y puede realizarse sin que sea necesario reiniciar el ordenador.

Ejemplo 11-2 Restablecimiento de dominios desde archivos de configuración XML

Los siguientes ejemplos muestran cómo usar el comando `ldm init-system` para restaurar el dominio `primary` y todos los dominios en un sistema desde la configuración `factory-default`.

- **Restaura el dominio `primary`.** La opción `-r` se usa para reiniciar el dominio `primary` después de haber completado la configuración. El archivo `primary.xml` contiene la configuración de dominio XML que ha guardado anteriormente.

```
primary# ldm init-system -r -i primary.xml
```

- **Restauración de todos los dominios en un sistema.** Restaura los dominios en el sistema a las configuraciones en el archivo XML `config.xml`. El archivo `config.xml` contiene las configuraciones de dominio XML que ha guardado anteriormente. El dominio `primary` es reiniciado automáticamente por el comando `ldm init-system`. Se restaura cualquier otro dominio, pero no se enlaza y reinicia.

```
# ldm init-system -r -i config.xml
```

Después del reinicio del sistema, los siguientes comandos enlazan y reinician los dominios `ldg1` y `ldg2`:

```
# ldm bind ldg1
# ldm start ldg1
# ldm bind ldg2
# ldm start ldg2
```

Administración de las configuraciones Dominios lógicos

Una *configuración* Dominios lógicos es una descripción completa de todos los dominios y las asignaciones de recursos en un solo sistema. Puede guardar y almacenar las configuraciones en el procesador de servicios (SP) para usarlas más adelante.

Cuando enciende un sistema el SP inicia la configuración seleccionada. Si se inicia una configuración, el sistema ejecuta el mismo conjunto de dominios, y usa las mismas asignaciones de recursos de virtualización y partición que se especifican en la configuración. La configuración predeterminada es la que se ha guardado más recientemente.

Desde la versión Dominios lógicos 1.2, se guarda automáticamente una copia de la configuración actual en el dominio de control cuando se cambia la configuración de Dominios lógicos.

La operación de autoguardado se realiza inmediatamente, incluso en las siguientes situaciones:

- Cuando la nueva configuración no se ha guardado explícitamente en el SP
- Cuando el cambio de la configuración actual no se realiza hasta que el dominio afectado se reinicia.

El operación de autoguardado le permite recuperar una configuración cuando las configuraciones guardadas en el SP se pierden. Esta operación también permite recuperar una

configuración cuando la configuración actual no se había guardado explícitamente en el SP cuando el sistema se ha apagado y encendido. En estas circunstancias, el Administrador de Dominios lógicos puede recuperar esa configuración en el reinicio si es más nueva que la configuración marcada para el siguiente inicio.

Nota – La administración de energía, los eventos de actualización FMA, ASR, y PRI no provocan una actualización de los archivos de autoguardado.

Puede restaurar automática o manualmente los archivos a configuraciones nuevas o existentes. De manera predeterminada, cuando una configuración de autoguardado es más nueva que la correspondiente configuración en ejecución, se escribe un mensaje en el registro Dominios lógicos. Por lo tanto, debe usar el comando `ldm add-spconfig -r` para actualizar manualmente una configuración existente o crear una nueva basada en los datos de autoguardado.

Nota – Cuando una reconfiguración retrasada está pendiente, los cambios en la configuración se autoguardan inmediatamente. Como resultado, si ejecuta el comando `ldm list-config -r`, se muestra la configuración de auto recuperación que es más nueva que la configuración actual.

Para más información sobre cómo usar los comandos `ldm *-spconfig` para administrar y recuperar manualmente los archivos de autoguardado, véase la página de comandos [man ldm\(1M\)](#).

Para más información sobre cómo seleccionar una configuración para iniciar, véase “[Uso de Dominios lógicos con el procesador de servicio](#)” en la página 200.

▼ Modificación de la normativa de auto recuperación

La normativa de autorecuperación especifica cómo administrar la recuperación de una configuración cuando una recuperación que se guarda automáticamente en el dominio de control es más nueva que la configuración correspondiente en ejecución. La normativa de autorecuperación se especifica configurando la propiedad `autorecovery_policy` del servicio SMF `ldmd`. La propiedad `autorecovery_policy` puede tener los siguientes valores:

- `autorecovery_policy=1` – Registra los mensajes de advertencia cuando una configuración de autoguardado es más nueva que la correspondiente configuración en ejecución. Estos mensajes se guardan en el archivo de registro SMF `ldmd`. El usuario debe realizar manualmente cualquier recuperación de la configuración. Esta es la normativa predeterminada.
- `autorecovery_policy=2` – Muestra un mensaje de notificación si una configuración autoguardada es más nueva que la configuración correspondiente que se está ejecutando. Este mensaje de notificación se imprime en la salida de cualquier comando `ldm` la primera vez que se emite un comando `ldm` después del reinicio del Administrador de Dominios lógicos. El usuario debe realizar manualmente cualquier recuperación de la configuración.
- `autorecovery_policy=3` – Actualiza automáticamente la configuración si una configuración autoguardada es más nueva que la configuración correspondiente que se está ejecutando. Esta acción sobrescribe la configuración SP que se usará durante el siguiente apagado y encendido. Esta configuración se actualiza con la configuración más nueva que se guarda en el dominio de control. Esta acción no tiene un impacto en la configuración que se está ejecutando actualmente. Sólo afecta a la configuración que se usará durante el siguiente apagado y encendido. También se registra un nuevo mensaje, que indica que se ha guardado la configuración más nueva en el SP y que se iniciará la próxima vez que el sistema se apague y encienda. Estos mensajes se guardan en el archivo de registro SMF `ldmd`.

1 Inicie la sesión en el dominio de control.

2 Conviértase en un superusuario o asuma una función equivalente.

Las funciones contienen autorizaciones y comandos con privilegios. Para más información sobre las funciones, véase “[Configuring RBAC \(Task Map\)](#)” de *System Administration Guide: Security Services*.

3 Véase el valor de la propiedad `autorecovery_policy`.

```
# svccfg -s ldmd listprop ldmd/autorecovery_policy
```

4 Pare el dispositivo `ldmd`.

```
# svcadm disable ldmd
```

5 Cambie el valor de la propiedad `autorecovery_policy`.

```
# svccfg -s ldmd setprop ldmd/autorecovery_policy=value
```

Por ejemplo, para fijar la normativa para realizar una auto recuperación, fije el valor de la propiedad a 3:

```
# svccfg -s ldmd setprop ldmd/autorecovery_policy=3
```

6 Actualice y reinicie el servicio ldmd.

```
# svcadm refresh ldmd  
# svcadm enable ldmd
```

Ejemplo 11-3 Modificación de la normativa de auto recuperación de archivo a auto recuperación

El siguiente ejemplo muestra cómo ver el valor actual de la propiedad `autorecovery_policy` y cambiarlo a un valor nuevo. El valor original de esta propiedad es 1, lo que significa que se registran los cambios de autoguardado. El comando `svcadm` se usa para parar y reiniciar el servicio `ldmd` y el comando `svccfg` se usa para ver y fijar el valor de la propiedad.

```
# 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
```


Realización de otras tareas administrativas

Este capítulo contiene información y tareas acerca del uso del software de Oracle VM Server for SPARC que no se describen en los anteriores capítulos.

Este capítulo trata sobre los siguientes temas:

- “Introducción de nombres en la CLI” en la página 195
- “Conexión a una consola huésped sobre una red” en la página 196
- “Uso de grupos de consolas” en la página 197
- “Paro de un dominio muy cargado que puede provocar un retraso de la conexión” en la página 198
- “Funcionamiento del SO de Solaris de Oracle con el Oracle VM Server for SPARC” en la página 198
- “Uso de Dominios lógicos con el procesador de servicio” en la página 200
- “Configuración de las dependencias de dominio” en la página 201
- “Determinación de dónde ocurren los errores por la asignación de CPU y direcciones de memoria” en la página 205
- “Uso de los identificadores únicos universales” en la página 207
- “Comando de información de dominio virtual y API” en la página 208

Introducción de nombres en la CLI

Las siguientes secciones describen las restricciones en la introducción de nombres en la CLI del Administrador de Dominios lógicos.

Nombres de archivo (*file*) y nombres de variables (*var-name*)

- El primer carácter debe ser una letra, un número o una barra diagonal (/).
- Los siguientes caracteres deben ser letras, números o puntuación.

backend del servidor de disco virtual y nombres del dispositivo del conmutador virtual

Los nombres deben contener letras, números o puntuación.

Nombre de configuración (*config-name*)

El nombre de la configuración de dominio lógico (*config-name*) que asigna a una configuración guardada en el procesador de servicios (SP) no debe tener más de 64 caracteres.

Todos los otros nombres

El resto de los nombres, como el nombre del dominio lógico (*ldom*), nombres de servicio (*vswitch-name*, *service-name*, *vdpcs-service-name* y *vcc-name*), nombre de la red virtual (*if-name*) y nombre del disco virtual (*disk-name*), deben presentar el siguiente formato:

- El primer carácter debe ser una letra o un número.
- Los siguientes caracteres deben ser letras, números o cualquier de los siguientes caracteres `-_+#. : ; ~ ()`.

Conexión a una consola huésped sobre una red

Puede conectarse a una consola huésped sobre la red si la propiedad `listen_addr` está fijada en la dirección IP del dominio de control en el manifiesto SMF `vntsd(1M)`. Por ejemplo:

```
$ telnet host-name 5001
```

Nota – La habilitación de acceso de red a una consola tiene implicaciones de seguridad. Cualquier usuario puede conectarse a una consola y por esta razón se deshabilita de manera predeterminada.

Un manifiesto de un dispositivo de administración de servicios es un archivo XML que describe un servicio. Para más información sobre la creación de un manifiesto SMF, consulte [Solaris 10 System Administrator Collection](http://docs.sun.com/app/docs/coll/47.16) (<http://docs.sun.com/app/docs/coll/47.16>).

Nota – Para acceder a un SO no en inglés en un dominio huésped a través de la consola, el terminal para la consola debe estar en la configuración regional requerida por el SO.

Uso de grupos de consolas

El daemon del servidor terminal de la red virtual, [vntsd\(1M\)](#), le permite ofrecer acceso para múltiples consolas de dominio usando un solo puerto TCP. En el momento de la creación del dominio, el Administrador de Dominios lógicos asigna un único puerto TCP a cada consola creando un nuevo grupo predeterminado para la consola de este dominio. Entonces, el puerto TCP se asigna al grupo de consolas en oposición a la consola misma. La consola puede enlazarse con un grupo existente usando el subcomando `set -vcons`.

▼ Combine múltiples consolas en un grupo

1 Enlace las consolas para los dominios en un grupo

El siguiente ejemplo muestra el enlazado de la consola para tres dominios diferentes (`ldg1` `ldg2` y `ldg3`) al mismo grupo de consola (`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 Conecte el puerto TCP asociado (`localhost` al puerto `5000` en este ejemplo).

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

Se le solicita que seleccione una de las consolas del dominio.

3 Enumere los dominios con el grupo seleccionando `l` (lista).

```
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
```

Nota – Para reasignar la consola a un grupo diferente o instancia `vcc`, el dominio debe estar desenlazado, esto es, tiene que estar en estado inactivo. Consulte la página de comandos `man` de SO 10 de Solaris de Oracle [vntsd\(1M\)](#) para más información sobre cómo configurar y usar el SMF para administrar `vntsd` y usar los grupos de consola.

Paro de un dominio muy cargado que puede provocar un retraso de la conexión

Un comando `ldm stop-domain` puede finalizar antes de que el dominio haya completado el apagado. Cuando esto sucede, el Administrador de Dominios lógicos genera un error parecido al siguiente.

```
LDom ldg8 stop notification failed
```

En cualquier caso, el dominio puede estar aun procesando la solicitud de apagado. Use el comando `ldm list-domain` para comprobar el estado del dominio. Por ejemplo:

```
# ldm list-domain ldg8
NAME      STATE  FLAGS  CONS  VCPU  MEMORY  UTIL  UPTIME
ldg8      active s---- 5000   22   3328M  0.3% 1d 14h 31m
```

La anterior lista muestra el dominio como activo, pero la etiqueta `s` indica que el dominio está en proceso de paro. Este debe ser un estado transitorio.

El siguiente ejemplo muestra el dominio que ahora ha parado.

```
# ldm list-domain ldg8
NAME      STATE  FLAGS  CONS  VCPU  MEMORY  UTIL  UPTIME
ldg8      bound  ----- 5000   22   3328M
```

Funcionamiento del SO de Solaris de Oracle con el Oracle VM Server for SPARC

Esta sección describe los cambios en el comportamiento cuando se usa el SO de Solaris de Oracle que se producen cuando se crean instancias de una configuración creada por el Administrador de Dominios lógicos.

El firmware OpenBoot no está disponible cuando se arranca el SO de Solaris de Oracle

El firmware OpenBoot no está disponible cuando se arranca el SO de Solaris de Oracle, porque se elimina de la memoria.

Para llegar al mensaje `ok` desde el SO de Solaris de Oracle, debe parar el dominio. Puede usar el comando `halt` del SO de Solaris de Oracle para parar el dominio.

Apagado y reencendido de un servidor

Siempre que realice cualquier mantenimiento en un sistema que ejecuta el software del Oracle VM Server for SPARC que requiere un apagado y encendido del servicio, debe guardar las configuraciones del dominio lógico actual en el SP antes.

▼ Guardar las configuraciones de dominio lógico en el SP

- Utilice el comando siguiente:

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

No use el comando `psradm(1M)` en CPU activas en un dominio gestionado por energía

No intente cambiar el estado de funcionamiento de una CPU en un dominio administrado por energía usando el comando `psradm(1M)`.

Resultado de las interrupciones de SO de Solaris de Oracle

El comportamiento descrito en esta sección se observa cuando realiza las siguientes operaciones:

1. Pulse la secuencia de teclas L1-A cuando el dispositivo de entrada esté fijado en teclado.
2. Introduzca el comando `send break` cuando la consola virtual esté en situación `telnet`.

Después de todas estas interrupciones, recibe el siguiente mensaje:

```
c)ontinue, s)ync, r)eset, h)alt?
```

Escriba la letra que representa lo que quiere que haga el sistema después de este tipo de interrupciones.

Resultados de detener o reiniciar el dominio de control

En la siguiente tabla se muestra el comportamiento previsto al detener o reiniciar el dominio de control (`primary`).

TABLA 12-1 Comportamiento previsto al detener o reiniciar el dominio de control (primary)

Comando	¿Otro dominio configurado?	Comportamiento
halt	No configurado	Host apagado y se mantiene apagado hasta encenderlo en el SP.
	Configurado	El software se reinicia y arranca si la variable <code>auto-boot?=true</code> . El software se reinicia y detiene en el indicador Aceptar si la variable <code>auto-boot?=false</code> .
reinicio	No configurado	Reinicia el sistema, que no se apaga.
	Configurado	Reinicia el sistema, que no se apaga.
shutdown -i 5	No configurado	Host apagado, se mantiene apagado hasta encenderlo en el SP.
	Configurado	El software se reinicia.

Para más información sobre las consecuencias del reinicio de un sistema de control que tiene la función de dominio raíz, véase [“Reinicio del dominio primary” en la página 73](#).

Uso de Dominios lógicos con el procesador de servicio

Esta sección describe la información que hay que tener en cuenta cuando se usa el procesador de servicios (SP) del Integrated Lights Out Manager (ILOM) con el Administrador de Dominios lógicos. Para más información sobre el uso del software ILOM, vea los documentos para la plataforma específica, como [Sun SPARC Enterprise T5120 and T5220 Servers Topic Set](#) para los servidores Sun SPARC Enterprise T5120 y T5220.

Hay disponible otra opción en el comando ILOM existente.

```
-> set /HOST/bootmode config=config-name
```

La opción `config=config-name` le permite fijar la configuración en el siguiente encendido a otra configuración, incluyendo la configuración de envío `factory-default`.

Puede invocar el comando tanto si el host está apagado como encendido. Se efectúa en el siguiente restablecimiento del host o cuando se enciende.

▼ Restablezca la configuración del dominio a la configuración predeterminada u otra.

- Restablezca la configuración del dominio lógico en el siguiente encendido a la configuración de envío predeterminada ejecutando este comando:

```
-> set /HOST/bootmode config=factory-default
```

También puede seleccionar otras configuraciones creadas con el Administrador de Dominios lógicos usando el comando `ldm add-config` y guardado en el procesador de servicios (SP). El nombre que especifica en el comando de Administrador de Dominios lógicos `ldm add-config` puede usarse para seleccionar esa configuración con el comando `bootmode` de ILOM. Por ejemplo, consideremos que ha guardado una configuración con el nombre `ldm-config1`.

```
-> set /HOST/bootmode config=ldm-config1
```

Ahora, debe apagar y encender el sistema para cargar la nueva configuración.

Véase la página de comando `man ldm(1M)` para más información sobre el comando `ldm add-config`.

Configuración de las dependencias de dominio

Puede usar el Administrador de Dominios lógicos para establecer las relaciones de dependencia entre dominios. Un dominio que tiene uno o varios dominios que dependen de él se llama un *dominio maestro*. Un dominio que depende de otro dominio se llama un *dominio esclavo*.

Cada dominio esclavo puede especificar hasta cuatro dominios maestros fijando la propiedad `master`. Por ejemplo, el dominio esclavo `pine` especifica los cuatro dominios maestros en la siguiente lista separada por comas:

```
# ldm add-domain master=apple,lemon,orange,peach pine
```

Cada dominio maestro puede especificar qué pasa a los dominios esclavos en caso que el dominio maestro falle. Por ejemplo, si falla un dominio maestro, puede ser necesario que los dominios esclavos generen un mensaje de error grave. Si un dominio esclavo tiene más de un dominio maestro, el primer dominio maestro que falla acciona la normativa de fallo definida en todos los dominios esclavos.

Nota – Si falla más de un dominio esclavo simultáneamente, sólo una de las normativas de fallo especificadas se forzará en todos los dominios esclavos afectados. Por ejemplo, si los dominios maestro que han fallado tiene normativas de `stop` y `panic`, todos los dominios esclavos se pararán o generarán un error crítico.

La normativa de fallos del dominio maestro se controla configurando uno de los siguientes valores en la propiedad `failure-policy`:

- `ignore` ignora cualquier dominio esclavo cuando falla el dominio maestro.
- `panic` se genera el mensaje de error grave en cualquier dominio esclavo cuando falla el dominio maestro.
- `reset` se restablece cualquier dominio esclavo cuando falla el dominio maestro.
- `stop` se para cualquier dominio esclavo cuando falla el dominio maestro.

En este ejemplo, los dominios maestros especifican la normativa de fallo de la siguiente manera:

```
# 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
```

Puede usar este mecanismo para crear dependencias explícitas entre dominios. Por ejemplo, un dominio huésped depende implícitamente del dominio de servicio para ofrecer los dispositivos virtuales. Un dominio huésped E/S se bloquea cuando el dominio de servicio del que depende no está funcionando y en ejecución. Si se define un dominio huésped como esclavo del dominio de servicio, se puede especificar el comportamiento del dominio huésped cuando se cae el dominio de servicio. Cuando no se establece esta dependencia, un dominio huésped simplemente espera a que el dominio de servicio vuelva a funcionar.

Nota – El Administrador de Dominios lógicos no le permite crear relaciones de dominio que creen un ciclo de dependencia. Para más información, véase [“Ciclos de dependencia” en la página 203](#).

Para ejemplos XML de dependencia de dominios, véase el [Ejemplo D-6](#).

Ejemplo de dependencia de dominios

Los siguientes ejemplos muestran cómo configurar dependencias de dominios.

- El primer comando crea un dominio maestro llamado `twizzle`. Este comando usa `failure-policy=reset` para especificar que los dominios esclavos se restablecen si falla el dominio `twizzle`. El segundo comando modifica un dominio maestro llamado `primary`. Este comando usa `failure-policy=panic` para especificar que los dominios esclavos generan un error crítico si el dominio `primary` falla. El tercer comando crea un dominio esclavo llamado `chocktaw` que depende de los dos dominios maestros, `twizzle` y `primary`. El dominio esclavo utiliza `master=twizzle,primary` para especificar los dominios maestros. En caso que el dominio `twizzle` o `primary` falle, el dominio `chocktaw` se restablecerá o generará un error crítico. El primer dominio maestro que falla es el que determina el comportamiento de los dominio esclavos.

```
# ldm add-domain failure-policy=reset twizzle
# ldm set-domain failure-policy=panic primary
# ldm add-domain master=twizzle,primary chocktaw
```

- Este ejemplo muestra cómo usar el comando `ldm set-domain` para modificar el dominio `orange` para asignar `primary` como dominio maestro. El segundo comando usa el comando `ldm set-domain` para asignar `orange` y `primary` como dominios maestros para el dominio `tangerine`. El tercer comando incluye la información sobre todos estos dominios.

```
# 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

```

```

VARIABLES
  test_var=Aloha

```

```

-----
NAME          STATE      FLAGS    UTIL
tangerine     bound     -

```

```

HOSTID
  0x84f948e9

```

```

CONTROL
  failure-policy=ignore

```

```

DEPENDENCY
  master=orange,primary

```

```

VARIABLES
  test_var=A hui hou

```

- A continuación se muestra un ejemplo de un listado con salida analizable:

```
# ldm list -o domain -p
```

Ciclos de dependencia

El Administrador de Dominios lógicos no le permite crear relaciones de dominio que creen un ciclo de dependencia. Un *ciclo de dependencia* es una relación entre dos o más dominios que lleva a una situación en la que un dominio esclavo depende de sí mismo, o un dominio maestro depende de sus dominios esclavos.

El Administrador de Dominios lógicos determina si existe un ciclo de dependencia antes de agregar una dependencia. El Administrador de Dominios lógicos se pone en marcha en el dominio esclavo y busca todas las rutas especificadas por la matriz del maestro hasta haber alcanzado el final de la ruta. Cualquier ciclo de dependencia detectado se indica como error.

El siguiente ejemplo muestra cómo puede crearse un ciclo de dependencia. El primer comando crea un dominio esclavo llamado `mohawk` que especifica su dominio maestro como `primary`. Así que `mohawk` depende de `primary` en la siguiente cadena de dependencia:

FIGURA 12-1 Dependencia de dominio individual



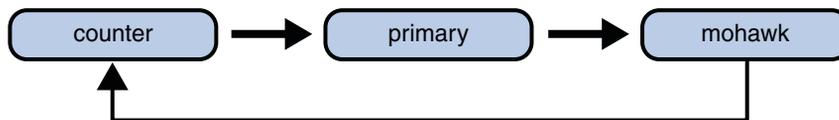
El segundo comando crea un dominio esclavo llamado `primary` que especifica su dominio maestro como `counter`. Así pues, `mohawk` depende de `primary`, que depende de `counter` en la siguiente cadena de dependencia:

FIGURA 12-2 Dependencia de dominios múltiples



El tercer comando intenta crear una dependencia entre los dominios `counter` y `mohawk`, lo que produce el siguiente ciclo de dependencia:

FIGURA 12-3 Ciclo de dependencia de dominio



El comando `ldm set -domain` fallará con el siguiente mensaje de error:

```

# 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
  
```

Determinación de dónde ocurren los errores por la asignación de CPU y direcciones de memoria

Esta sección muestra cómo puede correlacionar la información obtenida por arquitectura de administración de fallos (FMA) de Solaris de Oracle con los recursos del dominio lógico marcados como erróneos.

La FMA indica error de CPU en términos de número de CPU físicos y errores de memoria en términos de direcciones de memoria física.

Si desea determinar en qué dominio lógico se ha producido un error y el correspondiente número de la CPU virtual o dirección de memoria real en el dominio, debe realizar una asignación.

Asignación de CPU

El dominio y el número de CPU virtual en el dominio, al que corresponde el número de CPU física dado, puede determinarse con los siguientes procedimientos.

▼ Determinación del número de CPU

- 1 Genere una lista larga analizable para todos los dominios.

```
primary# ldm list -l -p
```

- 2 Busque la entrada en las secciones VCPU de la lista que tenga un campo `pid` igual al número de la CPU física.
 - Si encuentra esta entrada, la CPU está en el dominio bajo el que se enumera la entrada, y el número de CPU virtual en el dominio es dado por el campo `vid` de la entrada.
 - Si no encuentra esta entrada, la CPU no está en ningún dominio.

Asignación de memoria

El dominio y la dirección de la memoria real en el dominio, que corresponde a la dirección de la memoria física dada (PA), pueden determinarse de la siguiente manera.

▼ Determinación de la dirección de la memoria real

- 1 Genere una lista larga analizable para todos los dominios.

```
primary# ldm list -l -p
```

- 2 **Busque la línea en las secciones de la lista MEMORY donde la PA está dentro del rango inclusivo pa a $(pa + size - 1)$; esto es, $pa \leq PA < (pa + size - 1)$.**
Aquí pa y $size$ se refieren a los valores en los correspondientes campos de la línea.
 - Si encuentra esta entrada, la PA está en el dominio bajo el que se enumera y la correspondiente dirección real en el dominio es dada por $ra + (PA - pa)$.
 - Si no encuentra esta entrada, la PA no está en ningún dominio.

Ejemplos de asignación de CPU y memoria

Supongamos que tiene una configuración de dominio lógico como se muestra en el [Ejemplo 12-1](#) y desea determinar el dominio y la CPU virtual que corresponde al número de CPU física 5 y el dominio y la dirección real que corresponde a la dirección física `0x7e816000`.

Si se busca en las entradas VCPU en la lista para un campo `pid` igual a 5, puede encontrar la siguiente entrada bajo el dominio lógico `ldg1`.

```
|vid=1|pid=5|util=29|strand=100
```

Por lo tanto, el número de la CPU física 5 está en el dominio `ldg1` y dentro del dominio tiene el número de CPU virtual 1.

Si busca en las entradas de MEMORY en la lista, puede encontrar la siguiente entrada bajo el dominio `ldg2`.

```
ra=0x8000000|pa=0x7800000|size=1073741824
```

Dónde $0x78000000 \leq 0x7e816000 \leq (0x78000000 + 1073741824 - 1)$; esto es, $pa \leq PA \leq (pa + size - 1)$. Por lo tanto, la PA está en el dominio `ldg2` y la dirección real correspondiente es $0x8000000 + (0x7e816000 - 0x78000000) = 0xe816000$.

EJEMPLO 12-1 Lista larga analizable de configuraciones de Dominios lógicos

```
primary# ldm list -l -p
VERSION 1.0
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
...
```

EJEMPLO 12-1 Lista larga analizable de configuraciones de Dominios lógicos (Continuación)

```

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
...

```

Uso de los identificadores únicos universales

Desde la versión del Oracle VM Server for SPARC 2.0, a cada dominio se le asigna un identificador único universal (UUID). Se asigna el UUID cuando se crea un dominio. Para dominios de herencia, el UUID se asigna cuando el daemon `ldmd` inicializa.

Nota – El UUID se pierde si se usa el comando `ldm migrate-domain -f` para migrar un dominio a un equipo de destino que ejecuta una versión anterior del Administrador de Dominios lógicos. Cuando migra un dominio desde un equipo de origen que ejecuta una versión anterior del Administrador de Dominios lógicos, se asigna al dominio un nuevo UUID como parte de la migración. En caso contrario, el UUID migra.

Puede obtener un UUID para un dominio ejecutando los comandos `ldm list -l`, `ldm list -bindings` o `ldm list -o domain`. Los siguientes ejemplos muestran el UUID para el dominio `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 1.4
DOMAIN|name=ldg1|state=inactive|flags=|cons=|ncpu=|mem=|util=|uptime=
UUID|uuid=6c908858-12ef-e520-9eb3-f1cd3dbc3a59

```

Comando de información de dominio virtual y API

El comando `virtinfo` le habilita para obtener información sobre un dominio virtual en ejecución. También puede usar la API de información de dominio virtual para crear programas que obtienen información sobre los dominios virtuales.

La siguiente lista muestra parte de la información que puede obtener sobre un dominio virtual utilizando un comando o API:

- Tipo de dominio (implementación, control, huésped, E/S, servicio, raíz)
- Nombre de dominio determinado por el administrador de dominio virtual
- Identificador único universal (UUID) del dominio
- Nombre del nodo de red del dominio de control del dominio
- Número serial de chasis en el que se está ejecutando el dominio

Para más información sobre el comando `virtinfo`, véase la página de comandos `man virtinfo(1M)`. Para más información sobre el API, véase las páginas de comando `man libv12n(3LIB)` y `v12n(3EXT)`.

Herramienta de conversión física a virtual del Oracle VM Server for SPARC

Este apéndice trata sobre los siguientes temas:

- “Información general de la herramienta P2V del Oracle VM Server for SPARC” en la página 209
- “Dispositivos de componente posterior” en la página 212
- “Instalación de la herramienta P2V de Oracle VM Server for SPARC” en la página 213
- “Uso del comando `ldmp2v`” en la página 215

Información general de la herramienta P2V del Oracle VM Server for SPARC

La herramienta P2V del Oracle VM Server for SPARC convierte automáticamente un sistema físico existente que se ejecuta en un dominio lógico en un sistema de multiprocesamiento de chip (CMT). El sistema de origen puede ser cualquiera de los siguientes:

- Cualquier sistema sun4u basado en SPARC que ejecuta al menos el SO 8 de Solaris
- Cualquier sistema sun4v que ejecute el SO 10 de Solaris de Oracle, pero no se ejecute en un dominio lógico

La conversión de un sistema físico a un sistema virtual se realiza en las siguientes fases:

- **Fase de recogida.** Funciona sobre el sistema de origen físico. En la fase de recogida, una imagen del sistema de archivos se crea basándose en la información de configuración que recoge sobre el sistema de origen.
- **Fase de preparación.** Se ejecuta sobre el dominio de control del sistema de destino. En la fase de preparación se crea un dominio lógico en el sistema de destino basado en la información de configuración recogida en la fase de recogida. La imagen del sistema de archivos se restaura a uno o varios discos virtuales. Puede usar la herramienta P2V para crear discos virtuales en archivos planos o volúmenes ZFS. También puede crear discos virtuales en discos físicos o LUN o en volúmenes de administradores de volúmenes que haya creado. La imagen se modifica para permitir que se ejecute como un dominio lógico.
- **Fase de conversión.** Se ejecuta en el dominio de control del sistema de destino. En la fase de conversión, el dominio lógico creado se convierte en un dominio lógico que ejecuta el SO 10 de Solaris de Oracle usando el proceso de actualización estándar de Solaris.

Para más información sobre la herramienta P2V, véase la página de comandos `man ldmp2v(1M)`.

Las siguientes secciones describen cómo la conversión del sistema físico al sistema virtual se ejecuta en fases.

Fase de recogida

La fase de recogida se ejecuta en el sistema que se debe convertir. Para crear una imagen del sistema de archivos coherente, asegúrese que el sistema está suficientemente inactivo y que se ha parado todas las aplicaciones. El comando `ldmp2v` crea una copia de seguridad de todos los archivos de sistema UFS montados, asegúrese de que cualquier sistema de archivos que deba moverse a un dominio lógico esté montado. Puede excluir los sistemas de archivos montados que no desee desplazar, como los sistemas de archivos en almacenamientos SAN o sistemas de archivos que serán desplazados por otros medios. Use la opción `-x` para excluir dichos sistemas de archivos. Los sistemas de archivos excluidos con la opción `-x` no se vuelven a crear en el dominio huésped. Puede usar la opción `-0` para excluir los archivos y directorios.

No son necesarios cambios en el sistema de origen. El único requisito es la secuencia de comandos `ldmp2v` que se había instalado en el dominio de control. Asegúrese de que la utilidad `flarc` `create` está presente en el sistema de origen.

Fase de preparación

La fase de preparación usa los datos recogidos durante la fase de recogida para crear un dominio lógico que se puede comparar con el sistema de origen.

Puede usar el comando `ldmp2v` prepare de las siguientes maneras:

- **Modo automático.** Este modo crea automáticamente discos virtuales y restaura los datos del sistema de archivos.
 - Crea el dominio lógico y los discos virtuales necesarios del mismo tamaño que en el sistema de origen.
 - Efectúa una partición del disco y restaura los sistemas de archivos.
Si el tamaño combinado de los sistemas de archivos `/`, `/usr` y `/var` es inferior a 10 Gbytes, los tamaños de los sistemas de archivos se ajustan automáticamente para permitir los requisitos de más espacio de disco del SO 10 de Solaris de Oracle. Puede deshabilitarse el cambio de tamaño automático usando la opción `-x no-auto-adjust -fs` o usando la opción `-m` para cambiar el tamaño de un sistema de archivos manualmente.
 - Modifica la imagen SO del dominio lógico para sustituir todas las referencias a un hardware físico con versiones que son adecuadas para el dominio lógico. Esto le permite actualizar el sistema al SO 10 de Solaris de Oracle usando el procedimiento de actualización normal de Solaris. Las modificaciones incluyen la actualización del archivo `/etc/vfstab` que representan los nuevos nombres del disco. Cualquier disco de inicio encapsulado Solaris Volume Manager o Veritas Volume Manager (VxVM) es automáticamente desencapsulado durante este proceso. Cuando se desencapsula un disco, se convierte en segmentos simples de disco. Si VxVM está instalado en el sistema de origen, el proceso P2V deshabilita VxVM en el dominio huésped creado.
- **Modo no automático.** Debe crear discos virtuales y restaurar los datos del sistema de archivos manualmente. Este modo le permite cambiar el tamaño y el número de discos, la partición y la distribución del sistema de archivos. La fase de preparación en este modo sólo ejecuta la creación del dominio lógico y los pasos de modificación de la imagen del SO en el sistema de archivos.
- **Modo de limpieza.** Elimina un dominio lógico y todos los dispositivos de componente posterior subyacentes que se crean con `ldmp2v`.

Fase de conversión

En la fase de conversión, el dominio lógico usa el proceso de actualización de Solaris para actualizar el SO 10 de Solaris de Oracle. La opción de actualización elimina los paquetes existentes e instala los paquete de Solaris de Oracle 10 `sun4v`, que realiza automáticamente una conversión `sun4u-a-sun4v`. La fase `convert` puede usar la imagen iso del DVD de Solaris de Oracle o una imagen de instalación de red. También puede personalizar JumpStart para realizar una opción de actualización completamente automatizada.

Dispositivos de componente posterior

Puede crear discos virtuales para un dominio huésped en un número de tipos de componente posterior: archivos (`file`), volúmenes ZFS (`zvol`), discos físicos o LUN (`disk`) o volúmenes del administrador de volumen (`disk`). El comando `ldmp2v` crea automáticamente archivos o volúmenes ZFS del tamaño adecuado si especifica `file` o `zvol` como el tipo de componente posterior de una de las siguientes maneras:

- Usando la opción `-b`
- Especificando el valor del parámetro `BACKEND_TYPE` en el archivo `/etc/ldmp2v.conf`

El tipo de componente posterior `disk` le permite usar un disco físico, LUN o volumen de administrador de volúmenes (Solaris Volume Manager y Veritas Volume Manager (VxVM)) como dispositivo de componente posterior para discos virtuales. Debe crear el disco o volumen con un tamaño adecuado antes de comenzar la fase de preparación. Para un disco físico o LUN, especifique el dispositivo de componente posterior como segmento 2 del bloque o dispositivo de carácter del disco, por ejemplo `/dev/dsk/c0t3d0s2`. Para un volumen del administrador de volúmenes, especifique el dispositivo de bloque o carácter para el volumen, por ejemplo `/dev/md/dsk/d100` para Solaris Volume Manager o `/dev/vx/dsk/ldomdg/vol1` para VxVM.

A menos que especifique los nombres del volumen y del disco virtual con la opción `-B backend:volume:vdisk`, se otorgan nombres predeterminados a los volúmenes y los discos virtuales que crea para el huésped.

- `backend` especifica el nombre del componente posterior que se debe usar. Debe especificar el `backend` para el tipo componente posterior de disco. `backend` es opcional para los tipos de componente posterior `file` y `zvol` y puede ser usado para fijar un nombre no predeterminado para el archivo o volumen ZFS que `ldmp2v` crea. El nombre predeterminado es `$BACKEND_PREFIX/guest-name/diskN`.
- `volume` es opcional para todos los tipos de componente posterior y especifica el nombre del volumen del servidor del disco virtual que se debe crear para el dominio huésped. Si no se especifica, `volume` es `guest-name-volN`.
- `vdisk` es opcional para todos los tipos de componente posterior y especifica el nombre del volumen en el dominio huésped. Si no se especifica, `vdisk` es `diskN`.

Nota – Durante el proceso de conversión, el disco virtual se nombra temporalmente `guest-name-diskN` para asegurarse de que el nombre en el dominio de control es único.

Para especificar un valor en blanco para `backend`, `volume` o `vdisk` incluya sólo el separador de coma. Por ejemplo, si se especifica `-B : :vdisk001` se fija el nombre del disco virtual para `vdisk001` y se usan nombres predeterminados para el componente posterior y el volumen. Si no especifica `vdisk`, puede omitir el separador de punto y coma del final. Por ejemplo, `-B`

`/ldoms/ldom1/vol001:vol001` especifica el nombre del archivo de componente posterior como `/ldoms/ldom1/vol001` y el nombre del volumen como `vol001`. El nombre del disco virtual predeterminado es `disk0`.

Instalación de la herramienta P2V de Oracle VM Server for SPARC

El paquete de la herramienta P2V de Oracle VM Server for SPARC debe ser instalado y configurado *sólo* en el dominio de control del sistema de destino. No es necesario instalar el paquete en el sistema de origen. En cambio, puede copiar la secuencia de comandos `/usr/sbin/ldmp2v` desde el sistema de destino al sistema de origen.

Requisitos previos

Antes de ejecutar la herramienta P2V de Oracle VM Server for SPARC, asegúrese de que se cumplen las siguientes condiciones:

- Los siguientes parches de la utilidad Flash se han instalado en el sistema de origen:
 - **Para el SO 8 de Solaris:** Como mínimo parche ID 109318-34
 - **Para el SO 9 de Solaris:** Como mínimo parche ID 113343-06
- El sistema de destino ejecuta al menos Dominios lógicos 1.1 en las siguientes:
 - SO 10 10/08 de Solaris de Oracle
 - SO 10 5/08 Solaris de Oracle con los parches de Dominios lógicos 1.1 adecuados
- El dominio huésped ejecuta al menos el SO 10 5/08 de Solaris de Oracle
- El sistema de origen ejecuta al menos el SO 8 de Solaris

Además de estos requisitos previos, configure un sistema de archivos NFS que será compartido por los sistemas origen y destino. Este archivo debería poder ser escrito por `root`. En cualquier caso, si no está disponible un sistema de archivos compartido, use un sistema de archivos local que sea lo suficientemente grande para albergar el volcado de un sistema de archivos o el sistema de origen en ambos sistemas, el de origen y el de destino.

Limitaciones

La versión 2.0 de la herramienta P2V del Oracle VM Server for SPARC presenta las siguientes limitaciones:

- Sólo se admiten sistemas de archivos UFS.
- Sólo se admiten en el sistema de origen los discos sencillos (`/dev/dsk/c0t0d0s0`), los metadispositivos Solaris Volume Manager (`/dev/md/dsk/dNNN`) y los discos de inicio encapsulados VxVM.
- Durante el proceso P2V, cada dominio huésped puede tener un solo conmutador virtual y un servidor de disco virtual. Puede agregar más conmutadores virtuales y servidores de disco virtual al dominio después de la conversión P2V.
- La asistencia técnica para los volúmenes VxVM se limita a los siguientes volúmenes en un disco de inicio encapsulado: `rootvol`, `swapvol`, `usr`, `var`, `opt` y `home`. Los segmentos originales para estos volúmenes aun deben estar presentes en el disco de inicio. La herramienta P2V admite Veritas Volume Manager 5.x en el SO 10 de Solaris. En cualquier caso, también puede usar la herramienta P2V para convertir los sistemas operativos Solaris 8 y Solaris 9 que usan VxVM.
- No puede convertir sistemas Solaris 10 que están configurados con zonas.

▼ Instalación de la herramienta P2V del Oracle VM Server for SPARC

1 Vaya a la página de descarga del Oracle VM Server for SPARC en <http://www.sun.com/servers/coolthreads/ldoms/get.jsp>.

2 Descargue el paquete de software P2V, `SUNWldmp2v`.

Desde la versión 1.2 de Dominios lógicos, el paquete `SUNWldmp2v` se incluye en el archivo ZIP del Oracle VM Server for SPARC.

3 Conviértase en un superusuario o asuma una función equivalente.

Las funciones contienen autorizaciones y comandos con privilegios. Para más información sobre las funciones, véase “Configuring RBAC (Task Map)” de *System Administration Guide: Security Services*.

4 Use el comando `pkgadd` para instalar el paquete `SUNWldmp2v`.

```
# pkgadd -d . SUNWldmp2v
```

5 Cree un archivo `/etc/ldmp2v.conf` y configure las siguientes propiedades predeterminadas:

- VDS – Nombre del servicio de disco virtual, como `VDS="primary-vds0"`

- `VSW` – Nombre del conmutador virtual, como `VSW="primary-vsw0"`
- `VCC` – Nombre del concentrador de la consola virtual, como `VCC="primary-vcc0"`
- `BACKEND_TYPE` – Tipo de componente posterior del archivo `zvol`, o disco
- `BACKEND_SPARSE` – Depende si crear dispositivos de componente posterior como volúmenes o archivos dispersos `BACKEND_SPARSE="yes"`, o volúmenes o archivos no dispersos `BACKEND_SPARSE="no"`
- `BACKEND_PREFIX` – Ubicación para crear dispositivos de componente posterior de disco virtual

Cuando `BACKEND_TYPE="zvol"`, especifique el valor `BACKEND_PREFIX` como nombre del conjunto de datos de ZFS. Cuando `BACKEND_TYPE="files"`, el valor `BACKEND_PREFIX` se interpreta como el nombre de la ruta de un directorio que es relativo a `/`.

Por ejemplo, `BACKEND_PREFIX="tank/ldoms"` provocaría que ZVOL se crearan en `tank/ldoms/del conjunto de datos del nombre de dominio` y los archivos se crearan en el subdirectorio `/tank/ldoms/del nombre de dominio`.

La propiedad `BACKEND_PREFIX` no se puede aplicar al componente posterior del disco.

- `BOOT_TIMEOUT` – Tiempo de espera para el arranque del SO de Solaris de Oracle en segundos

Para más información, véase el archivo de configuración `ldmp2v.conf.sample` que es parte del conjunto que se puede descargar.

Uso del comando `ldmp2v`

Esta sección incluye ejemplos para las tres fases.

EJEMPLO A-1 Ejemplos de la fase de recogida

Los siguientes ejemplos muestran cómo puede usar el comando `ldmp2v collect`.

- **Uso compartido de sistema de archivos montado en NFS.** El siguiente ejemplo muestra la manera más fácil de realizar el paso de recogida donde los sistemas de origen y de destino comparten un sistema de archivos montados en NFS.

Como superusuario, asegúrese de que todos los sistemas de archivo UFS están montados.

```
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
```

EJEMPLO A-1 Ejemplos de la fase de recogida (Continuación)

Los siguientes ejemplos muestra cómo ejecutar la herramienta de recogida cuando los sistema de origen y de destino comparten un sistema de archivos montados en 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.
```

- **Uso no compartido de un sistema de archivos montados en NFS.** Cuando los sistemas de origen y de destino comparten un sistema de archivos montados en NFS, la imagen del sistema de archivos puede escribirse en el almacenamiento local y copiarse posteriormente al dominio de control. La utilidad Flash excluye automáticamente el archivo que crea.

```
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.
```

Copie el archivo Flash y el archivo del manifiesto desde el directorio `/var/tmp/volumia` al sistema de destino.

- **Salto del paso de copia de seguridad del sistema de archivos.** Si las copias de seguridad del sistema ya están disponibles gracias al uso de una herramienta de copia de seguridad de terceras partes como NetBackup, puede saltar el paso de la copia de seguridad del sistema de archivos usando el método de archivado `none`. Si usa esta opción, sólo se crea el manifiesto de la configuración del sistema.

```
volumia# ldmp2v collect -d /home/dana/p2v/volumia -a none
Collecting system configuration ...
The following file system(s) must be archived manually: / /u01 /var
```

Tenga en cuenta que si los sistema de origen y de destino no comparten el directorio especificado por `-d`, debe copiar los contenidos de dicho directorio en el dominio de control. Los contenidos del directorio deben copiarse al dominio de control antes de la fase de preparación.

EJEMPLO A-2 Ejemplos de la fase de preparación

Los siguientes ejemplos muestran cómo usar el comando `ldmp2v prepare`.

- El siguiente ejemplo crea un dominio lógico llamado `volumia` usando las opciones predeterminadas configuradas en `/etc/ldmp2v.conf` manteniendo al mismo tiempo las direcciones MAC del sistema físico:

```
# ldmp2v prepare -d /home/dana/p2v/volumia -o keep-mac volumia
Creating vdisks ...
Creating file systems ...
```

EJEMPLO A-2 Ejemplos de la fase de preparación (Continuación)

```

Populating file systems ...
Modifying guest domain OS image ...
Removing SVM configuration ...
Unmounting guest file systems ...
Creating domain volumia ...
Attaching vdisks to domain volumia ...

```

- El siguiente comando muestra la información sobre el dominio lógico `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

```

- Los siguientes ejemplos muestran que puede eliminar completamente un dominio y sus dispositivos de componente posterior usando la opción `-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 ...

```

- A continuación se muestra que se puede cambiar el tamaño de uno o varios sistemas de archivos durante P2V especificando el punto de enlace y el nuevo tamaño con la opción `-m`:

```

# ldmp2v prepare -d /home/dana/p2v/normaal -m /:8g normaal
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 normaal ...
Attaching vdisks to domain normaal ...

```

EJEMPLO A-3 Ejemplos de la fase de conversión

Los siguientes ejemplos muestran cómo usar el comando `ldmp2v convert`.

- Uso del servidor de la instalación de red.** El comando `ldmp2v convert` inicia el dominio sobre la red usando la interfaz de red virtual especificada. Debe ejecutar las secuencias de comandos `setup_install_server` y `add_install_client` en el servidor de instalación. Puede usar la característica personalizada de JumpStart para realizar una conversión completamente automatizada. Esta característica requiere la creación y configuración del `sysidcfg` adecuado y archivos de perfiles para el cliente en el servidor JumpStart. El perfil debe consistir en las siguientes líneas:

```
install_type    upgrade
root_device    c0d0s0
```

El archivo `sysidcfg` sólo se usa para la operación de actualización, de manera que una configuración como la siguiente debería bastar:

```
name_service=NONE
root_password=uQkoX\MLCsZhI
system_locale=C
timeserver=localhost
timezone=Europe/Amsterdam
terminal=vt100
security_policy=NONE
nfs4_domain=dynamic
network_interface=PRIMARY {netmask=255.255.255.192
                           default_route=none protocol_ipv6=no}
```

Para más información sobre un uso personalizado de JumpStart, véase *Guía de instalación de Oracle Solaris 10 9/10: Instalaciones JumpStart personalizadas y avanzadas*.

```
# 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.
```

EJEMPLO A-3 Ejemplos de la fase de conversión (Continuación)

```

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/SUNWjjet.
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...
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
[...]

```

- **Uso de una imagen ISO.** El comando `ldmp2v` convert adjunta una imagen ISO en el DVD de Solaris de Oracle en el dominio lógico y arranca desde allí. Para actualizar, conteste todas las solicitudes de `sysid` y seleccione Actualizar.

Nota – Las respuestas a las preguntas `sysid` se usan *sólo* durante la duración del proceso de actualización. Estos datos no se aplican a la imagen de SO existente en el disco. La manera más rápida y fácil de ejecutar la conversión es seleccionar Sin conexión a red. No es necesario que la contraseña `root` coincida con la contraseña `root` del sistema de origen. La identidad original del sistema está preservada por la actualización y se efectúa después del inicio posterior a la actualización. El tiempo necesario para realizar la actualización depende del Clúster de Solaris de Oracle que está instalado en el sistema original.

EJEMPLO A-3 Ejemplos de la fase de conversión (Continuación)

```
# ldmp2v convert -i /tank/iso/s10s_u5.iso -d /home/dana/p2v/volumia volumia
Testing original system status ...
LDom volumia started
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.

EJEMPLO A-3 Ejemplos de la fase de conversión (Continuación)

F2_Upgrade F3_Go Back F4_Initial F5_Exit F6_Help

Asistente para la configuración de Oracle VM Server for SPARC

El asistente para la configuración de Oracle VM Server for SPARC le guiará a través de la configuración del dominio lógico configurando las propiedades básicas. Se ejecuta en sistema de multiprocesamiento de chip (CMT).

Después de obtener los datos de configuración, el asistente para la configuración crea una configuración que es adecuada para el inicio como dominio lógico. También puede usar los valores predeterminados seleccionados por el asistente para la configuración para crear una configuración del sistema utilizable.

El asistente para configuración está disponible tanto como interfaz gráfica de usuario (GUI) que como herramienta basada en terminal, `ldmconfig`.

Para obtener más información sobre la herramienta basada en terminal, véase [“Uso del asistente para la configuración \(`ldmconfig`\)” en la página 224](#) y la páginas de comandos `man ldmconfig(1M)`.

Para más información sobre cómo iniciar la herramienta GUI, véase [“Uso del asistente para la configuración \(GUI\)” en la página 223](#).

Uso del asistente para la configuración (GUI)

El asistente para la configuración GUI se entrega como parte del paquete zip del Oracle VM Server for SPARC.

Asegúrese de que sistema de destino ejecuta como mínimo el software Dominios lógicos 1.2 y que su sistema ejecuta al menos la versión 1.6 del Java SE Runtime Environment.

Para ejecutar la GUI del asistente para la configuración desde la línea de comandos, escriba lo siguiente:

```
$ java -jar "Configurator.jar"
```

Esta herramienta de GUI incluye documentación en pantalla para ayudarle a crear la configuración para el sistema.

Uso del asistente para la configuración (`ldmconfig`)

El asistente para la configuración basado en terminal, `ldmconfig`, trabaja a través de una serie de operaciones que corresponden a las pantallas de la interfaz de usuario. El resultado final es la creación de una configuración que puede implementarse en un dominio lógico.

La siguiente sección describe cómo instalar el comando `ldmconfig` y algunas características de la herramienta del asistente para la configuración.

Instalación del asistente para la configuración

El asistente para la configuración se entrega como parte del paquete `SUNWldm`.

Después de haber instalado el paquete `SUNWldm`, puede encontrar el comando `ldmconfig` en el directorio `/usr/sbin`. El comando también se instala en el directorio `/opt/SUNWldm/bin` para la administración de equipos de herencia.

Requisitos previos

Antes de instalar y ejecutar el asistente para la configuración, asegúrese de que se cumplen las siguientes condiciones:

- El sistema de destino debe estar ejecutando al menos el programa Dominios lógicos 1.2.
- La ventana del terminal debe tener un ancho de al menos 80 caracteres y una longitud de 24 líneas.

Limitaciones y problemas conocidos

El asistente para la configuración presenta las siguientes limitaciones:

- Si se cambia el tamaño del terminal mientras se usa `ldmconfig` se pueden producir resultados ilegibles
- Sólo admiten los archivos de disco UFS como discos virtuales
- Sólo funciona con sistemas donde no están presentes configuraciones de dominios lógicos existentes
- Los puertos concentradores de la consola virtual son de 5000 a 5100
- Se usan nombres predeterminados para dominios huésped, servicios y dispositivos, que no pueden cambiarse

ldmconfig: Características

El asistente para la configuración basado en terminal, `ldmconfig`, trabaja a través de una serie de operaciones que corresponden a las pantallas de la interfaz de usuario. Puede navegar hacia atrás (anterior) y adelante (siguiente) a través de las pantallas hasta llegar al paso final. El paso final genera la configuración. Puede salir en cualquier momento del asistente para la configuración o restablecer la configuración para usar los valores predeterminados. Desde la pantalla final, puede implementar la configuración a un dominio lógico.

Primero, el asistente para la configuración revisa automáticamente el sistema para determinar los valores propietarios predeterminados más adecuados basándose en las mejores prácticas, y después muestra las propiedades que son necesarias para controlar una implementación. Tenga en cuenta que ésta no es una lista exhaustiva. Puede ver otras propiedades para personalizar ulteriormente la configuración.

Para más información sobre cómo usar la herramienta `ldmconfig`, véase la página de comandos `man ldmconfig(1M)`.

Puede ajustar las siguientes propiedades:

- **Número de dominios huésped.** Especifica el número de dominios huésped que debe crear la aplicación. El mínimo es un dominio huésped. El valor máximo está determinado por la disponibilidad de recursos VCPU. Por ejemplo, puede crear hasta 60 dominios huésped con un solo subproceso cada uno en un sistema CMT de 64 subprocesos, y reservar cuatro subprocesos para el dominio de control. Si se seleccionan las mejores prácticas, el número mínimo de recursos VCPU por dominio huésped es un solo núcleo. Así pues, en un sistema de 8 núcleos, 8 subprocesos por núcleo con mejores prácticas seleccionadas, puede crear hasta siete dominios huésped con un núcleo cada uno. Asimismo, se asigna un núcleo al dominio de control.

El asistente para la configuración muestra el número máximo de dominios que pueden ser configurados para este sistema.

El asistente para la configuración realiza las siguientes tareas para crear dominios:

- **Para todos los dominios.**
 - Crea un servicio terminal virtual en los puertos de 5000 a 5100
 - Crea un servicio de disco virtual.
 - Crea un conmutador de red virtual en el adaptador de red nombrado.
 - Habilita el daemon del servidor terminal virtual.
- **Para cada dominio.**
 - Crea el dominio lógico
 - Configura las VCPU asignadas al dominio
 - Configura la memoria asignada al dominio
 - Crea un archivo de disco UFS para usar como disco virtual

- Crea un dispositivo de servidor de disco virtual (vdsdev) para el archivo de disco
 - Asigna el archivo de disco como disco virtual vdisk0 para el dominio
 - Agrega un adaptador de red virtual al interruptor virtual en el adaptador de red designado
 - Fija la propiedad OBP auto-boot?=true
 - Fija la propiedad OBP boot-device=vdisk0
 - Enlaza el dominio
 - Ejecuta el dominio
- **Red predeterminada.** Especifique el adaptador de red que usarán los nuevos dominios para las funciones de redes virtuales. El adaptador debe estar presente en el sistema. El asistente para la configuración marca los adaptadores que están actualmente en uso por el sistema como adaptadores predeterminados, y los que tienen un estado de enlace activo (adaptadores cableados).
- **Tamaño del disco virtual.** Crea discos virtuales para cada uno de los nuevos dominios. Estos discos virtuales se crean basándose en los archivos de disco ubicados en los sistemas de archivo local. Esta propiedad controla el tamaño de cada disco virtual en Gbytes. El tamaño mínimo, 8 Gbytes, se basa en el tamaño aproximado necesario para contener un SO 10 de Solaris de Oracle, y el tamaño máximo es 100 Gbytes.
- Si el asistente para la configuración no puede encontrar los sistemas de archivos que tienen un espacio adecuado para contener los archivos de discos para todos los dominios, se muestra una pantalla de error. En este caso, puede ser necesario realizar estos pasos antes de volver a ejecutar la aplicación:
- Reduce el tamaño de los discos virtuales.
 - Reduce el número de dominios.
 - Agrega más sistemas de archivos con una mayor capacidad.
- **Directorio de disco virtual.** Especifica un sistema de archivos que tiene suficiente capacidad en el que guardar los archivos que se deben crear como discos virtuales para los nuevos dominios. El directorio se basa en un número de dominios que han sido seleccionados y el tamaño de los discos virtuales. El valor debe recalcularse y deben seleccionarse directorios de destino si cambian estos valores de propiedad. El asistente para la configuración le ofrece una lista de sistemas de archivos que tienen suficiente espacio. Después de especificar el nombre del sistema de archivos, el asistente para la configuración crea un directorio en este sistema de archivos llamado /ldoms/disks en el que crear las imágenes del disco.
- **Mejores prácticas.** Especifica si usar las mejores prácticas para los valores de propiedad.
- Si el valor es sí, el asistente para la configuración usa las mejores prácticas para varios valores de propiedad de la configuración. Fuerza el mínimo de un núcleo por dominio, incluyendo los dominios de sistema. Como resultado, esto limita el número máximo de dominios huésped al número total de núcleos presente en el sistema menos un núcleo

para los dominios de sistema. Por ejemplo, en caso de un SPARC Enterprise T5140 con dos puntos de conexión con ocho núcleos cada uno, el número máximo de dominios huésped es 15 más el dominio de sistema.

- Si el valor es no, el asistente para la configuración permite la creación de dominios que tienen un mínimo de un subproceso, pero mantiene al menos cuatro subprocesos para el dominio de sistema.

Después, el asistente para la configuración resume la configuración de implementación que se debe crear, que incluye la siguiente información:

- Número de dominios
- CPU asignada a cada dominio huésped
- Memoria asignada a cada dominio huésped
- Tamaño y ubicación de los discos virtuales
- Adaptador de red que debe usarse para los servicios de redes virtuales para los dominios huésped.
- Cantidad de CPU y memoria que debe ser usada por el sistema para servicios
- Si se identifica un DVD de SO de Solaris de Oracle válido, se usará para crear un dispositivo CD-ROM virtual compartido para permitir que los dominios huésped instalen el SO de Solaris de Oracle

Finalmente, el asistente para la configuración configura el sistema para crear la implementación especificada de Dominios lógicos. También describe las acciones que deben realizarse y muestra los comandos que deben ejecutarse para configurar el sistema. Esta información le puede ayudar para saber cómo usar los comandos `ldm` necesarios para configurar el sistema.



Precaución – *No interactúe con este paso de configuración y no interrumpa este proceso ya que podría provocar una configuración parcial del sistema.*

Después de haber completado correctamente los comandos, reinicie el sistema para que se efectúen los cambios.

Descubrimiento del Administrador de Dominios lógicos

Administrador de Dominios lógicos pueden ser descubiertos en una subred usando mensajes multidifusión. El daemon `ldmd` puede escuchar en una red para un paquete multidifusión específico. Si el mensaje multidifusión es de un determinado tipo, `ldmd` responde al llamador. Esto permite que `ldmd` sea descubierto en sistemas que ejecutan el Oracle VM Server for SPARC.

Este apéndice ofrece información sobre el descubrimiento del Administrador de Dominios lógicos en ejecución en sistemas en una subred.

Descubrimiento de sistemas que ejecutan el Administrador de Dominios lógicos

Comunicación multidifusión

El mecanismo de descubrimiento usa la misma red multidifusión usada por el daemon `ldmd` para detectar colisiones cuando se asignan direcciones MAC automáticamente. Para configurar un punto de conexión multidifusión, debe suministrar la siguiente información:

```
#define MAC_MULTI_PORT 64535
#define MAC_MULTI_GROUP "239.129.9.27"
```

De manera predeterminada, *sólo* pueden enviarse paquetes multidifusión en la subred a la que está asociado el equipo. Puede cambiar el comportamiento configurando la propiedad `SMF ldmd/hops` para el daemon `ldmd`.

Formato del mensaje

Los mensajes de descubrimiento deben marcarse claramente de manera que no se confundan con otros mensajes. El siguiente formato de los mensajes multidifusión asegura que los mensajes de descubrimiento pueden ser distinguidos por el proceso de escucha de descubrimiento:

```
#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;
```

▼ Descubrimiento del Administrador de Dominios lógicos en ejecución en la subred

1 Abra un punto de conexión multidifusión.

Asegúrese de que usa el puerto y la información de grupo especificada en “Comunicación multidifusión” en la página 229.

2 Envíe un mensaje `multicast_msg_t` por el punto de conexión.

El mensaje debe incluir los siguientes datos:

- Valor válido para `version_no`, que es 1 tal y como definido por `MAC_MULTI_VERSION`
- Valor válido para `magic_no`, que es 92792004 tal y como definido por `MAC_MULTI_MAGIC_NO`
- `msg_type` de `LDMD_DISC_SEND`

3 Escuche en el punto de conexión multidifusión para detectar respuestas de Administrador de Dominios lógicos.

Las respuestas deben ser un mensaje `multicast_msg_t` con las siguientes características:

- Valor válido para `version_no`
- Valor válido para `magic_no`
- `msg_type` establecido a `LDMD_DISC_RESP`
- La carga debe consistir en una estructura `ldmd_discovery_t`, que contenga la siguiente información:
 - `ldmd_version` – Versión del Administrador de Dominios lógicos que se ejecuta en el sistema
 - `hostname` – Nombre del host del sistema
 - `ip_address` – Dirección IP del sistema
 - `port_no` – Número de puerto usado por el Administrador de Dominios lógicos para las comunicaciones, que debe ser el puerto XMPP 6482

Cuando escucha para recibir una respuesta del Administrador de Dominios lógicos, asegúrese de que no se tienen en cuenta los paquetes de detección de colisión de MAC de asignación automática.

Uso de la interfaz XML con el Administrador de Dominios lógicos

Este capítulo explica el mecanismo de comunicación del lenguaje de marcas extensible (XML) a través del cual programas para el usuario externos pueden comunicarse mediante interfaz con el software del Oracle VM Server for SPARC. Se tratan estos temas básicos:

- “Transporte de XML” en la página 233
- “Protocolo XML” en la página 234
- “Mensajes de eventos” en la página 239
- “Acciones de Administrador de Dominios lógicos” en la página 243
- “Recursos y propiedades de Administrador de Dominios lógicos” en la página 244

Para los varios esquemas que se pueden usar con el Administrador de Dominios lógicos, véase [Apéndice E, “Esquemas XML de Administrador de Dominios lógicos”](#).

Transporte de XML

Los programas externos pueden usar el protocolo extensible de mensajería y comunicación de presencia (XMPP – RFC 3920) para comunicar con el Administrador de Dominios lógicos. El XMPP se admite para las conexiones locales y remotas y está activado de forma predeterminada. Para apagar una conexión remota, configure la propiedad de SMF `ldmd/xmpp_enabled` en `false` y reinicie el Administrador de Dominios lógicos.

```
# svccfg -s ldmd/ldmd setprop ldmd/xmpp_enabled=false
# svcadm refresh ldmd
# svcadm restart ldmd
```

Nota – La deshabilitación del servidor XMPP también evita la migración de dominio y la reconfiguración dinámica de memoria.

Servidor XMPP

El Administrador de Dominios lógicos implementa un servidor XMPP que puede comunicar con numerosas aplicaciones y bibliotecas de cliente XMPP disponibles. El Administrador de Dominios lógicos usa los siguientes mecanismos de seguridad:

- La seguridad de capa de transporte (TLS) para asegurar el canal de comunicación entre el cliente y el mismo.
- Autenticación simple y capa de seguridad (SASL) para la autenticación. PLAIN es el único mecanismo SASL admitido. Debe enviar un nombre de usuario y contraseña al servidor, de manera que le autorice antes de permitir las operaciones de seguimiento o administración.

Conexiones locales

El Administrador de Dominios lógicos detecta si los clientes usuarios están en ejecución en el mismo dominio que él mismo y si es así, realiza un protocolo de enlace XMPP mínimo con el cliente. Específicamente, el paso de autenticación SASL después de la configuración de un canal seguro a través de TLS se omite. La autenticación y la autorización se realizan según las credenciales del proceso que implementa la interfaz del cliente.

Los clientes pueden elegir si implementar un cliente XMPP completo o simplemente ejecutar un analizador XML de transmisión, como el Simple API `libxml2` para analizador XML (SAX). En cualquier caso el cliente tiene que administrar el protocolo de enlace XMPP hasta el punto de la negociación TLS. Consulte la especificación XMPP para conocer la secuencia necesaria.

Protocolo XML

Después de completar la inicialización de la comunicación, los mensajes definidos en XML de Dominios lógicos se envían a continuación. Existen dos tipos generales de mensajes XML:

- Solicitud y respuesta de mensajes, utilice la etiqueta `<LDM_interface>`. Este tipo de mensaje XML se usa para los comandos de comunicación y obtener resultados del Administrador de Dominios lógicos, análogo a los comandos de ejecución usando la interfaz de línea de comandos (CLI). Esta etiqueta también se usa para el registro y anulación de registro de eventos.
- Los mensajes de evento usan la etiqueta `<LDM_event>`. Este tipo de mensaje XML se usa para informar de manera asincrónica de los eventos publicados por el Administrador de Dominios lógicos.

Mensajes de solicitud y respuesta

La interfaz XML en el Dominios lógicos tiene dos formatos diferentes:

- Un formato para enviar comandos al Administrador de Dominios lógicos
- Otro formato para el Administrador de Dominios lógicos para responder sobre el estado del mensaje entrante y las acciones necesarias para ese mensaje.

Los dos formatos comparten muchas estructuras XML comunes, pero están separados en esta discusión para entender mejor las diferencias entre ellos. Este documento también contiene un esquema XML que detalla el XML entrante y saliente combinado (Véase “[LDM_Event \(Esquema XML\)](#)” en la página 259).

Mensajes de solicitud

Una solicitud de XML entrante al Administrador de Dominios lógicos en el nivel más básico incluye una descripción de un solo comando, operando en un solo objeto. Las solicitudes más complicadas pueden manejar múltiples comandos y múltiples objetos por comando. A continuación se muestra la estructura de un comando XML básico.

EJEMPLO D-1 Formato de un solo comando operando en un solo objeto

```
<LDM_interface version="1.0">
  <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>
```

La etiqueta<LDM_interface>

Todos los comandos enviados al Administrador de Dominios lógicos deben empezar por la etiqueta <LDM_interface>. Cualquier documento enviado al Administrador de Dominios

lógicos debe tener sólo una etiqueta `<LDM_interface>` contenida en el mismo. La etiqueta `<LDM_interface>` debe incluir un atributo de versión tal y como se muestra en el [Ejemplo D-1](#).

La etiqueta `<cmd>`

En la etiqueta `<LDM_interface>`, el documento debe contener al menos una etiqueta `<cmd>`. Cada sección `<cmd>` debe tener sólo una etiqueta `<action>`. Use la etiqueta `<action>` para describir qué comando ejecutar. Cada etiqueta `<cmd>` debe incluir al menos una etiqueta `<data>` para describir los objetos en los que debe operar el comando.

La etiqueta `<cmd>` también puede tener una etiqueta `<option>`, que se usa para las opciones y etiquetas que están asociadas con algunos comandos. Los siguientes comandos usan las opciones:

- El comando `remove-domain` puede usar la opción `-a`.
- El comando `stop-domain` puede usar la opción `-f`.
- El comando `cancel-operation` puede usar la opción `migration` o `reconf`.
- El comando `add-spconfig` puede usar la opción `-r autosave-name`.
- El comando `remove-spconfig` puede usar la opción `-r`.
- El comando `list-spconfig` puede usar la opción `-r [autosave-name]`.

La etiqueta `<data>`

Cada sección `<data>` contiene una descripción de un objeto pertinente al comando especificado. El formato de la sección de datos se basa en la porción del esquema XML del borrador de especificación del formato abierto de virtualización (OVF). Este esquema define una sección `<Envelope>` que contiene una etiqueta `<References>` (no usada por Dominios lógicos) y secciones `<Content>` y `<Section>`.

Para Dominios lógicos, la sección `<Content>` se usa para identificar y describir un dominio especial. El nombre de dominio en el `id=` attribute del nodo `<Content>` identifica el dominio. En la sección `<Content>` hay una o varias secciones `<Section>` que describen los recursos del dominio según lo necesita un comando específico.

Si sólo necesita identificar un nombre de dominio, no necesita usar las etiquetas `<Section>`. Por el contrario, si no se necesita ningún identificador de dominio para el comando, debe incluir una sección `<Section>`, que describa los recursos necesarios para el comando, fuera de la sección `<Content>`, pero dentro de la sección `<Envelope>`.

Una sección `<data>` no necesita contener una etiqueta `<Envelope>` en casos donde la información del objeto puede deducirse. Esta situación afecta sobre todo al seguimiento de todos los objetos aplicables a una acción, y a las solicitudes de registro y eliminación del registro de los eventos.

Para permitir el uso del esquema de especificación OVF para definir correctamente todos los tipos de objetos, se han definido dos OVF adicionales:

- <gprop:GenericProperty> (Véase “El esquema XML GenericProperty” en la página 276.)
- <Binding> (Véase “Binding_Type (Esquema XML)” en la página 276.)

La etiqueta <gprop:GenericProperty> se ha definido para manejar cualquier propiedad del objeto para la que la especificación OVF no tiene una definición. El nombre de la propiedad se define en el atributo key= del nodo y el valor de la propiedad son los contenidos del nodo. La etiqueta <binding> se usa en la salida del subcomando list-bindings para definir los recursos que están enlazados a otros recursos.

Mensajes de respuesta

Una respuesta XML saliente corresponde estrechamente con la estructura de solicitud entrante en términos de los comandos y objetos incluidos, con adición de una sección <Response> para cada objeto y comando especificado, así como una sección general <Response> para la solicitud. Las secciones <Response> ofrecen información sobre el estado y el mensaje como se describe en el [Ejemplo D-2](#). A continuación se incluye la estructura de una respuesta a una solicitud XML básica.

EJEMPLO D-2 Formato de una respuesta a un comando único operando en un objeto único

```
<LDM_interface version="1.0">
  <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>
```

EJEMPLO D-2 Formato de una respuesta a un comando único operando en un objeto único
(Continuación)

```

    <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>
</response>
</LDM_interface>

```

Respuesta general

Esta sección <response>, que es el descendiente directo de la sección <LDM_interface>, indica el éxito o fallo general de toda la solicitud. A menos que el documento XML esté mal formado, la sección <response> incluye sólo una etiqueta <status>. Si este estado de respuesta indica un resultado correcto, todos los comandos en todos los objetos se han efectuado correctamente. Si este estado de respuesta es un fallo y no hay etiqueta <resp_msg>, entonces uno de los comandos incluidos en la solicitud original falla. La etiqueta <resp_msg> se usa sólo para describir algún problema con el mismo documento XML.

Respuesta de comando

La sección <response> bajo la sección <cmd> alerta al usuario del éxito o fallo de este comando particular. La etiqueta <status> muestra si ese comando es correcto o falla. Como con la respuesta general, si el comando falla, la sección <response> incluye sólo una etiqueta <resp_msg> si los contenidos de la sección <cmd> de la solicitud está mal formada. En caso contrario, el estado de fallo significa que uno de los objetos contra el que se ha ejecutado el comando ha provocado un fallo.

Respuesta de objeto

Finalmente, cada sección <data> en la sección <cmd> también tiene una sección <response>. Este muestra si el comando que se ejecuta en este objeto específico es satisfactorio o falla. Si el estado de la respuesta es SUCCESS, no hay etiqueta <resp_msg> en la sección <response>. Si el estado es FAILURE, hay una o más etiquetas <resp_msg> en el campo <response>, dependiendo de los errores detectados cuando se ha ejecutado el comando contra ese objeto. Los errores de objeto pueden derivar de problemas detectados cuando se ha ejecutado el comando, o el objeto está mal formado o es desconocido.

Además de la sección <response>, la sección <data> puede contener otra información. Esta información está en el mismo formato que el campo entrante <data>, que describe el objeto que ha provocado el fallo. Véase “[La etiqueta <data>](#)” en la página 236. Esta información adicional es especialmente útil en los siguientes casos:

- Cuando un comando falla contra una sección especial <data> pero pasa cualquier sección adicional <data>
- Cuando una sección <data> vacía se pasa en un comando y falla para algunos comandos pero pasa para otros

Mensajes de eventos

En lugar de esperar respuesta, puede suscribirse para recibir notificaciones de eventos de determinados cambios de estado que suceden. Hay tres tipos de eventos a los que puede suscribirse, individual o colectivamente. Véase “[Tipos de eventos](#)” en la página 240 para conocer todos los detalles.

Registro y anulación de registro

Use un mensaje <LDM_interface> para registrarse para eventos. Véase la etiqueta “[La etiqueta <LDM_interface>](#)” en la página 235. La etiqueta de acción detalla el tipo de evento para el que desea registrarse o eliminar del registro y la sección <data> se deja vacía.

EJEMPLO D-3 Ejemplo de mensaje de solicitud de registro de evento

```
<LDM_interface version="1.0">
  <cmd>
    <action>reg-domain-events</action>
    <data version="3.0"/>
  </cmd>
</LDM_interface>
```

El Administrador de Dominios lógicos responde con un mensaje de respuesta <LDM_interface> que indica si el registro o la eliminación de registro ha tenido un resultado satisfactorio.

EJEMPLO D-4 Ejemplo de mensaje de respuesta de registro de evento

```
<LDM_interface version="1.0">
  <cmd>
    <action>reg-domain-events</action>
    <data version="3.0"/>
    <response>
      <status>success</status>
    </response>
  </data>
```

EJEMPLO D-4 Ejemplo de mensaje de respuesta de registro de evento (Continuación)

```

    <response>
      <status>success</status>
    </response>
  </cmd>
  <response>
    <status>success</status>
  </response>
</LDM_interface>

```

La cadena de acción para cada tipo de evento se enumera en la subsección de eventos.

Los mensajes <LDM_event>

Los mensajes de evento tienen el mismo formato que un mensaje entrante <LDM_interface> con la excepción que la etiqueta de inicio para el mensaje es <LDM_event>. La etiqueta de acción del mensaje es la acción que ha sido realizada para accionar el evento. La sección de datos del mensaje describe el objeto asociado con el evento; los detalles dependen del tipo de evento que se ha producido.

EJEMPLO D-5 Ejemplo, notificación <LDM_event>

```

<LDM_event version='1.0'>
  <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>

```

Tipos de eventos

A continuación se incluyen los tipos de eventos a los que se puede suscribir:

- Eventos de dominio
- Eventos de hardware
- Eventos de progreso

- Eventos de recursos

Todos los eventos corresponden a los subcomandos `ldm`.

Eventos de dominio

Los eventos de dominio describen qué acciones pueden ser realizadas directamente en un dominio. La siguiente tabla muestra eventos del dominio que pueden ser enumerados en la etiqueta `<action>` en el mensaje `<LDM_event>`.

Eventos de dominio	Eventos de dominio	Eventos de dominio
<code>add-domain</code>	<code>remove-domain</code>	<code>bind-domain</code>
<code>unbind-domain</code>	<code>start-domain</code>	<code>stop-domain</code>
<code>domain-reset</code>	<code>panic-domain</code>	<code>migrate-domain</code>

Estos eventos siempre contienen *sólo* una etiqueta `<Content>` en la sección de datos OVF que describe en qué dominio se ha producido el evento. Para registrar para los eventos del dominio, envíe un mensaje `<LDM_interface>` con la etiqueta `<action>` fijada en **reg-domain-events**. La eliminación del registro para estos eventos requiere un mensaje `<LDM_interface>` con la etiqueta de acción fijada en **unreg-domain-events**.

Eventos de hardware

Los eventos de hardware pertenecen al cambio del hardware del sistema físico. En el caso de software de Oracle VM Server for SPARC, los únicos cambios de hardware que pueden realizarse son los del procesador de servicios (SP) cuando un usuario agrega, elimina o fija la configuración de un SP. Actualmente, los únicos tres eventos para este tipo son:

- `add-sponfig`
- `set-sponfig`
- `remove-sponfig`

Los eventos de hardware contienen *sólo* una etiqueta `<Section>` en la sección de datos OVF que describe qué configuración SP a qué evento está sucediendo. Para registrarse para estos eventos, envíe un mensaje `<LDM_interface>` con la etiqueta `<action>` fijada en **reg-hardware-events**. La eliminación de un registro para estos eventos requiere un mensaje `<LDM_interface>` con la etiqueta `<action>` fijada en **unreg-hardware-events**.

Eventos de progreso

Los eventos de progreso se expiden para comandos de ejecución larga, como una migración de dominio. Estos eventos indican la cantidad de progreso que se ha realizado durante la vida del comando. En este momento, sólo se indica el evento `migration-process`.

Los eventos de progreso siempre contienen *sólo* una etiqueta <Section> en la sección de datos OVF que describe la configuración SP afectada por el evento. Para registrarse para estos eventos, envíe un mensaje <LDM_interface> con la etiqueta <action> fijada en reg-hardware-events. La eliminación de un registro para estos eventos requiere un mensaje <LDM_interface> con la etiqueta <action> fijada en unreg-hardware-events.

La sección <data> de un evento de progreso consiste en una sección <content> que describe el dominio afectado. Esta sección <content> usa una etiqueta ldom_info <Section> para actualizar el progreso. Las siguientes propiedades genéricas se muestran en la sección ldom_info:

- --progress – Porcentaje del progreso realizado por el comando
- --status – Estado del comando, que puede ser continuo, error o realizado
- --source – Equipo que está informando del progreso

Eventos de recursos

Los eventos de recursos se producen cuando los recursos se agregan, se eliminan, o cambian en cualquier dominio. La sección de datos para algunos de estos eventos contiene la etiqueta <Content> con una etiqueta <Section> que da el nombre del servicio en la sección de datos OVF. La siguiente tabla muestra eventos que pueden ser enumerados en la etiqueta <action> en el mensaje <LDM_event>.

Eventos de recursos	Eventos de recursos
add-vdiskserverdevice	remove-vdiskserverdevice
set-vdiskserverdevice	remove-vdiskserver
set-vconscon	remove-vconscon
set-vswitch	remove-vswitch
remove-udpccs	

Los eventos de recursos restantes siempre contienen *sólo* la etiqueta <Content> en la sección de datos OVF que describe en qué dominio se ha producido el evento.

Eventos de recursos	Eventos de recursos	Eventos de recursos
add-vcpu	add-crypto	add-memory
add-io	add-variable	add-vconscon
add-vdisk	add-vdiskserver	add-vnet
add-vswitch	add-udpccs	add-udpcc

Eventos de recursos	Eventos de recursos	Eventos de recursos
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-udpcc

Para registrarse para los eventos de recursos, envíe un mensaje `<LDM_interface>` con la etiqueta `<action>` fijada en **reg-resource-events**. La eliminación de un registro para estos eventos requiere un mensaje `<LDM_interface>` con la etiqueta `<action>` fijada en **unreg-resource-events**.

Todos los eventos

También se puede registrar para los tres tipos de eventos sin tener que registrarse para cada uno individualmente. Para registrarse para los tres tipos de eventos simultáneamente, envíe un mensaje `<LDM_interface>` con la etiqueta `<action>` fijada en **reg-all-events**. La eliminación de un registro para estos eventos requiere un mensaje `<LDM_interface>` con la etiqueta `<action>` fijada en **unreg-all-events**.

Acciones de Administrador de Dominios lógicos

Los comandos especificados en la etiqueta `<action>`, con la excepción de los comandos `*-*-events`, corresponde a los de la interfaz de línea de comandos `ldm`. Para más detalles sobre los subcomandos `ldm`, véase la página de comando `man ldm(1M)`.

Nota – La interfaz de XML *no* admite el verbo o comando *alias* admitido por la CLI del Administrador de Dominios lógicos.

Las cadenas admitidas en la etiqueta `<action>` son las siguientes:

Acciones de Administrador de Dominios lógicos	Acciones de Administrador de Dominios lógicos	Acciones de Administrador de Dominios lógicos
list-bindings	list-services	list-constraints
list-devices	add-domain	remove-domain
list-domain	start-domain	stop-domain
bind-domain	unbind-domain	add-io

Acciones de Administrador de Dominios lógicos	Acciones de Administrador de Dominios lógicos	Acciones de Administrador de Dominios lógicos
remove-io	add-mau	set-mau
remove-mau	add-memory	set-memory
remove-memory	remove-reconf	add-spconfig
set-spconfig	remove-spconfig	list-spconfig
add-variable	set-variable	remove-variable
list-variable	add-vconscon	set-vconscon
remove-vconscon	set-vconsole	add-vcpu
set-vcpu	remove-vcpu	add-vdisk
remove-vdisk	add-vdiskserver	remove-vdiskserver
add-udpcc	remove-udpcc	add-udpccs
remove-udpccs	add-vdiskserverdevice	remove-vdiskserverdevice
add-vnet	set-vnet	remove-vnet
add-vswitch	set-vswitch	remove-vswitch
reg-domain-events	unreg-domain-events	reg-resource-events
unreg-resource-events	reg-hardware-events	unreg-hardware-events
reg-all-events	unreg-all-events	migrate-domain
cancel-operation	set-domain	

Recursos y propiedades de Administrador de Dominios lógicos

A continuación se indican los recursos del Administrador de Dominios lógicos y las propiedades que pueden definirse para cada uno de estos recursos. Los recursos y las propiedades se muestran en **negrita** en los ejemplos de XML. Estos ejemplos muestran los recursos, no la salida enlazada. La salida de restricción puede usarse para crear una entrada para las acciones del Administrador de Dominios lógicos. La excepción a esto es la salida de migración de dominio. Véase [“Migración de dominio” en la página 255](#). Cada recurso se define en una sección OVF <Section> y es especificado por una etiqueta <rasd:OtherResourceType>.

Recurso de información de dominio (ldom_info)

EJEMPLO D-6 Ejemplo, salida SML 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>
        <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="progress">45%</gprop:GenericProperty>
        <gprop:GenericProperty key="status">ongoing</gprop:GenericProperty>
        <gprop:GenericProperty key="source">dt90-319</gprop:GenericProperty>
      </Item>
    </Section>
  </Content>
</Envelope>
```

El recurso ldom_info siempre está contenido en una sección <Content>. Las siguientes propiedades en el recurso ldom_info son opcionales:

- <rasd:Address>, que especifica la dirección MAC que se debe asignar a un dominio.
- <gprop:GenericPropertykey="failure-policy">, que especifica cómo deben comportarse los dominios esclavos si el dominio maestro falla. El valor predeterminado es ignore. A continuación se incluyen los valores de propiedad válidos:
 - ignore ignora los fallos del dominio maestro (no afecta a los dominios esclavos).
 - panic se genera el mensaje de error grave en cualquier dominio esclavo cuando falla el dominio maestro.
 - reset se restablece cualquier dominio esclavo cuando falla el dominio maestro.
 - stop se para cualquier dominio esclavo cuando falla el dominio maestro.
- <gprop:GenericPropertykey="hostid">, que especifica el ID del host que debe ser asignado al dominio.
- <gprop:GenericPropertykey="master">, que especifica hasta cuatro nombres de dominio maestro separados por comas.
- <gprop:GenericPropertykey="progress">, que especifica el porcentaje de progreso realizado por el comando.
- <gprop:GenericPropertykey="source">, que especifica el equipo que informa del progreso del comando.
- <gprop:GenericPropertykey="status">, que especifica el estado del comando (realizado, fallo o continuo).

Recurso de CPU (cpu)

El equivalente de las acciones de solicitud de XML `add-vcpu`, `set-vcpu` y `remove-vcpu` se fija el valor de la etiqueta `<gpropGenericProperty key="wcore">` de la siguiente manera:

- Si se usa la opción `-c`, fije la propiedad `wcore` en el número de núcleos completos especificados.
- Si la opción `-c` *no* se usa, fije la propiedad `wcore` a `0`.

Tenga en cuenta que la propiedad de unidades de asignación, `<rasd:AllocationUnits>`, para el recurso `cpu` siempre especifica el número de CPU virtuales y no el número de núcleos.

EJEMPLO D-7 Ejemplo de XML de cpu

El siguiente ejemplo muestra la solicitud XML equivalente para el comando `ldm add-vcpu -c 1 ldg1`:

```
<?xml version="1.0"?>
<LDM_interface version="1.2"
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>
```

Un recurso `cpu` siempre está contenida en una sección `<Content>`.

Recurso de MAU (mau)

Nota – El recurso mau es cualquier unidad criptográfica admitida en un servidor admitido. Actualmente, las dos unidades criptográficas admitidas son unidad aritmética modular (MAU) y el Control Word Queue (CWQ).

EJEMPLO D-8 Ejemplo 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>
```

Un recurso mau siempre está contenido en una sección <Content>. La única propiedad es la etiqueta <rasd:AllocationUnits>, que significa el número de MAU u otras unidades criptográficas.

Recurso de memoria (memory)

EJEMPLO D-9 Ejemplo de XML de 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>
```

Un recurso de memoria siempre está contenido en una sección <Content>. La única propiedad es la etiqueta <rasd:AllocationUnits>, que significa la cantidad de memoria.

Recurso de servidor de disco virtual (vds)

EJEMPLO D-10 Ejemplo de XML de 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>
```

Un recurso de servidor de disco virtual (vds) puede estar en una sección <Content> como parte de la descripción de un dominio, o puede aparecer por sí misma en la sección <Envelope>. La única propiedad es la etiqueta <gprop:GenericProperty> con una tecla de service_name y que contiene el nombre del recurso vds que se está describiendo.

Recurso del volumen del servidor del disco virtual (vds_volume)

EJEMPLO D-11 Ejemplo 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>
```

Un recurso vds_volume puede estar en una sección <Content> contenido como parte de la descripción del dominio, o puede aparecer por sí mismo en una sección <Envelope>. Tiene que tener las etiquetas <gprop:GenericProperty> con las siguientes claves:

- vol_name – Nombre del volumen
- service_name – Nombre del servidor de disco virtual al que está enlazado el volumen

- `block_dev` – Nombre de archivo o dispositivo que se ha de asociar con este volumen

De manera opcional, un recurso `vds_volume` también puede tener las siguientes propiedades:

- `vol_opts` – Una o varias de las siguientes, separadas por comas, con una cadena: `{ro,slice,excl}`
- `mpgroup` – Nombre del grupo de ruta múltiple (conmutación por error)

Recurso de disco (disk)

EJEMPLO D-12 Ejemplo de XML de disco

```
<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>
  </Content>
</Envelope>
```

Un recurso de disco siempre está contenido en una sección `<Content>`. Tiene que tener las etiquetas `<gprop:GenericProperty>` con las siguientes claves:

- `vdisk_name` – Nombre del disco virtual
- `service_name` – Nombre del servidor de disco virtual al que está enlazado el disco virtual
- `vol_name` – Dispositivo del servicio de disco virtual al que debe asociarse este disco virtual

Opcionalmente, el recurso `disk` también puede tener la propiedad `timeout`, que es el valor de tiempo de espera en segundos para el establecimiento de una conexión entre un cliente de disco virtual (vdc) y un servidor de disco virtual (vds). Si hay múltiples rutas de disco virtual (`vdisk`), entonces el vdc puede intentar conectar a un vds diferente, y el tiempo de espera asegura que una conexión a cualquier vds se establece en la cantidad de tiempo especificada.

Recurso de conmutador virtual (vsw)

EJEMPLO D-13 Ejemplo de XML vsw

```
<Envelope>
  <References/>
  <Content xsi:type="ovf:VirtualSystem_Type" id="ldg1">
```

EJEMPLO D-13 Ejemplo de XML vsw (Continuación)

```

<Section xsi:type="ovf:VirtualHardwareSection_Type">
  <Item>
    <rasd:OtherResourceType>vsw</rasd:OtherResourceType>
    <gprop:GenericProperty key="service_name">vsw1-ldg1</gprop:GenericProperty>
    <gprop:GenericProperty key="dev_path">bge0</gprop:GenericProperty>
    <gprop:GenericProperty key="linkprop">phys-state</gprop:GenericProperty>
    <rasd:Address>00:14:4f:fc:00:01</rasd:Address>
    <gprop:GenericProperty key="mode">sc</gprop:GenericProperty>
    <gprop:GenericProperty key="pvid">12345678</gprop:GenericProperty>
    <gprop:GenericProperty key="vid">87654321</gprop:GenericProperty>
  </Item>
</Section>
</Content>
</Envelope>

```

Un recurso vsw puede estar en una sección <Content> como parte de la descripción del dominio, o puede aparecer por sí mismo en una sección <Envelope>. Tiene que tener las etiquetas <gprop:GenericProperty> con las siguientes claves:

- service_name – Nombre que debe asignarse al conmutador virtual.
- linkprop – Especifica si el dispositivo virtual debe obtener las actualizaciones de estado del enlace físico. Cuando el valor es phys-state, el dispositivo virtual obtiene las actualizaciones de estado del enlace físico. Cuando el valor está en blanco, el dispositivo virtual no obtiene las actualizaciones de estado del enlace físico. De manera predeterminada, el dispositivo virtual no obtiene las actualizaciones de estado de enlace físico.
- dev_path – Ruta del dispositivo de red que se debe asociar con este conmutador virtual

De manera opcional, el recurso vsw también puede tener las siguientes propiedades:

- <rasd:Address> – Asigna una dirección al conmutador virtual
- pvid – Identificador (ID) de la red de área local virtual (VLAN) del puerto indica la VLAN de la que la red virtual necesita ser miembro, en modo sin etiquetas.
- vid – Identificador (ID) de la red de área local virtual (VLAN) indica la VLAN de la que una red virtual y un conmutador virtual necesitan ser miembro, en modo con etiquetas.
- mode – sc para la asistencia técnica de respuesta de Clúster de Solaris de Oracle.

Recurso de red (network)

EJEMPLO D-14 Ejemplo de XML de network

```

<Envelope>
  <References/>
  <Content xsi:type="ovf:VirtualSystem_Type" id="ldg1">

```

EJEMPLO D-14 Ejemplo de XML de network (Continuación)

```

<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>
</Content>
</Envelope>

```

Un recurso de red siempre está contenido en una sección <Content>. Tiene que tener las etiquetas <gprop:GenericProperty> con las siguientes claves:

- linkprop – Especifica si el dispositivo virtual debe obtener las actualizaciones de estado del enlace físico. Cuando el valor es phys-state, el dispositivo virtual obtiene las actualizaciones de estado del enlace físico. Cuando el valor está en blanco, el dispositivo virtual no obtiene las actualizaciones de estado del enlace físico. De manera predeterminada, el dispositivo virtual no obtiene las actualizaciones de estado de enlace físico.
- vnet_name – Nombre de la red virtual (vnet)
- service_name – Nombre del conmutador virtual (vswitch) al que está enlazada esta red virtual

De manera opcional, el recurso red también puede tener las siguientes propiedades:

- <rasd:Address> – Asigna una dirección al conmutador virtual
- pvid – Identificador (ID) de la red de área local virtual (VLAN) del puerto indica la VLAN de la que la red virtual necesita ser miembro, en modo sin etiquetas.
- vid – Identificador ID) de la red de área local virtual (VLAN) indica la VLAN de la que una red virtual y un conmutador virtual necesitan ser miembro, en modo con etiquetas.
- mode – hybrid para habilitar la E/S híbrida para esa red virtual.

Recurso del concentrador de consola virtual (vcc)

EJEMPLO D-15 Ejemplo, XML de vcc

```

<Envelope>
  <References/>
  <Content xsi:type="ovf:VirtualSystem_Type" id="ldg1">
    <Section xsi:type="ovf:VirtualHardwareSection_Type">
      <Item>

```

EJEMPLO D-15 Ejemplo, XML de vcc (Continuación)

```

    <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>

```

Un recurso vcc puede estar en una sección <Content> como parte de la descripción del dominio, o puede aparecer por sí mismo en una sección <Envelope>. Puede tener etiquetas <gprop:GenericProperty> con las siguientes claves:

- service_name – Nombre que se debe asignar al servicio de concentrador de consola virtual
- min_port – Número de puerto mínimo que se debe asignar con este vcc
- max_port – Número de puerto máximo que se debe asociar con este vcc

Recurso de variable (var)

EJEMPLO D-16 Ejemplo de XML de 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>

```

Un recurso var siempre está contenido en una sección <Content>. Puede tener etiquetas <gprop:GenericProperty> con las siguientes claves:

- name – Nombre de la variable
- value – Valor de la variable

Recurso de dispositivo de E/S físico (physio_device)

EJEMPLO D-17 Ejemplo de XML de 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>
```

Un recurso physio_device siempre está contenido en una sección <Content>. La única propiedad es la etiqueta <gprop:GenericProperty> con el valor de propiedad clave name, que es el nombre del dispositivo de E/S que se describe.

Recurso de configuración SP (spconfig)

EJEMPLO D-18 Ejemplo de XML de 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>
```

Un recurso de configuración del procesador de servicio (SP) (spconfig) siempre aparece por sí mismo en una sección <Envelope>. Puede tener las etiquetas <gprop:GenericProperty> con las siguientes claves

- spconfig_name – Nombre de la configuración que debe guardarse en el SP
- spconfig_status – El estado actual de una determinada configuración SP. La propiedad se usa en la salida de un comando `ldm list -spconfig`.

Recurso del servicio de canal plano de datos virtual (vdpcs)

EJEMPLO D-19 Ejemplo de XML de vdpcs

```
<Envelope>
  <References/>
  <Content xsi:type="ovf:VirtualSystem_Type" id="ldg1">
    <Section xsi:type="ovf:VirtualHardwareSection_Type">
      <Item>
        <rasd:OtherResourceType>vdpcs</rasd:OtherResourceType>
        <gprop:GenericProperty key="service_name">dg1-vdpcs</gprop:GenericProperty>
      </Item>
    </Section>
  </Content>
</Envelope>
```

Este recurso sólo es interesante en un entorno Netra DPS. Un recurso vdpcs puede estar en una sección <Content> como parte de la descripción de un dominio, o puede aparecer por sí mismo en una sección <Envelope>. La única propiedad es la etiqueta <gprop:GenericProperty> con el valor de propiedad clave service_name, que es el nombre del recurso del servicio de canal plano de datos virtuales (vdpcs) que se están describiendo.

Recurso de cliente de canal plano de datos virtuales (vdpc)

EJEMPLO D-20 Ejemplo de XML de vdpc

```
<Envelope>
  <References/>
  <Content xsi:type="ovf:VirtualSystem_Type" id="ldg1">
    <Section xsi:type="ovf:VirtualHardwareSection_Type">
      <Item>
        <rasd:OtherResourceType>vdpc</rasd:OtherResourceType>
        <gprop:GenericProperty key="vdpc_name">vdpc</gprop:GenericProperty>
        <gprop:GenericProperty
          key="service_name">ldg1-vdpc</gprop:GenericProperty>
      </Item>
    </Section>
  </Content>
</Envelope>
```

Este recurso sólo es interesante en un entorno Netra DPS. Un recurso de cliente de canal plano de datos virtuales siempre está contenido en la sección <Content>. Puede tener etiquetas <gprop:GenericProperty> con las siguientes claves:

- vdpcc_name – Nombre del cliente del canal plano de datos virtuales (vdpcc)
- service_name – Nombre del servicio de canal plano de datos virtuales al que debe enlazarse el vdpcc

Recurso de consola (console)

EJEMPLO D-21 Ejemplo de XML de 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>
```

Un recurso de consola siempre está contenido en una sección <Content>. Puede tener etiquetas <gprop:GenericProperty> con las siguientes claves:

- port – Puerto al que se debe cambiar esta consola virtual (console)
- service_name – Servicio de concentrador de consola virtual (vcc) al que se enlaza esta consola
- group – Nombre del grupo al que enlazar esta consola

Migración de dominio

Este ejemplo muestra lo que contiene la sección <data> para un subcomando migrate-domain.

EJEMPLO D-22 Ejemplo migrate-domain Sección <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>
```

EJEMPLO D-22 Ejemplo migrate-domain Sección <data> (Continuación)

```
<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>
```

Donde:

- Primero, el nodo <Content> (sin una sección <ldom_info>) es el dominio de origen para migrar.
- Segundo, el nodo <Content> (con una sección <ldom_info>) es el dominio de destino al que migrar. Los nombres del dominio de origen y destino pueden ser los mismos.
- La sección <ldom_info> para el dominio de destino describe el equipo al que migrar y los detalles necesarios para migrar a dicho equipo:
 - target-host es el equipo de destino al que migrar.
 - user-name es el nombre del usuario de inicio de sesión para el equipo de destino. Debe presentar codificación de 64 bits SASL.
 - password es la contraseña que se debe usar para el inicio de sesión en el equipo de destino. Debe presentar codificación de 64 bits SASL.

Nota – El Administrador de Dominios lógicos usa `sasl_decode64()` para decodificar el nombre de usuario de destino y la contraseña y usa `sasl_encode64()` para codificar estos valores. La codificación SASL 64 es equivalente a la codificación base64.

Esquemas XML de Administrador de Dominios lógicos

Este apéndice ofrece varios esquemas XML para usar con el Administrador de Dominios lógicos.

Este capítulo trata sobre los siguientes temas:

- “LDM_interface Esquema XML” en la página 257
- “LDM_Event (Esquema XML)” en la página 259
- “El esquema ovf-envelope.xsd” en la página 260
- “El esquema ovf-section.xsd” en la página 262
- “El esquema ovf-core.xsd” en la página 262
- “El esquema ovf-virtualhardware.xsc” en la página 267
- “El esquema cim-rasd.xsd” en la página 268
- “El esquema cim-vssd.xsd” en la página 272
- “El esquema cim-common.xsd” en la página 272
- “El esquema XML GenericProperty” en la página 276
- “Binding_Type (Esquema XML)” en la página 276

LDM_interface Esquema XML

Este esquema es una instantánea del borrador de especificación de formato abierto de virtualización (OVF) versión 0.98

EJEMPLO E-1 LDM_interface (Esquema XML)

```
<?xml version="1.0"?>
xs:schema
  xmlns:ovf="/var/opt/SUNWldom/envelope"
  xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:import namespace="/var/opt/SUNWldom/envelope" schemaLocation="ovf-envelope.xsd"/>

  <xs:annotation>
    <xs:documentation>
      Copyright (c) 2007, 2010, Oracle and/or its affiliates. All rights reserved.
```

EJEMPLO E-1 LDM_interface (Esquema XML) (Continuación)

```

        </xs:documentation>
    </xs:annotation>

<!--
=====
Type Definitions
=====
-->
<xs:simpleType name="statusStringType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="success"/>
    <xs:enumeration value="failure"/>
  </xs:restriction>
</xs:simpleType>

<xs:complexType name="responseType">
  <xs:sequence>
    <xs:element name="status" type="statusStringType"/>
    <xs:element name="resp_msg" type="xs:string" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>

<!-- LDM interface document -->
<xs:element name="LDM_interface">
  <xs:complexType>
    <xs:sequence>

      <!-- START cmd -->
      <xs:element name="cmd" minOccurs="1" maxOccurs="unbounded">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="action" type="xs:string" minOccurs="0"/>

            <!-- START data -->
            <xs:element name="data" minOccurs="0" maxOccurs="unbounded">
              <xs:complexType>
                <xs:choice minOccurs="1" maxOccurs="unbounded">

                  <!--OVF Envelope Version 0.9 -->
                  <xs:element name="Envelope" type="ovf:Envelope_Type"/>
                  <!-- DATA response -->
                  <xs:element name="response" type="responseType" minOccurs="0" maxOccurs="1"/>
                </xs:choice>
                <xs:attribute name="version" type="xs:string" use="required"/>
              </xs:complexType>
            </xs:element> <!-- END data -->

            <!-- CMD response -->
            <xs:element name="response" type="responseType" minOccurs="0" maxOccurs="1"/>

          </xs:sequence>
        </xs:complexType>
      </xs:element> <!-- END cmd -->

      <!-- DOCUMENT response -->
      <xs:element name="response" type="responseType" minOccurs="0" maxOccurs="1"/>
    </xs:sequence>
  </xs:complexType>
</xs:element> <!-- END LDM_interface -->

```

EJEMPLO E-1 LDM_interface (Esquema XML) (Continuación)

```

    </xs:sequence>
    <xs:attribute name="version" type="xs:string" use="required"/>
  </xs:complexType>
</xs:element> <!-- LDM interface document -->

</xs:schema>

```

LDM_Event (Esquema XML)

EJEMPLO E-2 LDM_Event (Esquema XML)

```

<?xml version="1.0"?>
<xs:schema
  xmlns:ovf="/var/opt/SUNWldom/envelope"
  xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:import namespace="/var/opt/SUNWldom/envelope" schemaLocation="ovf-envelope.xsd"/>

  <xs:annotation>
    <xs:documentation>
      Copyright (c) 2007, 2010, Oracle and/or its affiliates. All rights reserved.
    </xs:documentation>
  </xs:annotation>

  <!-- LDM interface document -->
  <xs:element name="LDM_event">
    <xs:complexType>
      <xs:sequence>

        <!-- START cmd -->
        <xs:element name="cmd" minOccurs="1" maxOccurs="unbounded">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="action" type="xs:string" minOccurs="0"/>

              <!-- START data -->
              <xs:element name="data" minOccurs="0" maxOccurs="unbounded">
                <xs:complexType>
                  <xs:choice minOccurs="1" maxOccurs="unbounded">

                    <!--OVF Evelope Version 0.9 -->
                    <xs:element name="Envelope" type="ovf:Envelope_Type"/>

                    </xs:choice>
                    <xs:attribute name="version" type="xs:string" use="required"/>
                  </xs:complexType>
                </xs:element> <!-- END data -->

              </xs:sequence>
            </xs:complexType>
          </xs:element> <!-- END cmd -->

        </xs:sequence>
      </xs:complexType>
    </xs:element> <!-- END LDM_event -->
  </xs:schema>

```

EJEMPLO E-2 LDM_Event (Esquema XML) (Continuación)

```

    </xs:sequence>
    <xs:attribute name="version" type="xs:string" use="required"/>
  </xs:complexType>
</xs:element> <!-- LDM interface document -->

</xs:schema>

```

El esquema ovf-envelope.xsd

EJEMPLO E-3 Elesquema ovf-envelope.xsd

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
  targetNamespace="/var/opt/SUNWldom/envelope"
  xmlns:ovf="/var/opt/SUNWldom/envelope"
  xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <!-- Include virtual hardware schema -->
  <xs:include schemaLocation="./ovf-section.xsd"/>
  <xs:include schemaLocation="./cim-virtualhardware.xsd"/>
  <xs:include schemaLocation="./ovf-core.xsd"/>

  <!-- Root element of a OVF package -->
  <xs:element name="Envelope" type="ovf:Envelope_Type"/>

  <xs:complexType name="Envelope_Type">
    <xs:sequence>
      <!--- References to all external files -->
      <xs:element name="References" type="ovf:References_Type"/>

      <!-- Package level meta-data -->
      <xs:element name="Section" type="ovf:Section_Type" minOccurs="0" maxOccurs="unbounded"/>

      <!-- Content. A virtual machine or a vService -->
      <xs:element name="Content" type="ovf:Entity_Type" minOccurs="0" maxOccurs="unbounded"/>

      <xs:any namespace="##targetNamespace" processContents="lax" minOccurs="0" maxOccurs="unbounded"/>
      <xs:any namespace="##other" processContents="lax" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="signed" type="xs:boolean" use="optional"/>
    <xs:attribute name="manifest" type="xs:boolean" use="optional"/>
    <xs:anyAttribute namespace="##any"/>
  </xs:complexType>

  <xs:complexType name="References_Type">
    <xs:sequence>
      <xs:element name="File" type="ovf:File_Type" minOccurs="0" maxOccurs="unbounded"/>
      <xs:any namespace="##targetNamespace" processContents="lax" minOccurs="0" maxOccurs="unbounded"/>
      <xs:any namespace="##other" processContents="lax" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <xs:anyAttribute namespace="##any"/>
</xs:schema>

```

EJEMPLO E-3 El esquema ovf - envelope.xsd (Continuación)

```

<!-- Type for an external reference to a resource -->
<xs:complexType name="File_Type">
  <xs:sequence>
    <xs:any namespace="##targetNamespace" processContents="lax" minOccurs="0" maxOccurs="unbounded"/>
    <xs:any namespace="##other" processContents="lax" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>

  <!-- Reference key used in other parts of the package -->
  <xs:attribute name="id" type="xs:string" use="required"/>
  <!-- Same as using a single part element -->
  <xs:attribute name="href" type="xs:string" use="required"/>
  <!-- Size in bytes of the files (if known) -->
  <xs:attribute name="size" type="xs:integer" use="optional"/>
  <!-- Estimated size in bytes of the files (if a good guess is known) -->
  <xs:attribute name="estSize" type="xs:integer" use="optional"/>
  <!-- Compression type (gzip or bzip2) -->
  <xs:attribute name="compression" type="xs:string" use="optional"/>
  <!-- Chunk size (except of last chunk) -->
  <xs:attribute name="chunkSize" type="xs:long" use="optional"/>

  <xs:anyAttribute namespace="##any"/>
</xs:complexType>

<!-- Base class for an entity -->
<xs:complexType name="Entity_Type" abstract="true">
  <xs:sequence>
    <xs:element name="Info" type="ovf:Info_Type" minOccurs="0" maxOccurs="unbounded"/>
    <xs:element name="Section" type="ovf:Section_Type" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
  <xs:attribute name="id" type="xs:string" use="required"/>
</xs:complexType>

<!-- A Virtual Machine Entity -->
<xs:complexType name="VirtualSystem_Type">
<xs:complexContent>
  <xs:extension base="ovf:Entity_Type"> </xs:extension>
</xs:complexContent>
</xs:complexType>

<!-- A Composite Service -->
<xs:complexType name="VirtualSystemCollection_Type">
  <xs:complexContent>
    <xs:extension base="ovf:Entity_Type">
      <xs:sequence>
        <xs:element name="Content" type="ovf:Entity_Type" minOccurs="0" maxOccurs="unbounded"/>
        <xs:any namespace="##targetNamespace" processContents="lax" minOccurs="0" maxOccurs="unbounded"/>
        <xs:any namespace="##other" processContents="lax" minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
</xs:schema>

```

El esquema ovf-section.xsd

EJEMPLO E-4 Elesquema ovf-section.xsd

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
  targetNamespace="/var/opt/SUNWldom/envelope"
  xmlns:ovf="/var/opt/SUNWldom/envelope"
  xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:import namespace="http://www.w3.org/XML/1998/namespace"
    schemaLocation="http://www.w3.org/2001/xml.xsd"/>

  <!-- The base class for a section. Subclassing this is the most common form of extensibility -->
  <xs:complexType name="Section_Type" abstract="true">
    <xs:sequence>
      <!-- The info element specifies the meaning of the section. This is typically shown
           if the section is not understood by the importer -->
      <xs:element name="Info" type="ovf:Info_Type" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  <!-- Whether the import should fail or not, if the section is not understood -->
  <xs:attribute name="required" type="xs:boolean" use="optional"/>
  <xs:anyAttribute namespace="##any"/>
  <!-- Subtypes defines more specific elements -->
</xs:complexType>

<!-- A basic type for a localizable string -->
<xs:complexType name="Info_Type">
  <xs:simpleContent>
    <xs:extension base="xs:string">
      <xs:attribute ref="xml:lang"/>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>
</xs:schema>
```

El esquema ovf-core.xsd

EJEMPLO E-5 Elesquema ovf-core.xsd

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
  targetNamespace="/var/opt/SUNWldom/envelope"
  xmlns:ovf="/var/opt/SUNWldom/envelope"
  xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:include schemaLocation="ovf-section.xsd"/>
  <xs:import namespace="http://www.w3.org/XML/1998/namespace"
    schemaLocation="http://www.w3.org/2001/xml.xsd"/>

  <!-- A user defined annotation on an entity -->
  <xs:complexType name="AnnotationSection_Type">
    <xs:complexContent>
      <xs:extension base="ovf:Section_Type">
        <xs:sequence>
```

EJEMPLO E-5 El esquema ovf-core.xsd (Continuación)

```

<!-- Several localized annotations can be included -->
<xs:element name="Annotation" type="ovf:Info_Type" minOccurs="0" maxOccurs="unbounded"/>
<xs:any namespace="##targetNamespace" processContents="lax" minOccurs="0"
  maxOccurs="unbounded"/>
<xs:any namespace="##other" processContents="lax" minOccurs="0" maxOccurs="unbounded"/>
</xs:sequence>
<xs:anyAttribute namespace="##any"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>

<!-- Product information about a virtual appliance -->
<xs:complexType name="ProductSection_Type">
  <xs:complexContent>
    <xs:extension base="ovf:Section_Type">
      <xs:sequence>
        <xs:element name="Product" type="ovf:Info_Type" minOccurs="0" maxOccurs="unbounded"/>
        <xs:element name="Vendor" type="ovf:Info_Type" minOccurs="0" maxOccurs="unbounded"/>
        <xs:element name="Version" type="xs:string" minOccurs="0"/>
        <xs:element name="Full-version" type="xs:string" minOccurs="0"/>
        <xs:element name="ProductUrl" type="xs:string" minOccurs="0"/>
        <xs:element name="VendorUrl" type="xs:string" minOccurs="0"/>
        <xs:element name="AppUrl" type="xs:string" minOccurs="0"/>
        <xs:any namespace="##targetNamespace" processContents="lax" minOccurs="0" maxOccurs="unbounded"/>
        <xs:any namespace="##other" processContents="lax" minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
      <xs:anyAttribute namespace="##any"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

<!-- Configuration parameters that can be passed to the virtual machine for
  application-level configuration -->
<xs:complexType name="PropertySection_Type">
  <xs:complexContent>
    <xs:extension base="ovf:Section_Type">
      <xs:sequence>
        <xs:element name="Property" maxOccurs="unbounded">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="Description" type="ovf:Info_Type" minOccurs="0" maxOccurs="unbounded"/>
              <xs:any namespace="##targetNamespace" processContents="lax" minOccurs="0"
                maxOccurs="unbounded"/>
              <xs:any namespace="##other" processContents="lax" minOccurs="0" maxOccurs="unbounded"/>
            </xs:sequence>
            <xs:attribute name="key" type="xs:string"/>
            <xs:attribute name="type" type="xs:string"/>
            <xs:attribute name="configurableByUser" type="xs:boolean" use="optional"/>
            <xs:attribute name="configurableAtRuntime" type="xs:boolean" use="optional"/>
            <xs:attribute name="defaultValue" type="xs:string" use="optional"/>
            <xs:anyAttribute namespace="##any"/>
          </xs:complexType>
        </xs:element>
        <xs:any namespace="##targetNamespace" processContents="lax" minOccurs="0" maxOccurs="unbounded"/>
        <xs:any namespace="##other" processContents="lax" minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

```

EJEMPLO E-5 Elesquema ovf-core.xsd (Continuación)

```

    <!-- A comma-separated list of transports that are supported by the virtual machine to
    access the OVF environment. -->
    <xs:attribute name="transport" type="xs:string" use="optional"/>
    <xs:anyAttribute namespace="##any"/>
  </xs:extension>
</xs:complexContent>
</xs:complexType>

<!-- Provides descriptions for the logical networks used within the package. These descriptions are
typically used as an aid when the package is deployed. -->
<xs:complexType name="NetworkSection_Type">
  <xs:complexContent>
    <xs:extension base="ovf:Section_Type">
      <xs:sequence>
        <xs:element name="Network" maxOccurs="unbounded">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="Description" type="ovf:Info_Type" minOccurs="0" maxOccurs="unbounded"/>
              <xs:any namespace="##targetNamespace" processContents="lax" minOccurs="0"
                maxOccurs="unbounded"/>
              <xs:any namespace="##other" processContents="lax" minOccurs="0" maxOccurs="unbounded"/>
            </xs:sequence>
            <xs:attribute name="name" type="xs:string" use="required"/>
            <xs:anyAttribute namespace="##any"/>
          </xs:complexType>
        </xs:element>
        <xs:any namespace="##targetNamespace" processContents="lax" minOccurs="0" maxOccurs="unbounded"/>
        <xs:any namespace="##other" processContents="lax" minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
      <xs:anyAttribute namespace="##any"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

<!-- Provides meta-information description of the virtual disks in the package -->
<xs:complexType name="DiskSection_Type">
  <xs:complexContent>
    <xs:extension base="ovf:Section_Type">
      <xs:sequence>
        <xs:element name="Disk" type="ovf:VirtualDiskDesc_Type" minOccurs="0" maxOccurs="unbounded"/>
        <xs:any namespace="##targetNamespace" processContents="lax" minOccurs="0" maxOccurs="unbounded"/>
        <xs:any namespace="##other" processContents="lax" minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
      <xs:anyAttribute namespace="##any"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<!-- Disk -->
<xs:complexType name="VirtualDiskDesc_Type">
  <!-- A logical ID for the virtual disk within this package -->
  <xs:attribute name="diskId" type="xs:string" use="required"/>
  <!-- A file reference to the virtual disk file. If this is not specified a blank virtual disk is
  created of the given size -->
  <xs:attribute name="fileRef" type="xs:string" use="optional"/>
  <!-- Capacity in bytes. The capacity can be specified as either a size or as a reference to a property
  using $(property_name) -->

```

EJEMPLO E-5 El esquema ovf-core.xsd (Continuación)

```

<xs:attribute name="capacity" type="xs:string" use="required"/>
<!-- Format of the disk. The format is an URL that identifies the disk type,
     e.g., http://www.vmware.com/format/vmdk.html#sparse -->
<xs:attribute name="format" type="xs:string" use="required"/>
<!-- Populated size of disk. This is an estimation of how much storage the disk needs if backed by
     a non pre-allocated (aka. sparse) disk. This size does not take the meta-data into
     account used by a sparse disk. -->
<xs:attribute name="populatedSize" type="xs:long" use="optional"/>
<!-- Reference to a potential parent disk -->
<xs:attribute name="parentRef" type="xs:string" use="optional"/>
</xs:complexType>

<!-- CPU Architecture requirements for the guest software. -->
<xs:complexType name="CpuCompatibilitySection_Type">
  <xs:complexContent>
    <xs:extension base="ovf:Section_Type">
      <xs:sequence>
        <xs:element name="Level" maxOccurs="unbounded">
          <xs:complexType>
            <xs:attribute name="level" type="xs:int" use="optional"/>
            <xs:attribute name="eax" type="xs:string" use="optional"/>
            <xs:attribute name="ebx" type="xs:string" use="optional"/>
            <xs:attribute name="ecx" type="xs:string" use="optional"/>
            <xs:attribute name="edx" type="xs:string" use="optional"/>
          </xs:complexType>
        </xs:element>
        <xs:any namespace="##targetNamespace" processContents="lax" minOccurs="0" maxOccurs="unbounded"/>
        <xs:any namespace="##other" processContents="lax" minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
      <xs:attribute name="Vendor" type="xs:string"/>
      <xs:anyAttribute namespace="##any"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

<!-- Specification of the operating system installed in the guest -->
<xs:complexType name="OperatingSystemSection_Type">
  <xs:complexContent>
    <xs:extension base="ovf:Section_Type">
      <xs:sequence>
        <xs:element name="Description" type="ovf:Info_Type" minOccurs="0" maxOccurs="unbounded"/>
        <xs:any namespace="##targetNamespace" processContents="lax" minOccurs="0" maxOccurs="unbounded"/>
        <xs:any namespace="##other" processContents="lax" minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
      <!-- The IDs are the enumeration used in CIM_OperatingSystem_Type -->
      <xs:attribute name="id" type="xs:string"/>
      <xs:anyAttribute namespace="##any"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

<!-- End-User License Agreement -->
<xs:complexType name="EulaSection_Type">
  <xs:complexContent>
    <xs:extension base="ovf:Section_Type">
      <xs:sequence>

```

EJEMPLO E-5 Elesquema ovf-core.xsd (Continuación)

```

<!-- Contains the license agreement in plain text. Several different locales can be
specified -->
<xs:element name="License" type="ovf:Info_Type" minOccurs="1" maxOccurs="unbounded"/>
<xs:any namespace="##targetNamespace" processContents="lax" minOccurs="0" maxOccurs="unbounded"/>
<xs:any namespace="##other" processContents="lax" minOccurs="0" maxOccurs="unbounded"/>
</xs:sequence>
<xs:anyAttribute namespace="##any"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>

<!-- For a VirtualSystemCollection, this section is used to specify the order in which the
contained entities are to be powered on. -->
<xs:complexType name="StartupSection_Type">
<xs:complexContent>
<xs:extension base="ovf:Section_Type">
<xs:sequence>
<xs:element name="item" minOccurs="0" maxOccurs="unbounded">
<xs:complexType>
<!-- Id of entity in collection -->
<xs:attribute name="id" type="xs:string"/>
<!-- Startup order. Entities are started up starting with lower-numbers first. Items with
same order identifier may be started up concurrently or in any order.
The order is reversed for shutdown. -->
<xs:attribute name="order" type="xs:int"/>
<!-- Delay in seconds to wait for the power on to complete -->
<xs:attribute name="startDelay" type="xs:int"/>
<!-- Whether to resume power-on sequence, once the guest reports ok. -->
<xs:attribute name="waitingForGuest" type="xs:boolean"/>
<!-- Delay in seconds to wait for the power on to complete -->
<xs:attribute name="stopDelay" type="xs:int"/>
<!-- Stop action to use. Valid values are: 'powerOn' (default), 'none'. -->
<xs:attribute name="startAction" type="xs:string"/>
<!-- Stop action to use. Valid values are: 'powerOff' (default), 'guestShutdown',
'suspend'. -->
<xs:attribute name="stopAction" type="xs:string"/>
<xs:anyAttribute namespace="##any"/>
</xs:complexType>
</xs:element>
<xs:any namespace="##targetNamespace" processContents="lax" minOccurs="0" maxOccurs="unbounded"/>
<xs:any namespace="##other" processContents="lax" minOccurs="0" maxOccurs="unbounded"/>
</xs:sequence>
<!-- A comma-separated list of transports that the virtual machine supports to provide
feedback. -->
<xs:anyAttribute namespace="##any"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>

<!-- If this section is present, it indicates that the virtual machine needs to be initially
booted to install and configure the software. -->
<xs:complexType name="InstallSection_Type">
<xs:complexContent>
<xs:extension base="ovf:Section_Type">
<xs:sequence>
<xs:any namespace="##targetNamespace" processContents="lax" minOccurs="0"

```

EJEMPLO E-5 El esquema ovf-core.xsd (Continuación)

```

        maxOccurs="unbounded"/>
    <xs:any namespace="##other" processContents="lax" minOccurs="0" maxOccurs="unbounded"/>
</xs:sequence>
<!-- A comma-separated list of transports that the virtual machine supports to provide
feedback. -->
<xs:attribute name="transport" type="xs:string"/>
<xs:anyAttribute namespace="##any"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>
</xs:schema>

```

El esquema ovf-virtualhardware.xsc

EJEMPLO E-6 El esquema ovf-virtualhardware.xsc

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
    targetNamespace="/var/opt/SUNWldom/envelope"
    xmlns:ovf="/var/opt/SUNWldom/envelope"
    xmlns:xs="http://www.w3.org/2001/XMLSchema"
    xmlns:vssd="/var/opt/SUNWldom/CIM_VirtualSystemSettingData"
    xmlns:rasd="/var/opt/SUNWldom/CIM_ResourceAllocationSettingData">

    <xs:import namespace="http://www.w3.org/XML/1998/namespace"
        schemaLocation="http://www.w3.org/2001/xml.xsd"/>

    <xs:include schemaLocation="ovf-section.xsd"/>

    <xs:import namespace="/var/opt/SUNWldom/CIM_VirtualSystemSettingData" schemaLocation="cim-vssd.xsd"/>
    <xs:import namespace="/var/opt/SUNWldom/CIM_ResourceAllocationSettingData"
        schemaLocation="cim-rasd.xsd"/>

    <!-- Specifies the virtual hardware for a virtual machine -->
    <xs:complexType name="VirtualHardwareSection_Type">
        <xs:complexContent>
            <xs:extension base="ovf:Section_Type">
                <xs:sequence>
                    <xs:element name="System" type="vssd:CIM_VirtualSystemSettingData_Type" minOccurs="0"/>
                    <xs:element name="Item" type="rasd:CIM_ResourceAllocationSettingData_Type"
                        minOccurs="0" maxOccurs="unbounded"/>
                </xs:sequence>
            </xs:extension>
        </xs:complexContent>
    </xs:complexType>

    <!-- Specifies a section for resource constraints on a VirtualSystemCollection -->
    <xs:complexType name="ResourceAllocationSection_Type">
        <xs:complexContent>
            <xs:extension base="ovf:Section_Type">
                <xs:sequence>
                    <xs:element name="Item" type="rasd:CIM_ResourceAllocationSettingData_Type"
                        minOccurs="0" maxOccurs="unbounded"/>
                </xs:sequence>
            </xs:extension>
        </xs:complexContent>
    </xs:complexType>

```

```

    </xs:sequence>
  </xs:extension>
</xs:complexContent>
</xs:complexType>
</xs:schema>

```

El esquema cim-rasd.xsd

EJEMPLO E-7 Elesquema cim-rasd.xsd

```

<?xml version='1.0' encoding='utf-8'?>
<xs:schema
  targetNamespace="/var/opt/SUNWldom/CIM_ResourceAllocationSettingData"
  xmlns:class="/var/opt/SUNWldom/CIM_ResourceAllocationSettingData"
  xmlns:cim="/var/opt/SUNWldom/common"
  xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:import namespace="/var/opt/SUNWldom/common" schemaLocation="cim-common.xsd"/>

  <xs:element name="Caption" nillable="true" type="cim:cimString"/>

  <xs:element name="Description" nillable="true" type="cim:cimString"/>

  <xs:element name="InstanceId" nillable="true" type="cim:cimString"/>

  <xs:element name="ResourceType" nillable="true">
    <xs:complexType>
      <xs:simpleContent>
        <xs:restriction base="xs:anyType">
          <xs:simpleType>
            <xs:union>
              <xs:simpleType>
                <xs:restriction base="xs:unsignedShort">
                  <xs:enumeration value="1"/> <!-- Other -->
                  <xs:enumeration value="2"/> <!-- Computer System -->
                  <xs:enumeration value="3"/> <!-- Processor-->
                  <xs:enumeration value="4"/> <!-- Memory-->
                  <xs:enumeration value="5"/> <!-- IDE Controller -->
                  <xs:enumeration value="6"/> <!-- Parallel SCSI HBA -->
                  <xs:enumeration value="7"/> <!-- FC HBA -->
                  <xs:enumeration value="8"/> <!-- iSCSI HBA -->
                  <xs:enumeration value="9"/> <!-- IB HCA -->
                  <xs:enumeration value="10"/> <!-- Ethernet Adapter -->
                  <xs:enumeration value="11"/> <!-- Other Network Adapter -->
                  <xs:enumeration value="12"/> <!-- I/O Slot -->
                  <xs:enumeration value="13"/> <!-- I/O Device -->
                  <xs:enumeration value="14"/> <!-- Floppy Drive -->
                  <xs:enumeration value="15"/> <!-- CD Drive -->
                  <xs:enumeration value="16"/> <!-- DVD drive -->
                  <xs:enumeration value="17"/> <!-- Disk Drive -->
                  <xs:enumeration value="18"/> <!-- Tape Drive -->
                  <xs:enumeration value="19"/> <!-- Storage Extent -->
                  <xs:enumeration value="20"/> <!-- Other storage device -->
                
```

EJEMPLO E-7 El esquema cim-rasd.xsd (Continuación)

```

<xs:enumeration value="21"/> <!-- Serial port -->
<xs:enumeration value="22"/> <!-- Parallel port -->
<xs:enumeration value="23"/> <!-- USB Controller -->
<xs:enumeration value="24"/> <!-- Graphics controller -->
<xs:enumeration value="25"/> <!-- IEEE 1394 Controller -->
<xs:enumeration value="26"/> <!-- Partitionable Unit -->
<xs:enumeration value="27"/> <!-- Base Partitionable Unit -->
<xs:enumeration value="28"/> <!-- Power Supply -->
<xs:enumeration value="29"/> <!-- Cooling Device -->
<xs:enumeration value="29"/> <!-- Cooling Device -->
<xs:enumeration value="31"/> <!-- PS2 Controller -->
<xs:enumeration value="32"/> <!-- SIO Controller -->
<xs:enumeration value="33"/> <!-- Keyboard -->
<xs:enumeration value="34"/> <!-- Pointing Device -->
</xs:restriction>
</xs:simpleType>
<xs:simpleType>
  <xs:restriction base="xs:unsignedShort">
    <xs:minInclusive value="30"/>
    <xs:maxInclusive value="32769"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType>
  <xs:restriction base="xs:unsignedShort">
    <xs:minInclusive value="32768"/>
    <xs:maxInclusive value="65535"/>
  </xs:restriction>
</xs:simpleType>
</xs:union>
</xs:simpleType>
<xs:anyAttribute namespace="##any" />
</xs:restriction>
</xs:simpleContent>
</xs:complexType>
</xs:element>

<xs:element name="OtherResourceType" nillable="true" type="cim:cimString"/>
<xs:element name="ResourceSubType" nillable="true" type="cim:cimString"/>
<xs:element name="PoolID" nillable="true" type="cim:cimString"/>
<xs:element name="ConsumerVisibility" nillable="true">
  <xs:complexType>
    <xs:simpleContent>
      <xs:restriction base="xs:anyType">
        <xs:simpleType>
          <xs:union>
            <xs:simpleType>
              <xs:restriction base="xs:unsignedShort">
                <xs:enumeration value="0"/>
                <xs:enumeration value="2"/>
                <xs:enumeration value="3"/>
                <xs:enumeration value="4"/>
              </xs:restriction>
            </xs:simpleType>
          </xs:union>
        </xs:simpleType>
      </xs:restriction>
    </xs:simpleContent>
  </xs:complexType>
</xs:element>

```

EJEMPLO E-7 Elesquema cim-rasd.xsd (Continuación)

```

<xs:simpleType>
  <xs:restriction base="xs:unsignedShort">
    <xs:minInclusive value="5"/>
    <xs:maxInclusive value="32768"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType>
  <xs:restriction base="xs:unsignedShort">
    <xs:minInclusive value="32767"/>
    <xs:maxInclusive value="65535"/>
  </xs:restriction>
</xs:simpleType>
</xs:union>
</xs:simpleType>
<xs:anyAttribute namespace="##any"/>
</xs:restriction>
</xs:simpleContent>
</xs:complexType>
</xs:element>

<xs:element name="HostResource" nillable="true" type="xs:anyType"/>
<xs:element name="AllocationUnits" nillable="true" type="cim:cimString"/>
<xs:element name="VirtualQuantity" nillable="true" type="cim:cimUnsignedLong"/>
<xs:element name="Reservation" nillable="true" type="cim:cimUnsignedLong"/>
<xs:element name="Limit" nillable="true" type="cim:cimUnsignedLong"/>
<xs:element name="Weight" nillable="true" type="cim:cimUnsignedInt"/>
<xs:element name="AutomaticAllocation" nillable="true" type="cim:cimBoolean"/>
<xs:element name="AutomaticDeallocation" nillable="true" type="cim:cimBoolean"/>
<xs:element name="Parent" nillable="true" type="cim:cimString"/>
<xs:element name="Connection" nillable="true" type="cim:cimString"/>
<xs:element name="Address" nillable="true" type="cim:cimString"/>
<xs:element name="MappingBehavior" nillable="true">
  <xs:complexType>
    <xs:simpleContent>
      <xs:restriction base="xs:anyType">
        <xs:simpleType>
          <xs:union>
            <xs:simpleType>
              <xs:restriction base="xs:unsignedShort">
                <xs:enumeration value="0"/>
                <xs:enumeration value="1"/>
                <xs:enumeration value="2"/>
                <xs:enumeration value="3"/>
                <xs:enumeration value="4"/>
              </xs:restriction>
            </xs:simpleType>
            <xs:simpleType>
              <xs:restriction base="xs:unsignedShort">
                <xs:minInclusive value="5"/>
                <xs:maxInclusive value="32768"/>
              </xs:restriction>
            </xs:simpleType>
            <xs:simpleType>
              <xs:restriction base="xs:unsignedShort">
                <xs:minInclusive value="32767"/>
                <xs:maxInclusive value="65535"/>
              </xs:restriction>
            </xs:simpleType>
          </xs:union>
        </xs:restriction>
      </xs:simpleContent>
    </xs:complexType>
  </xs:element>

```

EJEMPLO E-7 El esquema cim-rasd.xsd (Continuación)

```

        </xs:restriction>
      </xs:simpleType>
    </xs:union>
  </xs:simpleType>
  <xs:anyAttribute namespace="##any"/>
</xs:restriction>
</xs:simpleContent>
</xs:complexType>
</xs:element>
<xs:element name="AddressOnParent" nillable="true" type="cim:cimString"/>

<xs:element name="BusNumber" nillable="true" type="cim:cimUnsignedShort"/>

<xs:complexType name="CIM_ResourceAllocationSettingData_Type">
  <xs:sequence>
    <xs:element ref="class:Caption" minOccurs="0" maxOccurs="unbounded"/>
    <xs:element ref="class:Description" minOccurs="0" maxOccurs="unbounded"/>
    <xs:element ref="class:InstanceId" minOccurs="0"/>
    <xs:element ref="class:ResourceType" minOccurs="0"/>
    <xs:element ref="class:OtherResourceType" minOccurs="0"/>
    <xs:element ref="class:ResourceSubType" minOccurs="0"/>
    <xs:element ref="class:PoolID" minOccurs="0"/>
    <xs:element ref="class:ConsumerVisibility" minOccurs="0"/>
    <xs:element ref="class:HostResource" maxOccurs="unbounded" minOccurs="0"/>
    <xs:element ref="class:AllocationUnits" minOccurs="0"/>
    <xs:element ref="class:VirtualQuantity" minOccurs="0"/>
    <xs:element ref="class:Reservation" minOccurs="0"/>
    <xs:element ref="class:Limit" minOccurs="0"/>
    <xs:element ref="class:Weight" minOccurs="0"/>
    <xs:element ref="class:AutomaticAllocation" minOccurs="0"/>
    <xs:element ref="class:AutomaticDeallocation" minOccurs="0"/>
    <xs:element ref="class:Parent" minOccurs="0"/>
    <xs:element ref="class:Connection" maxOccurs="unbounded" minOccurs="0"/>
    <xs:element ref="class:Address" minOccurs="0"/>
    <xs:element ref="class:MappingBehavior" minOccurs="0"/>
    <xs:element ref="class:AddressOnParent" minOccurs="0"/>
    <xs:element ref="class:BusNumber" minOccurs="0"/>
    <xs:any namespace="##other" processContents="lax" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
  <xs:anyAttribute namespace="##any"/>
</xs:complexType>

<xs:element name="CIM_ResourceAllocationSettingData"
  type="class:CIM_ResourceAllocationSettingData_Type"/>
</xs:schema>

```

El esquema cim-vssd.xsd

EJEMPLO E-8 Elesquema cim-vssd.xsd

```
<?xml version='1.0' encoding='utf-8'?>
<xs:schema
  targetNamespace="/var/opt/SUNWldom/CIM_VirtualSystemSettingData"
  xmlns:class="/var/opt/SUNWldom/CIM_VirtualSystemSettingData"
  xmlns:cim="/var/opt/SUNWldom/common"
  xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:import namespace="/var/opt/SUNWldom/common"
    schemaLocation="cim-common.xsd"/>

  <xs:element name="Caption" nillable="true" type="cim:cimString"/>

  <xs:element name="Description" nillable="true" type="cim:cimString"/>

  <xs:element name="InstanceId" nillable="true" type="cim:cimString"/>

  <xs:element name="VirtualSystemIdentifier" nillable="true" type="cim:cimString"/>

  <xs:element name="VirtualSystemType" nillable="true" type="cim:cimString"/>

  <xs:complexType name="CIM_VirtualSystemSettingData_Type">
    <xs:sequence>
      <xs:element ref="class:Caption" minOccurs="0" maxOccurs="unbounded"/>
      <xs:element ref="class:Description" minOccurs="0" maxOccurs="unbounded"/>
      <xs:element ref="class:InstanceId" minOccurs="0"/>
      <xs:element ref="class:VirtualSystemIdentifier" minOccurs="0"/>
      <xs:element ref="class:VirtualSystemType" minOccurs="0"/>
      <xs:any namespace="##other" processContents="lax" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:anyAttribute namespace="##any"/>
  </xs:complexType>

  <xs:element name="CIM_VirtualSystemSettingData" type="class:CIM_VirtualSystemSettingData_Type"/>
</xs:schema>
```

El esquema cim-common.xsd

EJEMPLO E-9 Elesquema cim-common.xsd

```
<?xml version="1.0" encoding="utf-8"?>
<xs:schema
  targetNamespace="/var/opt/SUNWldom/common"
  xmlns:cim="/var/opt/SUNWldom/common"
  xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">

  <!-- The following are runtime attribute definitions -->
  <xs:attribute name="Key" type="xs:boolean"/>

  <xs:attribute name="Version" type="xs:string"/>
```

EJEMPLO E-9 El esquema cim-common.xsd (Continuación)

```

<!-- The following section defines the extended WS-CIM datatypes -->
<xs:complexType name="cimDateTime">
  <xs:choice>
    <xs:element name="CIM_DateTime" type="xs:string" nillable="true"/>
    <xs:element name="Interval" type="xs:duration"/>
    <xs:element name="Date" type="xs:date"/>
    <xs:element name="Time" type="xs:time"/>
    <xs:element name="Datetime" type="xs:dateTime"/>
  </xs:choice>
  <xs:anyAttribute namespace="##any" processContents="lax"/>
</xs:complexType>

<xs:complexType name="cimUnsignedByte">
  <xs:simpleContent>
    <xs:extension base="xs:unsignedByte">
      <xs:anyAttribute namespace="##any" processContents="lax"/>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>

<xs:complexType name="cimByte">
  <xs:simpleContent>
    <xs:extension base="xs:byte">
      <xs:anyAttribute namespace="##any" processContents="lax"/>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>

<xs:complexType name="cimUnsignedShort">
  <xs:simpleContent>
    <xs:extension base="xs:unsignedShort">
      <xs:anyAttribute namespace="##any" processContents="lax"/>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>

<xs:complexType name="cimShort">
  <xs:simpleContent>
    <xs:extension base="xs:short">
      <xs:anyAttribute namespace="##any" processContents="lax"/>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>

<xs:complexType name="cimUnsignedInt">
  <xs:simpleContent>
    <xs:extension base="xs:unsignedInt">
      <xs:anyAttribute namespace="##any" processContents="lax"/>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>

<xs:complexType name="cimInt">
  <xs:simpleContent>
    <xs:extension base="xs:int">

```

EJEMPLO E-9 El esquema cim-common.xsd (Continuación)

```
<xs:anyAttribute namespace="##any" processContents="lax"/>
</xs:extension>
</xs:simpleContent>
</xs:complexType>

<xs:complexType name="cimUnsignedLong">
  <xs:simpleContent>
    <xs:extension base="xs:unsignedLong">
      <xs:anyAttribute namespace="##any" processContents="lax"/>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>

<xs:complexType name="cimLong">
  <xs:simpleContent>
    <xs:extension base="xs:long">
      <xs:anyAttribute namespace="##any" processContents="lax"/>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>

<xs:complexType name="cimString">
  <xs:simpleContent>
    <xs:extension base="xs:string">
      <xs:anyAttribute namespace="##any" processContents="lax"/>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>

<xs:complexType name="cimBoolean">
  <xs:simpleContent>
    <xs:extension base="xs:boolean">
      <xs:anyAttribute namespace="##any" processContents="lax"/>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>

<xs:complexType name="cimFloat">
  <xs:simpleContent>
    <xs:extension base="xs:float">
      <xs:anyAttribute namespace="##any" processContents="lax"/>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>

<xs:complexType name="cimDouble">
  <xs:simpleContent>
    <xs:extension base="xs:double">
      <xs:anyAttribute namespace="##any" processContents="lax"/>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>

<xs:complexType name="cimChar16">
  <xs:simpleContent>
    <xs:restriction base="cim:cimString">
```

EJEMPLO E-9 El esquema cim-common.xsd (Continuación)

```

        <xs:maxLength value="1"/>
        <xs:anyAttribute namespace="##any" processContents="lax"/>
    </xs:restriction>
</xs:simpleContent>
</xs:complexType>

<xs:complexType name="cimBase64Binary">
    <xs:simpleContent>
        <xs:extension base="xs:base64Binary">
            <xs:anyAttribute namespace="##any" processContents="lax"/>
        </xs:extension>
    </xs:simpleContent>
</xs:complexType>

<xs:complexType name="cimHexBinary">
    <xs:simpleContent>
        <xs:extension base="xs:hexBinary">
            <xs:anyAttribute namespace="##any" processContents="lax"/>
        </xs:extension>
    </xs:simpleContent>
</xs:complexType>

<xs:complexType name="cimReference">
    <xs:sequence>
        <xs:any namespace="##other" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:anyAttribute namespace="##any" processContents="lax"/>
</xs:complexType>

<!-- The following datatypes are used exclusively to define metadata fragments -->
<xs:attribute name="qualifier" type="xs:boolean"/>

<xs:complexType name="qualifierString">
    <xs:simpleContent>
        <xs:extension base="cim:cimString">
            <xs:attribute ref="cim:qualifier" use="required"/>
        </xs:extension>
    </xs:simpleContent>
</xs:complexType>

<xs:complexType name="qualifierBoolean">
    <xs:simpleContent>
        <xs:extension base="cim:cimBoolean">
            <xs:attribute ref="cim:qualifier" use="required"/>
        </xs:extension>
    </xs:simpleContent>
</xs:complexType>

<xs:complexType name="qualifierUInt32">
    <xs:simpleContent>
        <xs:extension base="cim:cimUnsignedInt">
            <xs:attribute ref="cim:qualifier" use="required"/>
        </xs:extension>
    </xs:simpleContent>
</xs:complexType>

```

EJEMPLO E-9 El esquema cim-common.xsd (Continuación)

```

<xs:complexType name="qualifierSInt64">
  <xs:simpleContent>
    <xs:extension base="cim:cimLong">
      <xs:attribute ref="cim:qualifier" use="required"/>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>
<!--
<xs:complexType name="qualifierSArray">
  <xs:complexContent>
    <xs:extension base="cim:qualifierString"/>
  </xs:complexContent>
</xs:complexType>
-->
<!-- The following element is to be used only for defining metadata -->
<xs:element name="DefaultValue" type="xs:anySimpleType"/>
</xs:schema>

```

El esquema XML GenericProperty

Este esquema es una extensión del esquema para formato abierto de virtualización (OVF).

EJEMPLO E-10 El esquema XML GenericProperty

```

<?xml version='1.0' encoding='utf-8'?>
<xs:schema
  targetNamespace="/var/opt/SUNWldom/GenericProperty"
  xmlns:class="/var/opt/SUNWldom/GenericProperty"
  xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:complexType name="GenericProperty_Type" type="xs:string">
    <xs:attribute name="key" type="xs:string" use="required"/>

  </xs:complexType>
  <xs:element name="GenericProperty" type="class:GenericProperty_Type"/>

</xs:schema>

```

Binding_Type (Esquema XML)

Este esquema es una extensión del esquema para formato abierto de virtualización (OVF).

EJEMPLO E-11 Binding_Type (Esquema XML)

```

<?xml version='1.0' encoding='utf-8'?>
<xs:schema
  targetNamespace="/var/opt/SUNWldom/Binding"
  xmlns:class="/var/opt/SUNWldom/Binding"
  xmlns:rasd="/var/opt/SUNWldom/CIM_ResourceAllocationSettingData"

```

EJEMPLO E-11 Binding_Type (Esquema XML) (Continuación)

```
xmlns:xs="http://www.w3.org/2001/XMLSchema">
<xs:import namespace="/var/opt/SUNWldom/CIM_ResourceAllocationSettingData"
  schemaLocation="cim-rasd.xsd"/>
<xs:complexType name="Binding_Type">
  <xs:sequence>
    <xs:element name="Item"
      type="rasd:CIM_ResourceAllocationSettingData_Type"/>
  </xs:sequence>
</xs:complexType>
</xs:schema>
```


Glosario

Esta lista define la terminología, abreviaciones, y acrónimos usados en la documentación de Oracle VM Server for SPARC.

A

API	Interfaz de programación de aplicaciones
auditreduce	Une y selecciona la información de auditoría desde los archivos de seguimiento de auditoría (véase la página de comandos man auditreduce(1M)).
auditoría	Uso de la auditoría del SO de Solaris de Oracle para identificar la fuente de los cambios de seguridad
autorización	Configuración de la autorización usando el RBAC del SO de Solaris de Oracle

B

bge	Controlador de Ethernet Broadcom Gigabit en dispositivos Broadcom BCM57xx
BSM	Módulo de seguridad básico
bsmconv	Habilita el BSM (véase la página de comandos man bsmconv(1M)).
bsmunconv	Deshabilita el BSM (véase la página de comandos man bsmunconv(1M)).

C

CD	Disco compacto
CLI	Interfaz de la línea de comandos
cumplimiento	Determina si la configuración de un sistema cumple el perfil de seguridad predefinido

configuración	Nombre de la configuración del dominio lógico que está guardado en el procesador de servicio
CMT	Multiprocesamiento de chip
restricciones	Para el Administrador de Dominios lógicos, las restricciones son uno o varios recursos que desea asignar a un dominio específico. O bien recibe todos los recursos que ha solicitado que se agreguen a un dominio o ninguno de éstos, dependiendo de los recursos disponibles.
dominio de control	Dominio que crea y administra otros dominios lógicos y servicios
CPU	Unidad central de procesamiento
CWQ	Control Word Queue; unidad criptográfica para las plataformas de Oracle Sun basadas en UltraSPARC T2

D

DHCP	Protocolo de configuración dinámica de host
DMA	Acceso directo a memoria, es la habilidad de transferir directamente datos entre la memoria y un dispositivo (por ejemplo, una tarjeta de red) sin implicar a la CPU.
DMP	Multirruta dinámica (Veritas)
Administrador de Dominios lógicos	Una CLI para crear y administrar dominios lógicos y asignar recursos a los dominios
DPS	Data plane software
DR	Reconfiguración dinámica
drd	daemon de reconfiguración dinámica del SO 10 de Solaris de Oracle para Administrador de Dominios lógicos (véase la página de comandos man drd(1M)).
DS	Módulo de servicios de dominio (SO 10 Solaris de Oracle)
DVD	Disco versátil digital

E

EFI	Interfaz extensible del firmware
ETM	Módulo de administración de la tabla de codificación (SO 10 de Solaris de Oracle)

F

FC_AL	Bucle arbitrado de canal de fibra
FMA	Arquitectura de administración de fallos
fmd	daemon del administrador de fallos del SO 10 de Solaris de Oracle (véase la página de comando man fmd(1M)).
formato	Utilidad de partición del disco y mantenimiento (véase la página de comandos man format(1M)).
fmthard	Populación de etiqueta en discos duros (véase la página de comandos man fmthard(1M)).
FTP	Protocolo de transferencia de archivos

G

Gb	Gigabit
dominio huésped	Usa servicios de E/S y dominios de servicios y está administrado por el dominio de control.
GLDv3	Controlador LAN genérico versión 3.

H

blindaje	Modificación de la configuración de SO de Solaris de Oracle para mejorar la seguridad
HDD	Disco duro
hypervisor	Capa de firmware interpuesta entre el sistema operativo y la capa de hardware

I

Dominio E/S	Dominio que tiene la propiedad directa y el acceso directo a los dispositivos físicos de E/S y que comparte esos dispositivos con otros dominios lógicos en forma de dispositivos virtuales
IB	Infiniband
IDE	Controlador electrónico incorporado
IDR	Lanzamiento de diagnóstico intermedio
ILOM	Integrated Lights Out Manager

io	Dispositivos E/S, como discos internos y controladores PCIe y los adaptadores y dispositivos acoplados
ioctl	Llamada de control de entrada/salida
IP	Protocolo de internet
IPMP	Ruta múltiple de red de protocolo de internet
ISO	Organización internacional para la estandarización

K

kaio	Entrada/salida asíncrona de núcleo
KB	Kilobyte
KU	Actualización de núcleo

L

LAN	Red de área local
LDAP	Protocolo de acceso ligero a directorios
LDC	Canal de dominio lógico
ldm	utilidad de Administrador de Dominios lógicos (véase la página de comandos <code>man ldm(1M)</code>).
ldmd	daemon de Administrador de Dominios lógicos
lofi	Archivo de bucle invertido
dominio lógico	Un equipo virtual formado por un agrupamiento lógico discreto de recursos, que tiene su propio sistema operativo e identidad en un sistema de ordenador individual
LUN	Número de unidad lógica

M

MAC	Dirección de control de acceso a medios, que Dominios lógicos puede asignar automáticamente o usted puede asignar manualmente
MAU	Unidad aritmética modular

MB	Megabyte
MD	Descripción de la máquina en la base de datos del servidor
mem, memoria	Unidad de memoria - tamaño predeterminado en bytes, o especificado en gigabytes (G), kilobytes (K) o megabytes (M). Memoria virtualizada del servidor que puede ser asignada a los dominios huésped.
metadb	Crea y borra replicas de la base de datos de estado de metadispositivos Solaris Volume Manager (véase la página de comandos man metadb(1M)).
metaset	Configura conjuntos de discos (véase la página del manual metaset(1M)).
mhd	Operaciones de control del disco multihost (véase la página de comandos man mhd(7i)).
MIB	Base de datos de información de administración (MIB)
minimización	Instalación del mínimo número de núcleo del paquete SO de Solaris de Oracle necesarios
MMF	Fibra de modo múltiple
MMU	Unidad de administración de la memoria
mpgroup	Nombre del grupo de ruta múltiple para conmutación por error de disco virtual
mtu	Unidad de transmisión máxima

N

NAT	Traducción de la dirección de red
ndpsldcc	Cliente del canal del dominio lógico Netra DPS. <i>Véase también</i> vdpcc.
ndpsldcs	Servicio del canal del dominio lógico Netra DPS. <i>Véase también</i> vdpcs.
NFS	Sistema de archivos de red
NIS	Servicios de información de red
NIU	Unidad de interfaz de red (servidores SPARC Enterprise T5120 y T5220 de Sun Oracle)
NTS	Servidor del terminal de red
NVRAM	Memoria de acceso aleatorio no volátil
nxge	Controlador para un adaptador de Ethernet de 10 Gb de NIU

O

SO	Sistema operativo
OVF	Formato abierto de virtualización

P

P2V	Herramienta de conversión física a virtual de Dominios lógicos
PA	Dirección física
PCI	Bus de interconexión de componentes periféricos
PCIe	Bus PCI EXPRESS
PCI-X	Bus PCI Extended
pcpu	CPU física
physio	Entrada/salida física
PICL	Información de plataforma y biblioteca de control
picld	Daemon de PICL (véase la página de comandos man picld(1M)).
PM	Administración de energía de CPU virtual y memoria
praudit	Imprime los contenidos y el archivos de seguimiento de auditoría (véase la página de comandos man praudit(1M)).
PRI	Prioridad

R

RA	Dirección real
RAID	Matriz redundante de discos independientes
RBAC	Control de acceso basado en funciones
RPC	Llamada de procedimiento remoto

S

SASL	Autenticación simple y capa de seguridad
SAX	Simple API para el analizador de XML, que atraviesa un documento XML. El analizador SAX se basa en eventos y se usa sobre todo para datos de streaming.
controlador de sistema (SC)	Véase también procesador de servicio
SCSI	Interfaz para sistemas de ordenadores pequeños
dominio de servicio	Dominio lógico que suministra dispositivos, como conmutadores virtuales, conectores de consola virtual y servidores de disco virtual a otros dominios lógicos
SMA	Agente de administración de sistema
SMF	Dispositivo de administración de servicios
SNMP	Protocolo simple de administración de red
procesador de servicio (SP)	El SP, también conocido como controlador de sistema (SC), efectúa un seguimiento y ejecuta el equipo físico.
SSH	Intérprete de órdenes segura
ssh	Comando de intérprete de órdenes segura (véase la página de comandos man ssh(1)).
sshd	Daemon del intérprete de órdenes segura (véase la página de comandos man sshd(1M)).
SunVTS	Sun Validation Test Suite
svcadm	Manipula instancias de servicio (véase la página de comandos man svcadm(1M)).

T

TCP	Protocolo de control de la transmisión
TLS	Seguridad de la capa de transporte

U

UDP	Protocolo del diagrama de usuario
UFS	Sistema de archivos UNIX

unicast	Comunicación de redes que se efectúa entre un remitente individual y un receptor individual.
USB	Bus universal en serie
uscsi	Interfaz de comando de SCSI del usuario (véase la página de comandos man uscsi(7I)).
UTP	Cable trenzado sin apantallar

V

var	Variable
VBSC	Controlador del sistema de servidor blade virtual
vcc, vconscn	Servicio de concentrador de consola virtual con un rango de puerto específico para asignar a los dominios huésped
vcons, vconsole	Consola virtual para acceder a los mensajes a nivel de sistema. Se consigue una conexión conectando el servicio vconscn en el dominio de control a un puerto específico.
vcpu	Unidad de procesamiento central virtual. Cada núcleo en un servidor está representado por una CPU virtual. Por ejemplo, un servidor Sun Fire T2000 de 8 núcleos de Oracle tiene 32 CPU virtuales que pueden ser asignadas a los dominios lógicos.
vdc	Cliente de disco virtual
vdisk	Un disco virtual es un dispositivo de bloque genérico asociado con diferentes tipos de dispositivos físico, volúmenes o archivos.
vdppc	Cliente de canal plano de datos virtuales en un entorno Netra DPS
vdpcs	Servicio de canal plano de datos virtuales en un entorno Netra DPS
vds, vdiskserver	El servidor del disco virtual le permite importar discos virtuales en un dominio lógico.
vdsdev, vdiskserverdevice	El dispositivo del servidor del disco virtual es exportado por el servidor del disco virtual. El dispositivo puede ser todo un disco, un segmento en un disco, un archivo o un volumen de disco.
VLAN	Red de área local virtual
vldc	Servicio de canal de dominio lógico virtual
vldcc	Cliente del canal del dominio lógico virtual
vnet	Un dispositivo de red virtual implementa y un dispositivo Ethernet virtual y se comunica con otros dispositivos vnet en el sistema usando el conmutador de red virtual (vswi tch)
vntsd	Daemon del servidor terminal de red virtual del SO 10 de Solaris de Oracle para las consolas de Dominios lógicos (véase la página de comandos man vntsd(1M)).
volfs	Sistema de archivos de Volume Management (véase la página de comandos man volfs(7FS)).

vsw, vswitch	Conmutador de red virtual que conecta los dispositivos de red virtual a la red externa e intercambia paquetes entre ellos
VTOC	Índice de contenido de volumen
VxDMP	Multirruta dinámica de Veritas
VxVM	Administrador del volumen de Veritas

W

WAN	Red de área amplia
------------	--------------------

X

XFP	eXtreme Fast Path
XML	Lenguaje de marcas extensible
XMPP	Protocolo extensible de mensajería y comunicación de presencia

Z

ZFS	Sistema de archivos Zettabyte (SO 10 Solaris de Oracle)
zpool	Grupo de almacenamiento ZFS (véase la página de comandos <code>man zpool(1M)</code>).
ZVOL	Controlador de emulación de volumen ZFS

Índice

Números y símbolos

, definición, 17

A

administración de energía (PM), 174

administración de energía de la memoria (PM), 174

Administrador de Dominios lógicos, 18, 19

 daemon (ldmd), 21

 esquema XML usado con, 233

 esquemas XML usados con, 257

 mecanismo de descubrimiento, 229

asignación, de un dispositivo de punto final a un dominio E/S, 69–79

asignación de, un bus PCIe a un dominio E/S, 64–68

autorización

 lectura, 41

 lectura y escritura, 41

 niveles, 41

autorizaciones, ldm subcomandos, 41

B

bus PCI EXPRESS (PCIe), 63–64

C

canal de dominio lógico (LDC), 19

cancel-operation reconf, 160

CLI, *Ver* interfaz de línea de comandos

comandos

 ldm(1M), 21

 ldmconfig(1M), 23, 223

 ldmp2v(1M), 210

commands, ldmconfig(1M), 225

configuración

 almacenamiento en el comando del procesador de servicios, 160

 límite de energía, 174

 marcos jumbo, 141–144

 seleccionando para iniciar, 22

controlador de sistema, *Ver* procesador de servicios (SP)

crear dominio E/S, bus PCIe completo, 65

D

daemon de reconfiguración dinámica (drd), 160

daemon del servidor terminal de la red virtual (vntsd), 22

daemons

 drd, 160

 ldmd, 21

 vntsd, 22

definición de, dominios lógicos, 17

deshabilita núcleo de CPU, 174

dispositivos físicos, 20, 21

dispositivos virtuales

 cliente de disco virtual (vdc), 22

 concentrador de la consola virtual (vcc), 22

 conmutador virtual (vsw), 21

dispositivos virtuales (*Continuación*)

- E/S, 21
 - red virtual (vnet), 21
 - servicio de disco virtual (vds), 22
- dominio, tipos de, 20
- dominio de control, 20
- dominio de servicios, 20, 21
- dominio E/S, 63–64, 64–68, 69–79
- asignación de un bus PCIe, 64–68
 - asignación de un dispositivo de punto final, 69–79
 - bus PCI EXPRESS (PCIe), 63–64
 - creación, 65
 - limitaciones de migración, 64
- dominio primary, 20
- reinicio, 73–74
- dominio raíz, 20
- dominios
- de servicios, 21
 - tipos de, 20
- dominios huésped, 20
- dominios lógicos, funciones, 20
- DR, *Ver* reconfiguración dinámica

E

- E/S directa (DIO), planificación, 72
- equipo físico, 19
- equipo virtual, 19
- esquema XML
 - Administrador de Dominios lógicos usado con, 233
 - Administrador de Dominios lógicos usados con, 257

F

- funciones, dominios lógicos, 20

G

- GUI del asistente para la configuración, 223

I

- interfaz de línea de comandos, 21
- IPMP basado en enlace, uso de, 128–131

L

- LDC, *Ver* canal de dominio lógico
- ldm, ls-dom, 160
- ldm subcomandos
 - autorizaciones del usuario, 41
 - cancel-operation reconf, 160
- ldm(1M) página de comando man, 21
- ldm(1M) comando, 21
- ldmconfig(1M) comando, 23, 223
- ldmconfig(1M) command, 225
- ldmd, daemon del Administrador de Dominios lógicos, 21
- ldmp2v(1M) comando, 210
- lectura, autorización, 41
- lectura y escritura, autorización, 41
- limitaciones de migración, dominio E/S, 64
- límite de energía, 174
- ls-dom subcommand, 160

M

- marcos jumbo, configuración, 141–144
- migración, no interactiva, 156
- migración de dominio, no interactiva, 156
- migración de dominio no interactiva, 156

O

- Omisión de ciclo de reloj de CPU, 174

P

- paquetes, SUNWldm, 21
- planificación
 - E/S directa (DIO), 72
 - E/S directa E/S (DIO), 72

plataformas

- servidor SPARC T3, 21
- servidor UltraSPARC T2 Plus, 21

procesador de servicios (SP)

- almacenamiento de configuraciones, 160
- efectúa un seguimiento y ejecuta los equipos físicos, 19

R

- reconfiguración dinámica (DR), 159
- reconfiguración dinámica de memoria (DR), 166
- reconfiguración retrasada, 160
- recursos
 - Ver también* dispositivos virtuales
 - definición, 19
- reinicio del dominio primary, 73–74
- ruta múltiple, disco virtual, 94
- ruta múltiple de disco virtual, 94

S

- servidor SPARC T3, 21
- servidor UltraSPARC T2 Plus, 21
- SUNWldm paquete, 21

U

- Uso de IPMP basado en enlace, 128–131

