

Oracle® VM Server for SPARC 2.0 リファレンスマニュアル

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

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

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

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.

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

Oracle と Java は Oracle Corporation およびその関連企業の登録商標です。その他の名称は、それぞれの所有者の商標または登録商標です。

AMD、Opteron、AMD ロゴ、AMD Opteron ロゴは、Advanced Micro Devices, Inc. の商標または登録商標です。Intel、Intel Xeon は、Intel Corporation の商標または登録商標です。すべての SPARC の商標はライセンスをもとに使用し、SPARC International, Inc. の商標または登録商標です。UNIX は X/Open Company, Ltd. からライセンスされている登録商標です。

このソフトウェアまたはハードウェア、そしてドキュメントは、第三者のコンテンツ、製品、サービスへのアクセス、あるいはそれらに関する情報を提供することがあります。オラクル社およびその関連会社は、第三者のコンテンツ、製品、サービスに関して一切の責任を負わず、いかなる保証もいたしません。オラクル社およびその関連会社は、第三者のコンテンツ、製品、サービスへのアクセスまたは使用によって損失、費用、あるいは損害が発生しても一切の責任を負いかねます。

目次

システム管理	5
ldm(1M)	6
ldmconfig(1M)	56
ldmd(1M)	57
ldmp2v(1M)	59
virtinfo(1M)	66
拡張ライブラリ関数	69
v12n(3EXT)	70
ライブラリのインタフェースとヘッダー	73
libv12n(3LIB)	74

参 照

システム管理

名前 ldm – Logical Domains Manager 用のコマンド行インタフェース

形式

```

ldm or ldm --help [subcommand]
ldm -V
ldm add-domain -i file
ldm add-domain [mac-addr=num] [hostid=num]
    [failure-policy=ignore|panic|reset|stop] [master=master-ldom1,...,master-ldom4] ldom
ldm add-domain ldom...
ldm set-domain -i file
ldm set-domain [mac-addr=num] [hostid=num]
    [failure-policy=ignore|panic|reset|stop] [master=[master-ldom1,...,master-ldom4]] ldom
ldm remove-domain -a
ldm remove-domain ldom...
ldm list-domain [-e] [-l] [-o format] [-p] [ldom...]
ldm migrate-domain [-f] [-n] [-p filename] source-ldom [user@]target-host[:target-ldom]
ldm add-vcpu [-c] number ldom
ldm set-vcpu [-c] number ldom
ldm remove-vcpu [-c] number ldom
ldm add-crypto number ldom
ldm set-crypto number ldom
ldm remove-crypto number ldom
ldm add-memory [--auto-adj] size[unit] ldom
ldm set-memory [--auto-adj] size[unit] ldom
ldm remove-memory [--auto-adj] size[unit] ldom
ldm start-reconf ldom
ldm cancel-reconf ldom
ldm cancel-operation (migration | reconf | memdr) ldom
ldm add-io bus-or-device ldom
ldm remove-io bus-or-device ldom
ldm list-io [-l] [-p]
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
ldm set-vsw [pvid=port-vlan-id] [vid=vlan-id1,vlan-id2,...] [mac-addr=num] [net-dev=device]
    [linkprop=[phys-state]] [mode=[sc]] [mtu=size] vswitch-name
ldm remove-vsw [-f] vswitch-name
ldm add-vnet [mac-addr=num] [mode=hybrid] [pvid=port-vlan-id] [vid=vlan-id1,vlan-id2,...]
    [linkprop=phys-state] [id=network-id] [mtu=size] if-name vswitch-name ldom
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
ldm remove-vnet [-f] if-name ldom
ldm add-vds service-name ldom
ldm remove-vds [-f] service-name
ldm add-vdsdev [-f] [options={ro,slice,excl}] [mpgroup=mpgroup] backend
    volume-name@service-name
ldm set-vdsdev [-f] options=[{ro,slice,excl}] [mpgroup=mpgroup]
    volume-name@service-name
ldm remove-vdsdev [-f] volume-name@service-name
ldm add-vdisk [timeout=seconds] [id=disk-id] disk-name volume-name@service-name ldom

```

```

ldm set-vdisk [timeout=seconds] [volume=volume-name@service-name] disk-name ldom
ldm remove-vdisk [-f] disk-name ldom
ldm add-vdpcs vdpcs-service-name ldom
ldm remove-vdpcs [-f] vdpcs-service-name
ldm add-vdpcc vdpcc-name vdpcs-service-name ldom
ldm remove-vdpcc [-f] vdpcc-name ldom
ldm add-vcc port-range=x-y vcc-name ldom
ldm set-vcc port-range=x-y vcc-name
ldm remove-vcc [-f] vcc-name
ldm set-vcons [port=[port-num]] [group=group] [service=vcc-server] ldom
ldm add-variable var-name=[value]... ldom
ldm set-variable var-name=[value]... ldom
ldm remove-variable var-name... ldom
ldm list-variable [var-name...] ldom
ldm start-domain (-a | -i file | ldom...)
ldm stop-domain [-f] (-a | ldom...)
ldm panic-domain ldom
ldm bind-domain (-i file | ldom)
ldm unbind-domain ldom
ldm list-bindings [-e] [-p] [ldom...]
ldm add-spconfig config-name
ldm add-spconfig -r autosave-name [new-config-name]
ldm set-spconfig config-name
ldm set-spconfig factory-default
ldm remove-spconfig [-r] config-name
ldm list-spconfig [-r [autosave-name]]
ldm list-constraints ([-x] | [-e] [-p]) [ldom...]
ldm list-devices [-a] [-p] [core] [cpu] [crypto] [memory] [io]
ldm list-services [-e] [-p] [ldom...]
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
ldm init-system [-rs] -i file

```

機能説明

ldm コマンドは、Logical Domains Manager と相互に作用し、論理ドメインの作成および管理に使用されます。Logical Domains Manager は、サーバーごとに 1 つだけ存在できます。Logical Domains Manager は制御ドメインに対して実行します。これは、サービスプロセッサにより作成される初期ドメインです。制御ドメインの名前は primary です。

論理ドメインは、独自のオペレーティングシステム、資源、および単一のコンピュータシステム内での識別情報を持つ個別の論理グループです。各論理ドメインは、サーバーの電源の再投入を必要とせずに、作成、削除、再構成、および再起動

を単独で行うことができます。セキュリティ上の理由から、論理ドメインを使用してさまざまなアプリケーションを異なるドメインで動作させて、アプリケーションの独立性を維持することができます。

論理ドメインはすべて同じですが、論理ドメインに対して指定する役割に基づいてそれぞれ区別できます。論理ドメインが実行できる役割は、次のとおりです。

制御ドメイン	ハイパーバイザと通信することによって、ほかの論理ドメインおよびサービスを作成および管理します。
サービスドメイン	仮想ネットワークスイッチ、仮想ディスクサービスなどのサービスをほかの論理ドメインに提供します。
I/O ドメイン	<p>PCI EXPRESS (PCIe) コントローラ内のネットワークカードなどの、物理 I/O デバイスに直接アクセスできます。I/O ドメインは、PCIe ルートコンプレックスを所有するか、静的ダイレクト I/O (Static Direct I/O (SDIO)、SDIO) 機能を使用して PCIe スロットまたはボード上の PCIe デバイスを所有できます。</p> <p>I/O ドメインは、I/O ドメインがサービスドメインとしても使用される場合に、仮想デバイスの形式でほかのドメインと物理 I/O デバイスを共有できます。</p>
ルートドメイン	<p>PCIe ルートコンプレックスが割り当てられます。このドメインは、PCIe ファブリックとすべての接続されているデバイスを所有し、ファブリックのエラー処理などのファブリック関連のサービスをすべて提供します。ルートドメインは I/O ドメインでもあり、物理 I/O デバイスを所有し、それらに直接アクセスできます。</p> <p>保持できるルートドメインの数は、プラットフォームアーキテクチャーによって決まります。たとえば、Sun SPARC Enterprise T5440 サーバーを使用している場合、最大で 4 つのルートドメインを保持できます。</p>
ゲストドメイン	I/O ドメインおよびサービスドメインのサービスを使用し、制御ドメインによって管理されます。

Logical Domains Manager を使用して、ドメイン間の依存関係を確立できます。

マスタードメイン	1 つ以上のドメインが依存しているドメインです。マスタードメインは、マスタードメインに障害が発生した場合にそのスレーブドメインによって適用される障害ポリシーを指定します。たとえば、マスタードメインに障害が発生した場合、マスタードメインの障害ポリシーに基づき、スレーブドメインに対して、放置、パニックの発生、再起動、または停止を行うことができます。
----------	---

スレーブドメイン 他のドメインに依存しているドメインです。ドメインは、1つ以上のマスタードメインに障害が発生した場合に障害ポリシーを適用するように指示するマスタードメインを、最大4つ指定できます。

サブコマンドの要約

サポートされているサブコマンドとその説明および各サブコマンドに必要な権限は、次のとおりです。ユーザーアカウントの承認の設定については、『[Oracle VM Server for SPARC 2.0 管理ガイド](#)』の「[ユーザーアカウントに対する承認およびプロファイルの作成と役割の割り当て](#)」を参照してください。

サブコマンド	説明	承認
add-spconfig	論理ドメイン構成をサービスプロセッサ (SP) に追加します。	solaris.ldoms.write
add-domain	論理ドメインを作成します。	solaris.ldoms.write
add-policy	既存の論理ドメインに資源管理ポリシーを追加します。	solaris.ldoms.write
add-resource	既存の論理ドメインに資源を追加します。資源の定義については、「資源」を参照してください。	solaris.ldoms.write
bind-domain	作成された論理ドメインに資源をバインドします。	solaris.ldoms.write
cancel-operation	遅延再構成 (reconf)、メモリー動的再構成 (Dynamic Reconfiguration、DR) 削除 (memdr)、またはドメイン移行 (migration) などの処理を取り消します。	solaris.ldoms.write
cancel-reconf	primary ドメインの遅延再構成処理を取り消します。	solaris.ldoms.write
list-domain	論理ドメインおよびその状態のリストを表示します。	solaris.ldoms.read
list-type	論理ドメインのバインド、制約、デバイス、サービス、構成など、サーバー資源を表示します。	solaris.ldoms.read
list-variable	論理ドメインの変数のリストを表示します。	solaris.ldoms.read
migrate-domain	マシン間で論理ドメインを移行します。	solaris.ldoms.write
panic-domain	指定された論理ドメインの Oracle Solaris OS でパニックを発生させます。	solaris.ldoms.write

サブコマンド	説明	承認
<code>remove-spconfig</code>	サービスプロセッサから論理ドメイン構成を削除します。	<code>solaris.ldoms.write</code>
<code>remove-domain</code>	論理ドメインを削除します。	<code>solaris.ldoms.write</code>
<code>remove-policy</code>	既存の論理ドメインから資源管理ポリシーを削除します。	<code>solaris.ldoms.write</code>
<code>remove-resource</code>	既存の論理ドメインから資源を削除します。資源の定義については、「資源」を参照してください。	<code>solaris.ldoms.write</code>
<code>remove-variable</code>	既存の論理ドメインから1つ以上の変数を削除します。	<code>solaris.ldoms.write</code>
<code>set-spconfig</code>	使用する論理ドメイン構成を指定します。	<code>solaris.ldoms.write</code>
<code>set-domain</code>	論理ドメインにプロパティを設定します。	<code>solaris.ldoms.write</code>
<code>set-policy</code>	既存の論理ドメインに資源管理ポリシーのプロパティを設定します。	<code>solaris.ldoms.write</code>
<code>set-resource</code>	既存の論理ドメインに対して資源を指定します。これは、プロパティの変更または数量の変更のいずれかに使用できます。資源 <code>vcpu</code> 、 <code>memory</code> 、または <code>crypto</code> に適用する場合、これは数量の変更を表します。数量の変更の場合、このサブコマンドは動的または遅延再構成処理となり、指定した資源の数量が、指定した論理ドメインに割り当てられます。論理ドメインに割り当てられている資源の数がこのサブコマンドの指定よりも多い場合は、いくつかの資源が削除されます。論理ドメインに割り当てられている資源の数がこのサブコマンドの指定よりも少ない場合は、いくつかの資源が追加されます。資源の定義については、「資源」を参照してください。	<code>solaris.ldoms.write</code>
<code>set-variable</code>	既存の論理ドメインに1つ以上の変数を設定します。	<code>solaris.ldoms.write</code>
<code>start-domain</code>	1つ以上の論理ドメインを起動します。	<code>solaris.ldoms.write</code>
<code>start-reconf</code>	<code>primary</code> ドメイン上で遅延再構成モードに入ります。	<code>solaris.ldoms.write</code>

サブコマンド	説明	承認
stop-domain	1つ以上の動作中の論理ドメインを停止します。	solaris.ldoms.write
unbind-domain	論理ドメインから、資源のバインドを解除 (資源を解放) します。	solaris.ldoms.write

注- すべてのサブコマンドが、すべての資源タイプでサポートされているわけではありません。

別名

次の表に、ldm サブコマンドの3種類の別名を示します。

別名の種類	省略形式	長文形式
処理の別名 (動詞)	ls	list
処理の別名 (動詞)	rm	remove
資源の別名 (名詞)	config	spconfig
資源の別名 (名詞)	crypto	mau
資源の別名 (名詞)	dom	domain
資源の別名 (名詞)	mem	memory
資源の別名 (名詞)	var	variable
資源の別名 (名詞)	vcc	vconscon
資源の別名 (名詞)	vcons	vconsole
資源の別名 (名詞)	vdpc	ndpsldcc
資源の別名 (名詞)	vdpcs	ndpsldcs
資源の別名 (名詞)	vds	vdiskserver
資源の別名 (名詞)	vdsdev	vdiskserverdevice
資源の別名 (名詞)	vsw	vswitch
サブコマンドのショートカット	bind	bind-domain
サブコマンドのショートカット	cancel-op	cancel-operation
サブコマンドのショートカット	create	add-domain

別名の種類	省略形式	長文形式
サブコマンドのショートカット	destroy	remove-domain
サブコマンドのショートカット	list	list-domain
サブコマンドのショートカット	migrate	migrate-domain
サブコマンドのショートカット	modify	set-domain
サブコマンドのショートカット	panic	panic-domain
サブコマンドのショートカット	start	start-domain
サブコマンドのショートカット	stop	stop-domain
サブコマンドのショートカット	unbind	unbind-domain

注- このマニュアルページの以降の構文および例では、省略形式の処理の別名および資源の別名を使用します。

資源	次の資源がサポートされています。
io	PCIe ルートコンプレックスなどの I/O デバイスと、それらに接続されているアダプタとデバイス。ダイレクト I/O 割り当て可能デバイスでもあります。
crypto	サポートされているサーバー上でサポートされている暗号化装置。現在、モジュラー演算ユニット (MAU) と Control Word Queue (CWQ) の 2 つの暗号化装置がサポートされています。
mem、memory	バイト単位のデフォルトのメモリーサイズ。つまり、G バイト (G)、K バイト (K)、または M バイト (M) を指定します。ゲストドメインに割り当てることができる、サーバーの仮想化されたメモリーです。
vcc、vconscon	ゲストドメインの作成時に各ゲストドメインに割り当てするための特定範囲の TCP ポートを持つ、仮想コンソール端末集配信装置 (コンセントレータ) サービス。

vcons、vconsole	システムレベルのメッセージにアクセスするための仮想コンソール。接続は、特定のポートで制御ドメイン上の vconscon サービスに接続することによって実現します。
vcpu	各仮想 CPU は、サーバーの 1 つの CPU スレッドを表します。たとえば、8 コアの Sun SPARC Enterprise T5120 サーバーには、論理ドメイン間で割り当てることができる 64 の仮想 CPU があります。
vdisk	仮想ディスクは、さまざまな種類の物理デバイス、ボリューム、またはファイルで構成される総称的なブロック型デバイスです。仮想ディスクは SCSI ディスクと同義ではありません。そのため、ディスク名内のターゲット ID (tN) は除外されます。論理ドメインの仮想ディスクの形式は、cNdNsN です。cN は仮想コントローラ、dN は仮想ディスク番号、および sN はスライスを示します。
vds、vdiskserver	ほかの論理ドメインに仮想ディスクをエクスポートできる仮想ディスクサーバー。
vdsdev、vdiskserverdevice	仮想ディスクサーバーがエクスポートしたデバイス。このデバイスには、ディスク全体、ディスクのスライス、ファイル、またはディスクボリュームを指定できます。
vdppc	仮想データプレーンのチャンネルクライアント。Netra Data Plane Software (NDPS) 環境でのみ使用されます。
vdpcs	仮想データプレーンのチャンネルサービス。Netra Data Plane Software (NDPS) 環境でのみ使用されます。
vnet	仮想 Ethernet デバイスを実装し、仮想ネットワークスイッチ (vsw) を使用するシステム内のほかの vnet デバイスと通信する仮想ネットワークデバイス。
vsw、vswitch	仮想ネットワークデバイスを外部ネットワークに接続し、仮想ネットワークデバイス間でのパケットの切り替えも行う仮想ネットワークスイッチ。

リストの種類

次の種類のリストがサポートされています。

bindings 論理ドメインにバインドされている資源のリスト。

config	サービスプロセッサ (Service Processor、SP) 上に格納される論理ドメインの構成を一覧表示します。Oracle の Sun UltraSPARC T3 ベースのサーバーでは、作成日の最も古いものから新しいものへという順番で構成が表示されます。
constraints	論理ドメインの作成に使用される制約のリスト。
devices	サーバー用の使用されていないすべてのデバイスの一覧表示。
services	論理ドメインによってエクスポートされるすべてのサービスの一覧表示。

オプション 次の表に、ldm コマンドのオプションを示します。適用可能な場合、オプションの省略形式に続いて長形式を示します。

-a	--all	すべての種類のオペランドで動作します。
	--auto-adj	<p>add-memory、set-memory、および remove-memory サブコマンドが、256M バイトの境界でのメモリーの変更を揃えることを指定します。--auto-adj オプションの動作は、影響を受けるドメインの状態によって決まります。</p> <ul style="list-style-type: none">■ アクティブなドメイン。「動的再構成」の場合、このオプションは、256M バイト境界で追加または削除されるメモリーの量を揃えます。量は add-memory 操作の場合は切り上げられ、remove-memory 操作の場合は切り下げられます。set-memory 操作は、add-memory 操作または remove-memory 操作として扱われます。これらのサブコマンドのいずれかに対して、--auto-adj 操作は、ドメインのメモリーの「結果として得られるサイズ」が要求されたサイズより大きいか等しくなるようにします。「遅延再構成」の場合、このオプションは境界または非アクティブドメインと同じ動作になります。遅延再構成は次の状況下で発生します。<ul style="list-style-type: none">■ ドメインが遅延再構成を開始する。■ 遅延再構成がドメイン内で未処理になっている。■ 境界ドメインまたは非アクティブドメイン。このオプションは、次の 256M バイトの境界まで切り上げることにより、ドメインの結果として得られるサイズを揃えます。この位置揃えは、add-memory、set-memory、および remove-memory 操作で発生します。

<code>-c number</code>	<code>--core number</code>	論理ドメインに対してコア全体の割り当てが実行されることを指定します。 <i>number</i> は、割り当てられるコア全体の数です。
<code>-e</code>	<code>--extended</code>	自動的に設定されるサービスおよびデバイスを含む拡張リストを生成します。これは制御できません。
<code>-f</code>	<code>--force</code>	処理を強制的に試行します。
<code>-i file</code>	<code>--input file</code>	論理ドメインの作成に使用する XML 構成ファイルを指定します。
<code>-l</code>	<code>--long</code>	長いリストを生成します。
<code>-n</code>	<code>--dry-run</code>	移行が成功するかどうかを確認するために、移行の予行演習を行います。ドメインの移行が実際に行われるわけではありません。
<code>-o format</code>	<code>--output format</code>	表示する内容に応じて、 <code>ldm list</code> コマンドの形式を指定します。 <code>console</code> 、 <code>core</code> 、 <code>cpu</code> 、 <code>crypto</code> 、 <code>disk</code> 、 <code>domain</code> 、 <code>memory</code> 、 <code>network</code> 、および <code>status</code> のうち 1 つ以上を指定します。複数の形式を指定する場合は、各形式をコンマで区切ります。スペースは入れません。
<code>-p</code>	<code>--parseable</code>	マシンが読み取り可能なバージョンの出力を生成します。
<code>-r</code>		<code>add-sponfig</code> 、 <code>list-sponfig</code> 、および <code>remove-sponfig</code> サブコマンドの場合: 手動の構成回復を実行します。
<code>-r</code>	<code>--reboot</code>	<code>init-system</code> サブコマンドの場合: 構成後にシステムを再起動します。
<code>-s</code>	<code>--services-only</code>	仮想サービス構成のみを復元します (<code>vds</code> 、 <code>vcc</code> 、および <code>vsw</code>)。
<code>-x file</code>	<code>--xml file</code>	論理ドメインの制約を含む XML ファイルを標準出力 (<code>stdout</code>) に書き込むことを指定します。バックアップファイルとして使用できます。
<code>-V</code>	<code>--version</code>	バージョン情報を表示します。
	<code>--help</code>	使用法の説明を表示します。

Properties

注 - さまざまな `ldm set-*` コマンドで、空の値を指定してプロパティをデフォルト値にリセットすることができます。たとえば、次の `ldm set-policy` コマンドを実行すると、`attack` プロパティをデフォルト値にリセットできます。

```
# ldm set-policy attack= high-policy ldom1
```

次の種類のプロパティがサポートされています。

<code>attack=value</code>	いずれか1つの資源制御サイクル中に追加する資源の最大量を指定します。使用可能な資源がこの量よりも少ない場合は、使用可能な資源がすべて追加されます。使用可能な仮想CPUをできるだけ多く追加できるよう、デフォルトでは <code>attack</code> に制限はありません。有効な値は、1から、システムで使用していない仮想CPUの数までです。
<code>decay=value</code>	いずれか1つの資源制御サイクル中に削除する資源の最大量を指定します。このプロパティで指定されている値のほうが大きい場合でも、現在バインドされている仮想CPUの数から <code>vcpu-min</code> の値を引いた数だけが削除できます。デフォルトでは、値は1です。有効な値は、1から、仮想CPUの合計数より1少ない数までです。
<code>default-vlan-id=</code>	仮想ネットワークデバイスまたは仮想スイッチをメンバーにする必要があるデフォルトの仮想ローカルエリアネットワーク(VLAN)を、タグ付きモードで指定します。最初のVLAN ID(<i>vid1</i>)は、この <code>default-vlan-id</code> に予約されています。
<code>elastic-margin=value</code>	<code>util-lower</code> と使用していない仮想CPU数の間のバッファ量を指定して、仮想CPUの数を減らした場合の変動を回避します。有効な値は、0から100までです。デフォルト値は5です。
<code>enable=yes no</code>	個々のドメインの資源管理を有効または無効にします。デフォルトでは、 <code>enable=yes</code> です。
<code>failure-policy=</code>	<p>マスタードメインの障害ポリシーを指定します。このポリシーは、マスタードメインに障害が発生した場合のスレーブドメインの動作を制御します。このプロパティは、マスタードメインで設定します。デフォルト値は<code>ignore</code>です。次に、有効なプロパティ値を示します。</p> <ul style="list-style-type: none">■ <code>ignore</code> は、マスタードメインの障害を無視します。スレーブドメインは影響を受けません。■ <code>panic</code> は、マスタードメインに障害が発生した場合、すべてのスレーブドメインにパニックを発生させます。■ <code>reset</code> は、マスタードメインに障害が発生した場合、すべてのスレーブドメインをリセットします。■ <code>stop</code> は、マスタードメインに障害が発生した場合、すべてのスレーブドメインを停止します。

group=	コンソールを接続するグループを指定します。グループ引数を使用すると、同一のTCP接続上で複数のコンソールを多重化できます。
hostid=	特定のドメインのホストIDを指定します。ホストIDを指定しない場合、Logical Domains Managerにより一意のホストIDが各ドメインに割り当てられます。
id=	新しい仮想ディスクデバイス、仮想ネットワークデバイス、および仮想スイッチデバイスのIDをそれぞれ指定します。
linkprop=phys-state	配下の物理ネットワークデバイスに基づいて、仮想デバイスがリンクステータスをレポートするかどうかを指定します。コマンドラインでlinkprop=phys-stateを指定すると、仮想デバイスのリンクステータスは物理リンクステータスを反映します。デフォルトでは、仮想デバイスのリンクステータスは物理リンクステータスを反映しません。
mac-addr=	MACアドレスを定義します。番号は、標準のオクテット記述法で指定する必要があります。たとえば、80:00:33:55:22:66とします。
master=	1つのスレーブドメインに対して最大4つのマスタードメインの名前を指定します。このプロパティは、スレーブドメインで設定します。デフォルトでは、ドメインに対してマスタードメインは設定されていません。ldm add-domain 操作の前に、そのドメインが存在している必要があります。 注 - Logical Domains Manager では、依存サイクルが生じるドメイン関係を作成することはできません。
mode=	add-vsw および set-vsw サブコマンドの場合: ゲストドメインで Oracle Solaris クラスタソフトウェアが動作していない場合は、仮想ネットワークのパフォーマンスに影響を与える可能性があるため、このオプションを指定しないでください。

そうでない場合は、次のいずれかを指定します。

- Logical Domains 環境で &ClusterPod; ハートビートパケットの処理を優先順位付けするために、仮想ネットワークのサポートを有効にするには、`mode=sc` を指定します。
- ハートビートパケット用の特別な処理を停止するには、`set-vsw` サブコマンドの `mode=` 引数を空白のままにします。

`add-vnet` および `set-vnet` サブコマンドの場合:

NIU ハイブリッド I/O を使用しない場合は、このオプションを省略します。

そうでない場合は、次のいずれかを指定します。

- `mode=hybrid` を設定して、可能な場合には NIU ハイブリッド I/O を使用するようにシステムに要求します。可能でない場合には、システムにより仮想 I/O に戻されます。『[Oracle VM Server for SPARC 2.0 管理ガイド](#)』の「[NIU ハイブリッド I/O の使用](#)」を参照してください。
- NIU ハイブリッド I/O を無効にするには、`set-vnet` サブコマンドの `mode=` 引数を空白のままにします。

`mpgroup=`

複数の仮想ディスクサーバーデバイス (`vdsdev`) のマルチパスグループ名を定義します。したがって、仮想ディスクが仮想ディスクサーバーデバイスと通信できない場合、マルチパスグループ内のほかの仮想ディスクサーバーデバイスに対してフェイルオーバーが開始されます。

`mtu=`

仮想スイッチ、その仮想スイッチにバインドされている仮想ネットワークデバイス、またはその両方の最大転送単位 (MTU) を指定します。有効な値の範囲は 1500 潤ち 16000 です。無効な値を指定すると、`ldm` コマンドでエラーが発生します。

`name=policy-name`

資源管理ポリシー名を指定します。

`net-dev=`

実際のネットワークデバイスのパス名を定義します。

`options=`

特定の仮想ディスクサーバーデバイスに、次のオプションのすべてまたはサブセットを指定します。2 つ以上のオプションは、空白文字を入れずにコンマで区切って、`ro,slice,excl` のように指定します。

- `ro` - 読み取り専用アクセスを指定

- `slice` - 単一スライスのディスクとしてバックエンドをエクスポート
- `excl` - 排他的なディスクアクセスを指定

`add-vdsdev` サブコマンドの `options=` 引数を指定しないか空白のままにすると、デフォルト値であるディスク、非排他的、および読み取り/書き込みになります。以前に指定したオプションをオフにするには、`set-vdsdev` サブコマンドの `options=` 引数を空白のままにします。

<code>port=</code>	特定のポート番号を指定するか、空白のままにして、Logical Domains Manager によるポート番号の設定を可能にします。
<code>port-range=</code>	TCP ポートの範囲を定義します。
<code>priority=value</code>	ポリシーの実効時間が重複する場合に、選択するポリシーを決めます。数値が低いほど、優先度は高く(良く)なります。有効な値は、1 から 9999 までです。デフォルト値は 99 です。
<code>pvid=</code>	仮想ネットワークデバイスをメンバーにする必要のある VLAN をタグなしモードで指定します。
<code>sample-rate=value</code>	サイクル時間を秒単位で指定します。有効な値は、1 から 10 までです。推奨値はデフォルトの 10 です。
<code>service=</code>	コンソール接続を処理する既存の仮想コンソール端末集配信装置の名前を指定します。
<code>timeout=</code>	仮想ディスククライアント (<code>vdc</code>) と仮想ディスクサーバー (<code>vds</code>) の間の接続を確立する際の秒数を定義します。複数の仮想ディスク (<code>vdisk</code>) パスがある場合、 <code>vdc</code> は、別の <code>vds</code> への接続を試みることができます。また、タイムアウトによって、いずれかの <code>vds</code> への接続が指定の時間内に確実に行われます。0 を指定すると <code>set-vdisk</code> サブコマンドのタイムアウトは無効になります。
<code>tod-begin=hh:mm[:ss]</code>	ポリシーの有効開始時刻を指定します。単位は時、分、秒 (オプション) です。デフォルト値は 00:00:00 です。
<code>tod-end=hh:mm[:ss]</code>	ポリシーの有効停止時刻を指定します。単位は時、分、秒 (オプション) です。デフォルト値は 23:59:59 です。
<code>util-lower=percent</code>	ポリシー分析がトリガーされる使用率の下限レベルを指定します。有効な値は、1 から、 <code>util-upper</code> より 1 少ない数までです。デフォルト値は 60 です。

<code>util-upper=percent</code>	ポリシー分析がトリガーされる使用率の上限レベルを指定します。有効な値は、 <code>util-lower</code> に 1 を足した数から 99 までです。デフォルト値は 85 です。
<code>vcpu-max=value</code>	ドメインの仮想 CPU 資源の最大値を指定します。デフォルトでは、仮想 CPU の最大数に制限はありません。有効な値は、 <code>vcpu-min</code> に 1 を足した数から、システムで使用していない仮想 CPU の合計数までです。
<code>vcpu-min=value</code>	ドメインの仮想 CPU 資源の最小値を指定します。有効な値は、1 から、 <code>vcpu-max</code> より 1 少ない数までです。デフォルト値は 1 です。
<code>vid=</code>	仮想ネットワークデバイスまたは仮想スイッチをメンバーにする必要がある VLAN を、タグ付きモードで指定します。
<code>volume=</code>	仮想ディスクのボリューム名を変更します。
<code>vswitch=</code>	仮想ネットワークの仮想スイッチ名を変更します。

`list` サブコマンド
出力内のフラグ

次に、`list` サブコマンド出力内のフラグの定義を示します。

- 可変部分
- c 制御ドメイン
- d 遅延再構成
- e エラー
- n 通常
- r 進行中のメモリー DR
- s 列 1 - 起動または停止
列 6 - ソースドメイン
- t 列 2 - 切り替え
列 6 - ターゲットドメイン
- v 仮想 I/O サービスドメイン

リストフラグ値は位置に依存します。次に、左から順に 5 つの列のそれぞれに表示される可能性のある値を示します。

表1 リストフラグの位置

列1	列2	列3	列4	列5	列6
s または -	n または t	d、r、または -	c または -	v または -	s、t、または e

サブコマンドの 使用法

この節では、サポートされているコマンド行インタフェース (CLI) のすべての処理、つまり、すべてのサブコマンドと資源の組み合わせについて説明します。

ドメインの追 加、設定、削 除、および移行

ドメインの追加

このサブコマンドは、1つ以上の論理ドメイン名を指定するかまたは XML 構成ファイルを使用して、1つ以上の論理ドメインを追加します。MAC アドレス、ホスト ID、マスタートメインのリスト、障害ポリシーなど、ドメインをカスタマイズするためのプロパティ値を指定することもできます。これらのプロパティ値を指定しない場合、Logical Domains Manager により自動的にデフォルト値が割り当てられます。

```
ldm add-dom -i file
ldm add-dom [mac-addr=num] [hostid=num]
[failure-policy=ignore|panic|reset|stop] [master=master-ldom1,...,master-ldom4] ldom
ldm add-dom ldom...
```

ここでは、次のように指定します。

- `-i file` は、論理ドメインの作成に使用する XML 構成ファイルを指定します。
- `mac-addr=num` は、このドメインの MAC アドレスです。番号は、標準のオクテット記述法で指定する必要があります。たとえば、`80:00:33:55:22:66` とします。
- `hostid` は、特定のドメインのホスト ID を指定します。ホスト ID を指定しない場合、Logical Domains Manager により一意のホスト ID が各ドメインに割り当てられます。
- `failure-policy` は、マスタートメインの障害ポリシーを指定します。このポリシーは、マスタートメインに障害が発生した場合のスレーブドメインの動作を制御します。このプロパティは、マスタートメインで設定します。デフォルト値は `ignore` です。次に、有効なプロパティ値を示します。
 - `ignore` は、マスタートメインの障害を無視します。スレーブドメインは影響を受けません。
 - `panic` は、マスタートメインに障害が発生した場合、すべてのスレーブドメインにパニックを発生させます。
 - `reset` は、マスタートメインに障害が発生した場合、すべてのスレーブドメインをリセットします。

- **stop** は、マスタートメインに障害が発生した場合、すべてのスレーブドメインを停止します。
- **master** には、1つのスレーブドメインに対して最大4つのマスタートメインの名前を指定します。このプロパティは、スレーブドメインで設定します。デフォルトでは、ドメインに対してマスタートメインは設定されていません。ldm add-domain 操作の前に、マスタートメインが存在している必要があります。

注 - Logical Domains Manager では、依存サイクルが生じるドメイン関係を作成することはできません。

- **ldom** は、追加する論理ドメインを指定します。

ドメインのオプションの設定

このサブコマンドでは、各ドメインの **mac-addr**、**hostid**、**failure-policy**、および **master** プロパティを変更できます。

注 - スレーブドメインがバインドされている場合、ldm set-domain コマンドを呼び出す前に、指定したすべてのマスタートメインもバインドしておく必要があります。

```
ldm set-dom -i file
```

```
ldm set-dom [mac-addr=num] [hostid=num]
```

```
[failure-policy=ignore|panic|reset|stop] [master=[master-ldom1,...,master-ldom4]] ldom
```

ここでは、次のように指定します。

- **-i file** は、論理ドメインの作成に使用する XML 構成ファイルを指定します。
- **mac-addr=num** は、このドメインの MAC アドレスです。番号は、標準のオクテット記述法で指定する必要があります。たとえば、**80:00:33:55:22:66** とします。
- **hostid** は、特定のドメインのホスト ID を指定します。ホスト ID を指定しない場合、Logical Domains Manager により一意のホスト ID が各ドメインに割り当てられます。
- **failure-policy** は、マスタートメインの障害ポリシーを指定します。このポリシーは、マスタートメインに障害が発生した場合のスレーブドメインの動作を制御します。このプロパティは、マスタートメインで設定します。デフォルト値は **ignore** です。次に、有効なプロパティ値を示します。
 - **ignore** は、マスタートメインの障害を無視します。スレーブドメインは影響を受けません。
 - **panic** は、マスタートメインに障害が発生した場合、すべてのスレーブドメインにパニックを発生させます。
 - **reset** は、マスタートメインに障害が発生した場合、すべてのスレーブドメインをリセットします。
 - **stop** は、マスタートメインに障害が発生した場合、すべてのスレーブドメインを停止します。

- **master** は、1つのスレーブドメインに対して最大4つのマスタードメインの名前を指定します。このプロパティは、スレーブドメインで設定します。デフォルトでは、ドメインに対してマスタードメインは設定されていません。この操作の前に、マスタードメインが存在している必要があります。

注 - Logical Domains Manager では、依存サイクルが生じるドメイン関係を作成することはできません。

- **ldom** は、オプションを設定する論理ドメインの名前を指定します。

ドメインの削除

次のサブコマンドは、1つ以上の論理ドメインを削除します。

```
ldm rm-dom -a
ldm rm-dom ldom...
```

ここでは、次のように指定します。

- **-a** は、制御ドメインを除くすべての論理ドメインを削除します。
- **ldom** は、削除する論理ドメインを指定します。
破棄するドメインがマスタードメインとして指定されている場合は、このドメインへの参照がすべてのスレーブドメインから削除されます。

Logical Domainsの移行

このサブコマンドは、ドメインをある場所から別の場所に移行します。

```
ldm migrate-domain [-f] [-n] [-p filename] source-ldom [user@]target-host[:target-ldom]
```

ここでは、次のように指定します。

- **-f** は、ドメインの移行を強制的に行なおうとします。
- **-n** は、移行が成功するかどうかを判別するために、移行の予行演習を行います。ドメインの移行が実際に行われるわけではありません。
- **-p filename** を使用すると、ターゲットマシンに必要なパスワードを **filename** の先頭行から読み取ることができます。このオプションでは、自動マイグレーションを実行できます。自動マイグレーションでは、ターゲットマシンのパスワードをプロンプトで入力する必要がありません。

この方法でパスワードを格納する場合は、ファイルのアクセス権の設定が 400 または 600 であること、つまり root 所有者 (特権ユーザー) のみがファイルの読み取りまたは書き込みを許可されていることを確認します。

- **source-ldom** は、移行対象の論理ドメインです。
- **user** は、ターゲットホスト上で Logical Domains Manager の実行を承認されているユーザー名です。ユーザー名を指定しない場合、デフォルトで、このコマンドを実行しているユーザーの名前が使用されます。
- **target-host** は、**target-ldom** の配置先のホストです。

- *target-ldom* は、ターゲットマシンで使用する論理ドメイン名です。デフォルトでは、ソースドメイン (*source-ldom*) で使用されているドメイン名が保持されます。

再構成処理

Logical Domains は次の種類の再構成処理をサポートしています。

- 動的再構成 (**Dynamic Reconfiguration, DR**) 処理。DR は、アクティブなドメインに対してリソースの追加、設定、または削除を行う機能です。特定の種類のリソースの動的再構成を実行できるかどうかは、論理ドメインで動作している特定のバージョンの OS でサポートされているかどうかに依存します。制御ドメインでは、DR を実行できない場合には、代わりに遅延再構成処理が実行されます。また、`ldm start-reconf primary` コマンドを実行することにより、制御ドメイン上で遅延再構成モードに手動で切り替えることもできます。
- 遅延再構成 (**Delayed reconfiguration, DR**) 処理。ただちに有効になる DR 処理とは対照的に、遅延再構成処理は次の OS の再起動後に有効になるか、OS が動作していない場合は論理ドメインの停止および起動後に有効になります。遅延再構成処理は、制御ドメインでのみ実行できます。動的に構成できない資源を変更する前に、ほかのドメインを停止する必要があります。

動的再構成と遅延再構成については、『[Oracle VM Server for SPARC 2.0 管理ガイド](#)』の第 1 章「[Oracle VM Server for SPARC ソフトウェアの概要](#)」を参照してください。

CPU の操作

仮想 CPU の追加

次のサブコマンドは、指定された数の仮想 CPU またはコア全体の数を論理ドメインに追加します。ドメインにはコア全体と CPU スレッドを同時に構成することはできません。コア全体の構成と CPU スレッドの構成は相互排他的です。

```
ldm add-vcpu [-c] number ldom
```

ここでは、次のように指定します。

- *-c* は、コア全体が論理ドメインに割り当てられることを示します。
- *-c* オプションが指定されていない場合、*number* は論理ドメインに追加される仮想 CPU の数です。*-c* オプションが指定されている場合、*number* は論理ドメインに追加されるコア全体の数です。
- *ldom* は、仮想 CPU を追加する論理ドメインを指定します。

仮想 CPU の設定

次のサブコマンドは、論理ドメインで設定する仮想 CPU の数またはコア全体の数を指定します。ドメインにはコア全体と CPU スレッドを同時に構成することはできません。コア全体の構成と CPU スレッドの構成は相互排他的です。

```
ldm set-vcpu [-c] number ldom
```


ここでは、次のように指定します。

- `-c` は、コア全体が論理ドメインに割り当てられることを示します。
- `-c` オプションが指定されていない場合、*number* は論理ドメインに対して設定される仮想 CPU の数です。`-c` オプションが指定されている場合、*number* は論理ドメインに対して設定されるコア全体の数です。
- *ldom* は、仮想 CPU の数を設定する論理ドメインです。

仮想 CPU の削除

次のサブコマンドは、指定された数の仮想 CPU またはコア全体を論理ドメインから削除します。ドメインにはコア全体と CPU スレッドを同時に構成することはできません。コア全体の構成と CPU スレッドの構成は相互排他的です。

ldm rm-vcpu [-c] *number ldom*

ここでは、次のように指定します。

- `-c` は、論理ドメインからコア全体が削除されることを示します。
- `-c` オプションが指定されていない場合、*number* は論理ドメインから削除される仮想 CPU の数です。`-c` オプションが指定されている場合、*number* は論理ドメインから削除されるコア全体の数です。
- *ldom* は、仮想 CPU を削除する論理ドメインを指定します。

注- コア内の暗号化装置を論理ドメインに割り当てている場合は、論理ドメインからそのコア上のすべての仮想 CPU を削除することはできません。

暗号化装置の操作

暗号化装置の追加

次のサブコマンドは、論理ドメインに追加する暗号化装置の数を指定します。現在、サポートされているサーバー上でサポートされている暗号化装置は、モジュラー演算ユニット (Modular Arithmetic Unit、MAU) および Control Word Queue (CWQ) です。

ldm add-crypto *number ldom*

ここでは、次のように指定します。

- *number* は、論理ドメインに追加する暗号化装置の数です。
- *ldom* は、暗号化装置を追加する論理ドメインを指定します。

暗号化装置の設定

次のサブコマンドは、論理ドメインで設定する暗号化装置の数を指定します。

ldm set-crypto *number ldom*

ここでは、次のように指定します。

- *number* は、論理ドメインで設定する暗号化装置の数です。
- *ldom* は、暗号化装置の数を設定する論理ドメインを指定します。

暗号化装置の削除

次のサブコマンドは、指定した数の暗号化装置を論理ドメインから削除します。

```
ldm rm-crypto number ldom
```

ここでは、次のように指定します。

- *number* は、論理ドメインから削除する暗号化装置の数です。
- *ldom* は、暗号化装置を削除する論理ドメインを指定します。

メモリーの操作

メモリーの追加

次のサブコマンドは、指定された量のメモリーを論理ドメインに追加します。

```
ldm add-mem [--auto-adj] size[unit] ldom
```

ここでは、次のように指定します。

- **--auto-adj** は、アクティブドメインに追加されるメモリーの量を自動的に 256M バイト単位に揃えることを指定します。これにより、要求されるメモリーサイズが増える場合があります。ドメインがアクティブでないか、バインドされているか、遅延再構成の状態にある場合、このオプションでは、次の 256M バイト境界に切り上げることにより、ドメインの結果として得られるサイズを自動的に揃えます。
- *size* は、論理ドメインに追加するメモリーのサイズです。
- *unit* は測定単位です。デフォルトはバイト単位です。測定単位を変更する場合、次のいずれかを指定します。*unit* では、大文字と小文字が区別されません。
 - G は G バイト
 - K は K バイト
 - M は M バイト
- *ldom* は、メモリーを追加する論理ドメインを指定します。

メモリーの設定

次のサブコマンドは、論理ドメインで一定量のメモリーを設定します。指定したメモリーの量に応じて、このサブコマンドは **add-memory** または **remove-memory** 操作として扱われます。

```
ldm set-mem [--auto-adj] size[unit] ldom
```

ここでは、次のように指定します。

- `--auto-adj` は、アクティブドメインで設定されるメモリーの量を自動的に 256M バイト単位に揃えることを指定します。これにより、要求されるメモリーサイズが増す場合があります。ドメインがアクティブでないか、バインドされているか、遅延再構成の状態にある場合、このオプションでは、次の 256M バイト境界に切り上げることににより、ドメインの結果として得られるサイズを自動的に揃えます。
- `size` は、論理ドメインで設定するメモリーのサイズです。
- `unit` は測定単位です。デフォルトはバイト単位です。測定単位を変更する場合、次のいずれかを指定します。`unit` では、大文字と小文字が区別されません。
 - `g` は G バイト
 - `k` は K バイト
 - `m` は M バイト
- `ldom` は、メモリーを変更する論理ドメインを指定します。

メモリーの削除

次のサブコマンドは、指定された量のメモリーを論理ドメインから削除します。

```
ldm rm-mem [--auto-adj] size[unit] ldom
```

ここでは、次のように指定します。

- `--auto-adj` は、アクティブドメインから削除されるメモリーの量を自動的に 256M バイト単位に揃えることを指定します。これにより、要求されるメモリーサイズが増える場合があります。ドメインがアクティブでないか、バインドされているか、遅延再構成の状態にある場合、このオプションでは、次の 256M バイト境界に切り上げることににより、ドメインの結果として得られるサイズを自動的に揃えます。
- `size` は、論理ドメインから削除するメモリーのサイズです。
- `unit` は測定単位です。デフォルトはバイト単位です。測定単位を変更する場合、次のいずれかを指定します。`unit` では、大文字と小文字が区別されません。
 - `g` は G バイト
 - `k` は K バイト
 - `m` は M バイト
- `ldom` は、メモリーを削除する論理ドメインを指定します。

遅延再構成モード
の入力

次のサブコマンドは、ドメインの遅延再構成モードへの切り替えを有効にします。`primary` は、`ldom` の唯一の有効な値です。

```
ldm start-reconf ldom
```

遅延再構成処理の
取り消し

次のサブコマンドは、遅延再構成を取り消します。primary は、ldom の唯一の有効な値です。

```
ldm cancel-reconf ldom
```

操作の取り消し

次のサブコマンドは、論理ドメインに対する遅延再構成 (reconf)、メモリー DR 削除 (memdr)、またはドメイン移行 (migration) を取り消します。reconf 操作の場合、primary は ldom の唯一の有効な値です。

```
ldm cancel-op migration ldom
```

```
ldm cancel-op reconf ldom
```

```
ldm cancel-op memdr ldom
```

I/O デバイス

I/O デバイスの追加

次のサブコマンドは、指定された論理ドメインに PCI バスまたはデバイスを追加します。

```
ldm add-io bus-or-device ldom
```

ここでは、次のように指定します。

- *bus-or-device* は、直接 I/O を割り当て可能なデバイスまたは PCIe バスです。このデバイスは、デバイスパス形式で、または仮名として指定できます。直接 I/O を割り当て可能なデバイスの例では、PCIe1 仮名は pci@400/pci@0/pci@c デバイスパスと一致します。仮名は、対応する I/O カードスロットを識別するためにシャーシに記載されている ASCII ラベルに基づき、プラットフォーム固有です。デバイスパス名の代わりに、デバイス仮名を使用してデバイスを指定します。PCIe バスの例では、pci_0 仮名は pci@400 デバイスパスと一致します。
- *ldom* は、バスまたはデバイスを追加する論理ドメインを指定します。

I/O デバイスの削除

次のサブコマンドは、指定された論理ドメインから PCI バスまたはデバイスを削除します。

```
ldm rm-io bus-or-device ldom
```

ここでは、次のように指定します。

- *bus-or-device* は、直接 I/O を割り当て可能なデバイスまたは PCIe バスです。このデバイスは、デバイスパス形式で、または仮名として指定できます。直接 I/O を割り当て可能なデバイスの例では、PCIe1 仮名は pci@400/pci@0/pci@c デバイスパスと一致します。仮名は、対応する I/O カードスロットを識別するためにシャーシに記載されている ASCII ラベルに基づき、プラットフォーム固有です。デバイスパス名の代わりに、デバイス仮名を使用してデバイスを指定します。PCIe バスの例では、pci_0 仮名は pci@400 デバイスパスと一致します。
- *ldom* は、バスまたはデバイスを削除する論理ドメインを指定します。

仮想ネットワーク サーバー

仮想スイッチの追加

次のサブコマンドは、指定した論理ドメインに仮想スイッチを追加します。

```
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
```

ここでは、次のように指定します。

- **default-vlan-id=vlan-id** は、仮想スイッチとそれに関連する仮想ネットワークデバイスが暗黙にタグなしモードで属するデフォルトの VLAN を指定します。これは、仮想スイッチおよび仮想ネットワークデバイスのデフォルトのポート VLAN ID (**pvid**) として機能します。このオプションを指定しない場合、このプロパティのデフォルト値は 1 です。通常、このオプションを指定する必要はありません。このオプションは、単にデフォルト値の 1 を変更する手段として用意されています。
- **pvid=port-vlan-id** には、仮想スイッチデバイスをメンバーにする必要のある VLAN をタグなしモードで指定します。このプロパティは、**set-vsw** サブコマンドにも適用されます。『[Oracle VM Server for SPARC 2.0 管理ガイド](#)』の「[VLAN のタグ付けの使用](#)」を参照してください。
- **linkprop=phys-state** では、配下の物理ネットワークデバイスに基づいて、仮想デバイスがリンクステータスをレポートするかどうかを指定できます。コマンドラインで **linkprop=phys-state** を指定すると、仮想デバイスのリンクステータスは物理リンクステータスを反映します。デフォルトでは、仮想デバイスのリンクステータスは物理リンクステータスを反映しません。
- **vid=vlan-id** は、仮想ネットワークデバイスまたは仮想スイッチをメンバーにする必要がある 1 つ以上の VLAN を、タグ付きモードで指定します。このプロパティは、**set-vsw** サブコマンドにも適用されます。詳細は、『[Oracle VM Server for SPARC 2.0 管理ガイド](#)』の「[VLAN のタグ付けの使用](#)」を参照してください。
- **mac-addr=num** は、このスイッチが使用する MAC アドレスです。番号は、標準のオクテット記述法で指定する必要があります。たとえば、80:00:33:55:22:66 とします。MAC アドレスを指定しない場合、スイッチには、Logical Domains Manager に割り当てられる公開 MAC アドレスの範囲からアドレスが自動的に割り当てます。
- **net-dev=device** は、このスイッチが動作するネットワークデバイスへのパスです。
- **mode=sc** は、Logical Domains 環境で Oracle Solaris クラスタ ハートビートパケットの処理を優先順位付けするために、仮想ネットワークのサポートを有効にします。Oracle Solaris クラスタ などのアプリケーションでは、輻輳した仮想ネットワークおよびスイッチデバイスによって高優先度のハートビートパケットがドロップされないようにする必要があります。このオプションを使用して、Oracle

Solaris クラスタ のハートビートフレームが優先され、これらのフレームが信頼性の高い方法で転送されるようにします。

Oracle Solaris クラスタ を Logical Domains 環境で実行し、ゲストドメインを Oracle Solaris クラスタ ノードとして使用している場合にこのオプションを設定する必要があります。ゲストドメインで Oracle Solaris クラスタ ソフトウェアを実行していない場合には、仮想ネットワークのパフォーマンスに影響を与える可能性があるため、このオプションを設定しないでください。

- `mtu=size` は、仮想スイッチデバイスの最大転送単位 (MTU) を指定します。有効な値の範囲は 1500 潤 16000 です。
- `id=switch-id` は、新しい仮想スイッチデバイスの ID です。デフォルトでは ID 値は自動的に生成されるため、OS で既存のデバイス名に一致させる必要がある場合に、このプロパティを設定します。
- `vswitch-name` は、サービスとしてエクスポートされるスイッチの一意の名前です。クライアント (ネットワーク) は、このサービスに接続できます。
- `ldom` は、仮想スイッチを追加する論理ドメインを指定します。

仮想スイッチのオプションの設定

次のサブコマンドは、すでに追加されている仮想スイッチのプロパティを変更します。

```
ldm set-vsw [pvid=port-vlan-id] [vid=vlan-id1,vlan-id2,...] [mac-addr=num] [net-dev=device]
[linkprop=phys-state] [mode=sc] [mtu=size] vswitch-name
```

ここでは、次のように指定します。

- `pvid=port-vlan-id` には、仮想スイッチデバイスをメンバーにする必要のある VLAN をタグなしモードで指定します。『[Oracle VM Server for SPARC 2.0 管理ガイド](#)』の「[VLAN のタグ付けの使用](#)」を参照してください。
- `vid=vlan-id` は、仮想ネットワークデバイスまたは仮想スイッチをメンバーにする必要がある 1 つ以上の VLAN を、タグ付きモードで指定します。『[Oracle VM Server for SPARC 2.0 管理ガイド](#)』の「[VLAN のタグ付けの使用](#)」を参照してください。
- `mac-addr=num` は、スイッチで使用される MAC アドレスです。番号は、標準のオクテット記述法で指定する必要があります。たとえば、80:00:33:55:22:66 とします。
- `net-dev=device` は、このスイッチが処理するネットワークデバイスへのパスです。
- `linkprop=phys-state` では、配下の物理ネットワークデバイスに基づいて、仮想デバイスがリンクステータスをレポートするかどうかを指定できます。コマンドラインで `linkprop=phys-state` を指定すると、仮想デバイスのリンクステータスは物理リンクステータスを反映します。デフォルトでは、仮想デバイスのリンクステータスは物理リンクステータスを反映しません。linkprop プロパ

ティールを指定していない場合や、`linkprop=` 引数を使用して `ldm set-vsw` コマンドを実行する場合は、デフォルトの状態になります。

- `mode=sc` は、Logical Domains 環境で Oracle Solaris クラスタ ハートビートパケットの処理を優先順位付けするために、仮想ネットワークのサポートを有効にします。Oracle Solaris クラスタ などのアプリケーションでは、輻輳した仮想ネットワークおよびスイッチデバイスによって高優先度のハートビートパケットがドロップされないようにする必要があります。このオプションを使用して、Oracle Solaris クラスタ のハートビートフレームが優先され、これらのフレームが信頼性の高い方法で転送されるようにします。

`mode=` (空白のまま) では、ハートビートパケットの特殊処理が停止されます。

Oracle Solaris クラスタ を Logical Domains 環境で実行し、ゲストドメインを Oracle Solaris クラスタ ノードとして使用している場合にこのオプションを設定する必要があります。ゲストドメインで Oracle Solaris クラスタ ソフトウェアを実行していない場合には、仮想ネットワークのパフォーマンスに影響を与える可能性があるため、このオプションを設定しないでください。

- `mtu=size` は、仮想スイッチデバイスの最大転送単位 (MTU) を指定します。有効な値の範囲は 1500 潤才 16000 です。
- `vswitch-name` は、サービスとしてエクスポートされるスイッチの一意の名前です。クライアント (ネットワーク) は、このサービスに接続できます。

仮想スイッチの削除

次のサブコマンドは、仮想スイッチを削除します。

```
ldm rm-vsw [-f] vswitch-name
```

ここでは、次のように指定します。

- `-f` は、仮想スイッチの強制削除を試行します。削除は失敗することがあります。
- `vswitch-name` は、サービスとして削除されるスイッチの名前です。

仮想ネットワーク-クライアント

仮想ネットワークデバイスの追加

次のサブコマンドは、指定した論理ドメインに仮想ネットワークデバイスを追加します。

```
ldm add-vnet [mac-addr=num] [mode=hybrid] [pvid=port-vlan-id] [vid=vlan-id1,vlan-id2,...]  
[linkprop=phys-state] [id=network-id] [mtu=size] if-name vswitch-name ldom
```


ここでは、次のように指定します。

- `mac-addr=num` は、このネットワークデバイスの MAC アドレスです。番号は、標準のオクテット記述法で指定する必要があります。たとえば、`80:00:33:55:22:66` とします。
- `mode=hybrid` は、可能な場合に、この `vnet` で NIU ハイブリッド I/O を使用するようシステムに要求します。可能でない場合は、システムは仮想 I/O に戻ります。このハイブリッドモードを制御ドメイン上のアクティブな `vnet` で設定した場合、このモードは遅延再構成とみなされます。『[Oracle VM Server for SPARC 2.0 管理ガイド](#)』の「[NIU ハイブリッド I/O の使用](#)」を参照してください。
- `pvid=port-vlan-id` には、仮想ネットワークデバイスをメンバーにする必要のある VLAN をタグなしモードで指定します。『[Oracle VM Server for SPARC 2.0 管理ガイド](#)』の「[VLAN のタグ付けの使用](#)」を参照してください。
- `vid=vlan-id` は、仮想ネットワークデバイスがタグ付きモードでメンバーとして属する必要のある 1 つ以上の VLAN を指定します。『[Oracle VM Server for SPARC 2.0 管理ガイド](#)』の「[VLAN のタグ付けの使用](#)」を参照してください。
- `mtu=size` は、仮想ネットワークデバイスの最大転送単位 (MTU) を指定します。有効な値の範囲は 1500 潤 16000 です。
- `linkprop=phys-state` では、配下の物理ネットワークデバイスに基づいて、仮想ネットワークデバイスがリンクステータスをレポートするかどうかを指定できます。コマンドラインで `linkprop=phys-state` を指定すると、仮想ネットワークデバイスのリンクステータスは物理リンクステータスを反映します。デフォルトでは、仮想ネットワークデバイスのリンクステータスは物理リンクステータスを反映しません。
- `id=network-id` は、新しい仮想ネットワークデバイスの ID です。デフォルトでは ID 値は自動的に生成されるため、OS で既存のデバイス名に一致させる必要がある場合に、このプロパティを設定します。
- `if-name` は、後続の `set-vnet` または `rm-vnet` サブコマンドで参照するためにこの仮想ネットワークデバイスのインスタンスに割り当てられる、論理ドメインで一意のインタフェース名です。
- `vswitch-name` は、接続する既存のネットワークサービス (仮想スイッチ) の名前です。
- `ldom` は、仮想ネットワークデバイスを追加する論理ドメインを指定します。

仮想ネットワークデバイスのオプションの設定

次のサブコマンドは、指定した論理ドメインで仮想ネットワークデバイスのオプションを設定します。

```
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
```


ここでは、次のように指定します。

- `mac-addr=num` は、このネットワークデバイスの MAC アドレスです。番号は、標準のオクテット記述法で指定する必要があります。たとえば、80:00:33:55:22:66 とします。
- `vswitch=vswitch-name` は、接続する既存のネットワークサービス (仮想スイッチ) の名前です。
- `mode=hybrid` を指定すると、この `vnet` での NIU ハイブリッド I/O 操作が有効になります。このオプションを制御ドメイン上のアクティブな `vnet` で設定した場合、遅延再構成とみなされます。NIU ハイブリッド I/O を無効にするには、`mode=` 引数を空白のままにします。
- `pvid=port-vlan-id` は、仮想ネットワークデバイスがタグなしモードでメンバーとして属する必要がある VLAN を指定します。『[Oracle VM Server for SPARC 2.0 管理ガイド](#)』の「[VLAN のタグ付けの使用](#)」を参照してください。
- `linkprop=phys-state` では、配下の物理ネットワークデバイスに基づいて、仮想デバイスがリンクステータスをレポートするかどうかを指定できます。コマンドラインで `linkprop=phys-state` を指定すると、仮想デバイスのリンクステータスは物理リンクステータスを反映します。デフォルトでは、仮想デバイスのリンクステータスは物理リンクステータスを反映しません。`linkprop` プロパティを指定していない場合や、`linkprop=` 引数を使用して `ldm set-vnet` コマンドを実行する場合は、デフォルトの状態になります。
- `vid=vlan-id` は、仮想ネットワークデバイスがタグ付きモードでメンバーとして属する必要がある 1 つ以上の VLAN を指定します。『[Oracle VM Server for SPARC 2.0 管理ガイド](#)』の「[VLAN のタグ付けの使用](#)」を参照してください。
- `mtu=size` は、仮想ネットワークデバイスの最大転送単位 (MTU) を指定します。有効な値の範囲は 1500 潤 16000 です。
- `if-name` は、設定する仮想ネットワークデバイスに割り当てられた一意のインタフェース名です。
- `ldom` は、仮想ネットワークデバイスを変更する論理ドメインを指定します。

仮想ネットワークデバイスの削除

次のサブコマンドは、指定した論理ドメインから仮想ネットワークデバイスを削除します。

```
ldm rm-vnet [-f] if-name ldom
```

ここでは、次のように指定します。

- `-f` は、論理ドメインからの仮想ネットワークデバイスの強制削除を試行します。削除は失敗することがあります。
- `if-name` は、削除する仮想ネットワークデバイスに割り当てられた一意のインタフェース名です。
- `ldom` は、仮想ネットワークデバイスを削除する論理ドメインを指定します。

仮想ディスク-
サービス

仮想ディスクサーバーの追加

次のサブコマンドは、指定した論理ドメインに仮想ディスクサーバーを追加します。

```
ldm add-vds service-name ldom
```

ここでは、次のように指定します。

- *service-name* は、仮想ディスクサーバーのこのインスタンスのサービス名です。この *service-name* は、サーバー上のすべての仮想ディスクサーバーインスタンスの中で一意である必要があります。
- *ldom* は、仮想ディスクサーバーを追加する論理ドメインを指定します。

仮想ディスクサーバーの削除

次のサブコマンドは、仮想ディスクサーバーを削除します。

```
ldm rm-vds [-f] service-name
```

ここでは、次のように指定します。

- **-f** は、仮想ディスクサーバーの削除を強制的に試行します。削除は失敗することがあります。
- *service-name* は、仮想ディスクサーバーのこのインスタンスの一意のサービス名です。

注意--f オプションを指定すると、削除前にすべてのクライアントのバインド解除が試行されます。書き込みが進行中の場合は、ディスクデータが失われる可能性があります。

仮想ディスクサーバーへのデバイスの追加

次のサブコマンドは、仮想ディスクサーバーにデバイスを追加します。このデバイスには、ディスク全体、ディスクのスライス、ファイル、またはディスクボリュームを指定できます。『[Oracle VM Server for SPARC 2.0 管理ガイド](#)』の第7章「[仮想ディスクの使用](#)」を参照してください。

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

ここでは、次のように指定します。

- `-f` は、すでに別の仮想ディスクサーバーに属しているブロックデバイスパスを指定する場合に、追加の仮想ディスクサーバーの作成を強制的に試行します。`-f` オプションを指定する場合は、引数リストの先頭に指定する必要があります。
- `options=` は、次の値を指定します。
 - `ro` - 読み取り専用アクセスを指定
 - `slice` - 単一スライスのディスクとしてバックエンドをエクスポート
 - `excl` - 排他的なディスクアクセスを指定

`options=` 引数を指定しないと、デフォルト値であるディスク、非排他的、および読み取り/書き込みになります。`options=` 引数を追加する場合は、特定の仮想ディスクサーバーデバイスに1つ以上のオプションを指定する必要があります。2つ以上のオプションは、空白文字を入れずにコンマで区切って、`ro,slice,excl` のように指定します。

- `mpgroup=mpgroup` は、仮想ディスクのフェイルオーバーをサポートするために使用されるディスクのマルチパスグループ名です。現在使用中の仮想ディスクサーバーデバイスへの接続に障害が発生した場合に備えて、仮想ディスクの複数の冗長パスを割り当てることができます。それには、複数の仮想ディスクサーバーデバイス (`vdsdev`) を1つのマルチパスグループ (`mpgroup`) にまとめます。すべての仮想ディスクサーバーデバイスが同じ `mpgroup` 名を持つこととなります。仮想ディスクがマルチパスグループ内の仮想ディスクサーバーデバイスのいずれかにバインドされると、その仮想ディスクは、`mpgroup` に属するすべての仮想ディスクサーバーデバイスにバインドされます。
- `backend` は、仮想ディスクのデータが格納される場所です。バックエンドには、ディスク、ディスクのスライス、ファイル、ボリューム (ZFS、SVM、VxVM など)、または任意の擬似ディスクデバイスを指定できます。ディスクラベルには、SMI VTOC、EFI、またはラベルなしを指定できます。バックエンドは、バックエンドをサービスドメインからエクスポートする際に `slice` オプションを設定するかどうかに応じて、フルディスクまたは1つのスライスディスクのいずれかとしてゲストドメインに表示されます。デバイスを追加する場合、`volume-name` を `backend` と組み合わせる必要があります。
- `volume-name` は、仮想ディスクサーバーに追加するデバイスに指定する必要がある一意の名前です。`volume-name` は、仮想ディスクサーバーのこのインスタンスで一意である必要があります。この名前は、追加のために仮想ディスクサーバーによってクライアントにエクスポートされるためです。デバイスを追加する場合、`volume-name` を `backend` と組み合わせる必要があります。
- `service-name` は、このデバイスを追加する仮想ディスクサーバーの名前です。

仮想ディスクサーバーデバイスのオプションの設定

次のサブコマンドは、仮想ディスクサーバーのオプションを設定します。『[Oracle VM Server for SPARC 2.0 管理ガイド](#)』を参照してください。

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

ここでは、次のように指定します。

- **-f** は、同じ論理ドメイン内の複数のボリュームが読み取り専用モード (option=ro) で同一のブロックデバイスパスを共有している場合に、読み取り専用制限を解除します。**-f** オプションを指定する場合は、引数リストの先頭に指定する必要があります。
- **options=** は、次の値を指定します。
 - **ro** - 読み取り専用アクセスを指定
 - **slice** - 単一スライスのディスクとしてバックエンドをエクスポート
 - **excl** - 排他的なディスクアクセスを指定
 - 以前に指定したオプションをオフにするには、**options=** 引数を空白のままにします。特定の仮想ディスクサーバーデバイスに、次のオプションのすべてまたはサブセットを指定できます。2つ以上のオプションは、空白文字を入れずにコンマで区切って、**ro,slice,excl** のように指定します。
- **mpgroup=mpgroup** は、仮想ディスクのフェイルオーバーをサポートするために使用されるディスクのマルチパスグループ名です。現在使用中の仮想ディスクサーバーデバイスへの接続に障害が発生した場合に備えて、仮想ディスクの複数の冗長パスを割り当てることができます。それには、複数の仮想ディスクサーバーデバイス (**vdsdev**) を1つのマルチパスグループ (**mpgroup**) にまとめます。すべての仮想ディスクサーバーデバイスが同じ **mpgroup** 名を持つことになります。仮想ディスクがマルチパスグループ内の仮想ディスクサーバーデバイスのいずれかにバインドされると、その仮想ディスクは、**mpgroup** に属するすべての仮想ディスクサーバーデバイスにバインドされます。
- **volume-name** は、**service-name** で指定したサービスによってエクスポートされた既存のボリュームの名前です。
- **service-name** は、変更する仮想ディスクサーバーの名前です。

仮想ディスクサーバーからのデバイスの削除

次のサブコマンドは、仮想ディスクサーバーからデバイスを削除します。

```
ldm rm-vdsdev [-f] volume-name@service-name
```

ここでは、次のように指定します。

- **-f** は、仮想ディスクサーバーのデバイスの削除を強制的に試行します。削除は失敗することがあります。
- **volume-name** は、仮想ディスクサーバーから削除するデバイスの一意の名前です。
- **service-name** は、このデバイスを削除する仮想ディスクサーバーの名前です。

注意 `--f` オプションを指定しない場合、デバイスがビジーだと `rm-vdsdev` サブコマンドは仮想ディスクサーバーのデバイスの削除を許可しません。 `-f` オプションを指定した場合は、開いているファイルのデータが失われる可能性があります。

仮想ディスク-クライアント

仮想ディスクの追加

次のサブコマンドは、指定した論理ドメインに仮想ディスクを追加します。仮想ディスクサーバーとの接続を確立することができない場合、オプションのタイムアウトプロパティを使用すると、仮想ディスクのタイムアウトを指定することができます。

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

ここでは、次のように指定します。

- `timeout=seconds` は、仮想ディスククライアント (vdc) と仮想ディスクサーバー (vds) の間の接続を確立する際の秒数です。複数の仮想ディスク (vdisk) パスがある場合、vdc は、別の vds への接続を試みることができます。また、タイムアウトによって、いずれかの vds への接続が指定の時間内に確実に行われます。
`timeout=` 引数を指定しないか、または `timeout=0` を設定すると、仮想ディスクは無制限に待機します。
- `id=disk-id` は、新しい仮想ディスクデバイスの ID です。デフォルトでは ID 値は自動的に生成されるため、OS で既存のデバイス名に一致させる必要がある場合に、このプロパティを設定します。
- `disk-name` は、仮想ディスクの名前です。
- `volume-name` は、接続する既存の仮想ディスクサーバーデバイスの名前です。
- `service-name` は、接続する既存の仮想ディスクサーバーの名前です。
- `ldom` は、仮想ディスクを追加する論理ドメインを指定します。

仮想ディスクのオプションの設定

次のサブコマンドは、指定した論理ドメインで仮想ディスクのオプションを設定します。仮想ディスクサーバーとの接続を確立することができない場合、オプションのタイムアウトプロパティを使用すると、仮想ディスクのタイムアウトを指定することができます。

```
ldm set-vdisk [timeout=seconds] [volume=volume-name@service-name] disk-name ldom
```

ここでは、次のように指定します。

- `timeout=seconds` は、仮想ディスククライアント (vdc) と仮想ディスクサーバー (vds) の間の接続を確立する際の秒数です。複数の仮想ディスク (vdisk) パスがある場合、vdc は、別の vds への接続を試みることができます。また、タイムアウトによって、いずれかの vds への接続が指定の時間内に確実に行われます。
タイムアウトを無効にするには、`timeout=0` と設定します。

`timeout=` 引数で、仮想ディスクが無制限に待機するように指定しないでください。

- `volume=volume-name` は、接続する仮想ディスクサーバーデバイスの名前です。`service-name` は、接続する仮想ディスクサーバーの名前です。
- `disk-name` は、既存の仮想ディスクの名前です。
- `ldom` は、以前にこの仮想ディスクを追加した既存の論理ドメインを指定します。

仮想ディスクの削除

次のサブコマンドは、指定した論理ドメインから仮想ディスクを削除します。

```
ldm rm-vdisk [-f] disk-name ldom
```

ここでは、次のように指定します。

- `-f` は、仮想ディスクの削除を強制的に試行します。削除は失敗することがあります。
- `disk-name` は、削除する仮想ディスクの名前です。
- `ldom` は、仮想ディスクを削除する論理ドメインを指定します。

仮想データプレーンのチャンネルサービス

仮想データプレーンのチャンネルサービスの追加

次のサブコマンドは、指定した論理ドメインに仮想データプレーンのチャンネルサービスを追加します。このサブコマンドは、Netra Data Plane Software (NDPS) 環境でのみ使用してください。

```
ldm add-vdpcs vdpcs-service-name ldom
```

ここでは、次のように指定します。

- `vdpcs-service-name` は、追加する仮想データプレーンのチャンネルサービスの名前です。
- `ldom` は、仮想データプレーンのチャンネルサービスを追加する論理ドメインを指定します。

仮想データプレーンのチャンネルサービスの削除

次のサブコマンドは、仮想データプレーンのチャンネルサービスを削除します。このサブコマンドは、Netra Data Plane Software (NDPS) 環境でのみ使用してください。

```
ldm rm-vdpcs [-f] vdpcs-service-name
```

ここでは、次のように指定します。

- `-f` は、仮想データプレーンのチャンネルサービスの削除を強制的に試行します。削除は失敗することがあります。
- `vdpcs-service-name` は、削除する仮想データプレーンのチャンネルサービスの名前です。

仮想データ
プレーンのチャンネル-クライアント

仮想データプレーンのチャンネルクライアントの追加

次のサブコマンドは、指定した論理ドメインに仮想データプレーンのチャンネルクライアントを追加します。このサブコマンドは、Netra Data Plane Software (NDPS) 環境でのみ使用してください。

```
ldm add-vdpcc vdpc-name vdpcs-service-name ldom
```

ここでは、次のように指定します。

- `vdpc-name` は、仮想データプレーンのチャンネルサービスクライアントの一意の名前です。
- `vdpcs-service-name` は、このクライアントに接続する仮想データプレーンのチャンネルサービスの名前です。
- `ldom` は、仮想データプレーンのチャンネルクライアントを追加する論理ドメインを指定します。

仮想データプレーンのチャンネルクライアントの削除

次のサブコマンドは、指定した論理ドメインから仮想データプレーンのチャンネルクライアントを削除します。このサブコマンドは、Netra Data Plane Software (NDPS) 環境でのみ使用してください。

```
ldm rm-vdpcc [-f] vdpc-name ldom
```

ここでは、次のように指定します。

- `-f` は、仮想データプレーンのチャンネルクライアントの削除を強制的に試行します。削除は失敗することがあります。
- `vdpc-name` は、削除する仮想データプレーンのチャンネルクライアントに割り当てられた一意の名前です。
- `ldom` は、仮想データプレーンのチャンネルクライアントを削除する論理ドメインを指定します。

仮想コンソール

仮想コンソール端末集配信装置の追加

次のサブコマンドは、指定した論理ドメインに仮想コンソール端末集配信装置を追加します。


```
ldm add-vcc port-range=x-y vcc-name ldom
```

ここでは、次のように指定します。

- **port-range=x-y** は、仮想コンソール端末集配信装置でコンソール接続に使用される TCP ポートの範囲です。
- **vcc-name** は、追加する仮想コンソール端末集配信装置の名前です。
- **ldom** は、仮想コンソール端末集配信装置を追加する論理ドメインを指定します。

仮想コンソール端末集配信装置のオプションの設定

次のサブコマンドは、特定の仮想コンソール端末集配信装置のオプションを設定します。

```
ldm set-vcc port-range=x-y vcc-name
```

ここでは、次のように指定します。

- **port-range=x-y** は、仮想コンソール端末集配信装置でコンソール接続に使用される TCP ポートの範囲です。変更後のポートの範囲には、端末集配信装置のクライアントに割り当てられているすべてのポートが含まれている必要があります。
- **vcc-name** は、設定する仮想コンソール端末集配信装置の名前です。

仮想コンソール端末集配信装置の削除

次のサブコマンドは、指定した論理ドメインから仮想コンソール端末集配信装置を削除します。

```
ldm rm-vcc [-f] vcc-name
```

ここでは、次のように指定します。

- **-f** は、仮想コンソール端末集配信装置の削除を強制的に試行します。削除は失敗することがあります。
- **vcc-name** は、削除する仮想コンソール端末集配信装置の名前です。

注意 **-f** オプションを指定すると、削除前にすべてのクライアントのバインド解除が試行されます。書き込みが進行中の場合は、データが失われる可能性があります。

仮想コンソールのオプションの設定

次のサブコマンドは、指定した論理ドメインで特定のポート番号とグループを設定します。また、接続されているコンソールのサービスを設定することもできます。このサブコマンドは、ドメインがアクティブでない場合にのみ使用できます。

```
ldm set-vcons [port=[port-num]] [group=group] [service=vcc-server] ldom
```


ここでは、次のように指定します。

- `port=port-num` は、このコンソールで使用する特定のポートです。Logical Domains Manager で自動的にポート番号を割り当てるには、`port-num` を空白のままにします。
- `group=group` は、このコンソールに接続する新しいグループです。グループ引数を使用すると、同一の TCP 接続上で複数のコンソールを多重化できます。この概念については、Oracle Solaris OS [vntsd\(1M\)](#) マニュアルページを参照してください。グループを指定した場合は、サービスも指定する必要があります。
- `service=vcc-server` は、コンソール接続を処理する既存の仮想コンソール端末集配信装置の名前です。グループを指定した場合は、サービスを指定する必要があります。
- `ldom` は、仮想コンソール端末集配信装置を設定する論理ドメインを指定します。

変数

変数の追加

次のサブコマンドは、論理ドメインに1つ以上の変数を追加します。

```
ldm add-var var-name=[value]... ldom
```

ここでは、次のように指定します。

- `var-name=value` は、追加する変数の名前と値の組み合わせです。値は省略可能です。
- `ldom` は、変数を追加する論理ドメインを指定します。

変数の設定

次のサブコマンドは、論理ドメインの変数を設定します。

```
ldm set-var var-name=[value]... ldom
```

ここでは、次のように指定します。

- `var-name=value` は、設定する変数の名前と値の組み合わせです。値は省略可能です。
- `ldom` は、変数を設定する論理ドメインを指定します。

注 - `value` を空白のままにすると、`var-name` は値なしに設定されます。

変数の削除

次のサブコマンドは、論理ドメインの変数を削除します。

```
ldm rm-var var-name... ldom
```

ここでは、次のように指定します。

- *var-name* は、削除する変数の名前です。
- *ldom* は、変数を削除する論理ドメインを指定します。

その他の操作

ドメインの起動

次のサブコマンドは、1つ以上の論理ドメインを起動します。

```
ldm start -a
ldm start -i file
ldm start ldom...
```

ここでは、次のように指定します。

- *-a* は、バインドされているすべての論理ドメインを起動します。
- *-i file* は、論理ドメインの起動に使用する XML 構成ファイルを指定します。
- *ldom* は、起動する 1つ以上の論理ドメインを指定します。

ドメインの停止

次のサブコマンドは、1つ以上の動作中の論理ドメインを停止します。Oracle Solaris OS が起動している場合、このサブコマンドは論理ドメインに shutdown(1M) 要求を送信します。

```
ldm stop [-f] -a
ldm stop [-f] ldom...
```

ここでは、次のように指定します。

- *-f* は、動作中の論理ドメインの停止を強制的に試行します。ドメインをその他の手段で停止できない場合にのみ使用してください。
- *-a* は、制御ドメインを除く動作中のすべての論理ドメインを停止します。
- *ldom* は、停止する 1つ以上の動作中の論理ドメインを指定します。

Oracle Solaris OS でのパニックの発生

次のサブコマンドは、指定した論理ドメイン上の Oracle Solaris OS でパニックを発生させます。Oracle Solaris OS でパニックが発生するように設定されている場合、このサブコマンドはバックトレースおよびクラッシュダンプを提供します。dumpadm(1M) コマンドは、クラッシュダンプを構成する手段を提供します。

```
ldm panic ldom
```

ldom は、パニックを発生させる論理ドメインを指定します。

ヘルプ情報の表示

次のサブコマンドは、すべてのサブコマンドまたは指定したサブコマンドの使用方法を表示します。また、ldm コマンドを単独で使用するだけでも、すべてのサブコマンドの使用方法を表示できます。

ldm --help [*subcommand*]

subcommand は、使用法についての情報を表示する ldm サブコマンドを指定します。

バージョン情報の表示

次のサブコマンドは、バージョン情報を表示します。

ldm --version

ldm -V

ドメインへの資源のバインド

次のサブコマンドは、論理ドメインに構成済みの資源をバインド (接続) します。

ldm bind-dom -i *file*

ldm bind-dom *ldom*

ここでは、次のように指定します。

- **-i** *file* は、論理ドメインのバインドに使用する XML 構成ファイルを指定します。
- *ldom* は、資源をバインドする論理ドメインを指定します。

ドメインからの資源のバインド解除

次のサブコマンドは、構成された論理ドメインにバインドされている資源を解放します。

ldm unbind-dom *ldom*

ldom は、資源のバインドを解除する論理ドメインを指定します。

構成の操作

論理ドメイン構成の追加

次のサブコマンドは、現在アクティブな構成または以前に自動保存された構成に基づいて、論理ドメイン構成を追加します。この構成は、サービスプロセッサ (SP) に格納されます。

ldm add-config *config-name*

ldm add-config -r *autosave-name* [*new-config-name*]

ここでは、次のように指定します。

- *config-name* は、追加する論理ドメイン構成の名前です。
- **-r** *autosave-name* は、自動保存構成データを次のいずれかに適用します。
 - SP 上の同じ名前の構成
 - SP 上に存在しない、新たに作成される構成 (*new-config-name*)

対象の構成が SP に存在していない場合は、その名前の構成が、対応する自動保存構成の内容に基づいて作成され、SP に保存されます。自動保存構成データが適用されたあと、これらの自動保存ファイルは、制御ドメインから削除されます。*autosave-name* が現在選択している構成を示していない場合、または *new-config-name* を指定した場合は、SP 上の現在の構成の状態や、制御ドメイン上のその構成の自動保存ファイルには影響がありません。

破損していることがわかっている自動保存構成を回復するには、*-r new-config-name* を指定する必要があります。破損していることがわかっている構成で既存の構成を上書きすることはできません。

- *new-config-name* は、追加する論理ドメイン構成の名前です。

論理ドメイン構成の設定

次のサブコマンドを使用すると、使用する論理ドメイン構成を指定できます。この構成は、SP に格納されます。

```
ldm set-config config-name
```

config-name は、使用する論理ドメイン構成の名前です。

デフォルトの構成名は、*factory-default* です。デフォルトの構成を指定するには、次のサブコマンドを使用します。

```
ldm set-config factory-default
```

論理ドメイン構成の削除

次のサブコマンドは、SP に格納されている論理ドメイン構成を削除し、対応する自動保存構成を制御ドメインから削除します。

```
ldm rm-config [-r] config-name
```

ここでは、次のように指定します。

- *-r* は、自動保存構成だけを制御ドメインから削除します。
- *config-name* は、削除する論理ドメイン構成の名前です。

リスト

ドメインおよび状態の一覧表示

次のサブコマンドは、論理ドメインおよびその状態のリストを表示します。論理ドメインを指定しない場合、すべての論理ドメインが表示されます。

```
ldm ls-dom [-e] [-l] [-o format] [-p] [ldom...]
```

ここでは、次のように指定します。

- `-e` は、自動的に設定されるサービスおよびデバイスを含む拡張リストを生成します。これは制御できません。
- `-l` は、長いリストを生成します。
- `-o` は、出力の *format* を、次に示すサブセットのうち 1 つ以上に制限します。複数の形式を指定する場合は、各形式をコンマで区切ります。スペースは入れません。
 - `console` - 出力には、仮想コンソール (`vcons`) および仮想コンソール端末集配信装置 (`vcc`) サービスが含まれます。
 - `core` - 出力には、コア、コア ID、および物理 CPU セットに関する情報が含まれます。
 - `cpu` - 出力には、仮想 CPU (`vcp`)、物理 CPU (`pcpu`)、およびコア ID (`cid`) に関する情報が含まれます。
 - `crypto` - 暗号化装置の出力には、モジュラー演算ユニット (`mau`)、およびサポートされているその他の暗号化装置 (Control Word Queue、CWQ) などが含まれます。
 - `disk` - 出力には、仮想ディスク (`vdisk`) および仮想ディスクサーバー (`vds`) が含まれます。
 - `domain` - 出力には、変数 (`var`)、ホスト ID (`hostid`)、ドメインの状態、フラグ、汎用一意識別子 (Universally Unique Identifier、UUID)、ソフトウェアの状態、利用率 (%)、スレーブのマスタードメイン、およびマスタードメインの障害ポリシーが含まれます。
 - `memory` - 出力には、メモリー (`memory`) が含まれます。
 - `network` - 出力には、メディアアクセス制御 (`mac`) アドレス、仮想ネットワークスイッチ (`vsw`)、および仮想ネットワーク (`vnet`) デバイスが含まれます。
 - `physio` - 物理入出力には、Peripheral Component Interconnect (`pci`) およびネットワークインタフェースユニット (`niu`) が含まれます。
 - `resgmt` - 出力は、動的資源管理 (Dynamic Resource Management、DRM) ポリシー情報を含み、現在実行中のポリシーを示し、`whole-core` および `max-core` 制約が有効であるかどうかを示します。
 - `serial` - 出力には、仮想論理ドメインチャンネル (`vldc`) サービス、仮想論理ドメインチャンネルクライアント (`vldcc`)、仮想データプレーンチャンネルクライアント (`vdpccl`)、および仮想データプレーンチャンネルサービス (`vdpcs`) が含まれます。
 - `status` - 出力には、移行中のドメインの状態が含まれます。
- `-p` は、解析可能でマシンが読み取り可能な形式でリストを生成します。
- `ldom` は、状態情報を表示する論理ドメインの名前です。

ドメインのバインドの一覧表示

次のサブコマンドは、論理ドメインのバインドのリストを表示します。論理ドメインを指定しない場合、すべての論理ドメインが表示されます。

```
ldm ls-bindings [-e] [-p] [ldom...]
```

ここでは、次のように指定します。

- **-e** は、自動的に設定されるサービスおよびデバイスを含む拡張リストを生成します。これは制御できません。
- **-p** は、解析可能でマシンが読み取り可能な形式でリストを生成します。
- **ldom** は、バインド情報を取得する論理ドメインの名前です。

ドメインのサービスの一覧表示

次のサブコマンドは、論理ドメインによってエクスポートされるすべてのサービスのリストを表示します。論理ドメインを指定しない場合、すべての論理ドメインが表示されます。

```
ldm ls-services [-e] [-p] [ldom...]
```

ここでは、次のように指定します。

- **-e** は、自動的に設定されるサービスおよびデバイスを含む拡張リストを生成します。これは制御できません。
- **-p** は、解析可能でマシンが読み取り可能な形式でリストを生成します。
- **ldom** は、サービス情報を取得する論理ドメインの名前です。

ドメインの制約の一覧表示

次のサブコマンドは、1つ以上の論理ドメインを作成するための制約のリストを表示します。論理ドメインを指定しない場合、すべての論理ドメインが表示されます。

```
ldm ls-constraints [-x] [ldom...]
```

```
ldm ls-constraints [-e] [-p] [ldom...]
```

ここでは、次のように指定します。

- **-x** は、XML形式の制約の出力を標準出力(stdout)形式で書き込みます。この出力は、バックアップとして使用できます。
- **ldom** は、制約を表示する論理ドメインの名前です。
- **-e** は、自動的に設定されるサービスおよびデバイスを含む拡張リストを生成します。これは制御できません。
- **-p** は、解析可能でマシンが読み取り可能な形式で制約の出力を書き込みます。

デバイスのリスト

次のサブコマンドは、使用していない(バインドされていない) 資源またはすべてのサーバー資源のいずれかのリストを表示します。デフォルトでは、使用していないすべての資源を表示します。

```
ldm ls-devices [-a] [-p] [core] [cpu] [crypto] [memory] [io]
```

ここでは、次のように指定します。

- **-a** は、すべてのサーバー資源(バインドされた資源およびバインドされていない資源)を表示します。
- **-p** は、解析可能でマシンが読み取り可能な形式で制約の出力を書き込みます。
- **core** は、コア、コア ID、および物理 CPU セットに関する情報を表示し、まだ割り当てられていないコア内の CPU を指定します。
- **cpu** は、仮想 CPU および物理 CPU 資源を表示します。
- **crypto** は、モジュラー演算ユニット資源のみを表示します。
- **memory** は、メモリー資源のみを表示します。
- **io** は、PCI バス、ネットワーク、またはダイレクト I/O 割り当て可能デバイスなど、入出力資源のみを表示します。

電源管理列 (PM) またはフィールド (pm=) で、yes は、仮想 CPU の電源が管理されていることを示します。no は、仮想 CPU の電源が投入されていることを示します。100% 使用されていない CPU は、デフォルトで電源が管理されます。

I/O デバイスの一覧表示

次のサブコマンドは、システム上で構成されている I/O デバイスのリストを表示します。デバイスのリストには、I/O バス (NIU を含む) およびダイレクト I/O 割り当て可能デバイスが含まれます。

出力は次のセクションに分かれます。

- **I/O バス情報。** IO 列にはバスまたはネットワークデバイスのデバイスパスが表示されます。PSEUDONYM 列には、バスまたはネットワークデバイスの関連付けられた仮名が表示されます。DOMAIN 列は、デバイスが現在バインドされているドメインを示します。
- **ダイレクト I/O 割り当て可能デバイス。** PCIE 列にはデバイスのデバイスパスが表示され、PSEUDONYM 列にはデバイスの関連付けられた仮名が表示されます。

STATUS 列は、プラグインカードを受け入れるスロットに適用されます。値は、OCC (配置済み)、EMP (空)、または UNK (不明) のいずれかです。ボード上のデバイスを示すスロットの状態は常に、occ になります。ルートドメインが静的ダイレクト I/O (Static Direct I/O、SDIO) 機能をサポートしない場合、スロットの状態は UNK です。SDIO 機能については、『Oracle VM Server for SPARC 2.0 管理ガイド』の「Using the Static Direct I/O Feature」を参照してください。

DOMAIN 列は、デバイスが現在バインドされているドメインを示します。このフィールドが空の場合、対応するデバイスはバインドされていません。このフィールドにハイフン文字 (-) が表示されている場合、次のいずれかの理由により、現在、デバイスを primary ドメインから削除できません。

- ルートドメインで実行されている OS では、SDIO 機能がサポートされていません。この場合、出力の PCIE 列に表示されるすべてのデバイスの STATUS フィールドは、UNK です。
- PCIe オプションカードが存在しないため、対応するスロットは空です。空のスロットをダイレクト I/O ドメインに割り当てることはできません。
- デバイスが接続されているルートコンプレックスは、分割 PCI 構成を使用して別のドメインに割り当てられます。この場合、デバイスの STATUS 列は UNK です。

ldm list-io [-l] [-p]

ここでは、次のように指定します。

- -l は、ダイレクト I/O 割り当て可能デバイスによりホストされるサブデバイスに関する情報を表示します。この出力は、ダイレクト I/O 割り当て可能デバイスによって受信ドメインに貸し出されるデバイスを示します。サブデバイス名は、コマンド入力のために使用することはできません。
- -p は、解析可能でマシンが読み取り可能な形式で出力を書き込みます。

論理ドメイン構成の一覧表示

次のサブコマンドは、サービスプロセッサ (Service Processor、SP) 上に格納される論理ドメインの構成のリストを表示します。

ldm ls-config [-r [autosave-name]]

-r [autosave-name] は、制御ドメインに存在する自動保存ファイルの構成を表示します。autosave-name が指定されている場合、autosave-name でのみレポートされます。出力には、対応する SP 構成よりも自動保存ファイルが新しいかどうかを示されます。

注-遅延再構成が保留中の場合は、構成の変更はただちに自動保存されます。そのため、ldm ls-config -r コマンドを実行すると、自動保存構成は現在の構成より新しいものとして表示されます。

変数の一覧表示

次のサブコマンドは、論理ドメインの 1 つ以上の変数のリストを表示します。ドメインのすべての変数を表示するには、var-name を空白のままにします。

ldm ls-var [var-name...] ldom

ここでは、次のように指定します。

- *var-name* は、表示する変数の名前です。名前を指定しない場合、そのドメインのすべての変数が表示されます。
- *ldom* は、1つ以上の変数を表示する論理ドメインの名前です。

資源管理ポ
リシーの追加、設
定、および削除

資源管理ポリシーの追加

このサブコマンドでは、1つ以上の論理ドメインの資源管理ポリシーを追加できます。資源管理ポリシーは、オプションのプロパティとそれらの値で構成されています。

電源管理がエラスティックモード「でない限り」、CPU DRをサポートするアクティブドメインで資源管理ポリシーを有効にできます。

```
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...
```

ここでは、次のように指定します。

- プロパティは「Properties」セクションに記述されています。
- *ldom* は、資源管理ポリシーを追加する論理ドメインを指定します。

資源管理ポリシーの変更

このサブコマンドでは、オプションのプロパティに値を指定して、1つ以上の論理ドメインの資源管理ポリシーを変更できます。

```
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...
```

ここでは、次のように指定します。

- プロパティは「Properties」セクションに記述されています。
- *ldom* は、資源管理ポリシーを変更する論理ドメインを指定します。

資源管理ポリシーの削除

このサブコマンドでは、1つ以上のポリシー名を指定して、論理ドメインから資源管理ポリシーを削除できます。

```
ldm remove-policy [name=]policy-name... ldom
```

ここでは、次のように指定します。

- `name` プロパティは、資源管理ポリシーの名前である *policy-name* を指定します。
- `ldom` は、資源管理ポリシーを削除する論理ドメインを指定します。

XML ファイルからの
ドメインの構成ま
たは再構成

次のサブコマンドでは、既存の構成を使用して、1つ以上のゲストドメインまたは制御ドメイン、あるいはこの両方の種類のドメインを構成できます。`ldm init-system` コマンドは、XML ファイル (`ldm ls-constraints -x` の出力など) を入力として取り込み、指定したドメインを構成し、制御ドメインを再起動します。出荷時のデフォルトの構成を使用してこのコマンドを実行します。

```
ldm init-system [-rs] -i file
```

ここでは、次のように指定します。

- `-i file` は、論理ドメインの作成に使用する XML 構成ファイルを指定します。
- `-r` は構成後にシステムを再起動します。
- `-s` は、仮想サービス構成 (`vds`、`vcc`、および `vsw`) のみを復元します。

使用例

例1 デフォルトのサービスの作成

3つのデフォルトのサービスである、仮想ディスクサーバー、仮想スイッチ、および仮想コンソール端末集配信装置を設定して、これらのサービスをゲストドメインにエクスポートできるようにします。

```
# ldm add-vds primary-vds0 primary
# ldm add-vsw net-dev=nxge0 primary-vsw0 primary
# ldm add-vcc port-range=5000-5100 primary-vcc0 primary
```

例2 サービスの一覧表示

サービスのリストを表示して、サービスが正常に作成されたこと、または使用可能なサービスを確認することができます。

```
# ldm ls-services primary
VCC
  NAME          LDOM    PORT-RANGE
  primary-vcc0  primary 5000-5100
VSW
  NAME          LDOM    MAC          NET-DEV  DEVICE      DEFAULT-VLAN-ID PVID VID MODE
  primary-vsw0  primary 00:14:4f:f9:68:d0 nxge0    switch@0  1          1
VDS
  NAME          LDOM    VOLUME      OPTIONS      MPGROUP     DEVICE
  primary-vds0  primary
```

例3 制御ドメインの初期設定

制御ドメインは、`primary` と呼ばれ、Logical Domains Manager のインストール時に存在する初期ドメインです。制御ドメインではすべての資源を利用でき、それらの資源は使用しているサーバーによって異なります。制御ドメインで維持する資源の

例3 制御ドメインの初期設定 (続き)

みを設定し、残りの資源をゲストドメインに割り当てられるようにします。次に、構成をサービスプロセッサに保存します。再起動して変更を有効にする必要があります。

制御ドメインとその他のドメイン間のネットワークを使用可能にする場合、制御ドメインで仮想スイッチを `plumb` します。ゲストドメインでコンソールを使用するには、仮想ネットワーク端末サーバーデーモン (`vntsd(1M)`) を使用可能にする必要があります。

```
# ldm start-reconf primary
# ldm set-crypto 1 primary
# ldm set-vcpu 4 primary
# ldm set-mem 4G primary
# ldm add-config initial
# shutdown -y -g0 -i6
# ifconfig -a
# ifconfig vsw0 plumb
# ifconfig nxge0 down unplumb
# ifconfig vsw0 IP-of-nxge0 netmask netmask-of-nxge0 broadcast + up
# svcadm enable vntsd
```

例4 バインドの一覧表示

バインドのリストを表示して、指定した資源が制御ドメインにあるかどうか、または任意のドメインにバインドされている資源を確認できます。

```
# ldm ls-bindings primary
```

NAME	STATE	FLAGS	CONS	VCPU	MEMORY	UTIL	UPTIME
primary	active	-t-cv		4	4G	12%	11m

MAC

```
08:00:90:11:11:10
```

VCPU

VID	PID	UTIL	STRAND
0	0	18%	100%
1	1	13%	100%
2	2	9.8%	100%
3	3	5.4%	100%

MEMORY

RA	PA	SIZE
0x4000000	0x4000000	4G

IO

DEVICE	PSEUDONYM	OPTIONS
pci@400	pci_0	

例4 バインドの一覧表示 (続き)

```

pci@500          pci_1
pci@400/pci@0/pci@9 PCIE2
pci@400/pci@0/pci@9 MB/SASHBA
pci@500/pci@0/pci@8 MB/NET0

VCC
  NAME          PORT-RANGE
  primary-vcc0   5000-5100

VSW
  NAME          MAC          NET-DEV  DEVICE  MODE
  primary-vsw0   00:14:4f:f9:68:d0 nxge0    switch@0 prog,promisc

VDS
  NAME          VOLUME          OPTIONS          DEVICE
  primary-vds0

```

例5 論理ドメインの作成

必要とするゲストドメイン構成を作成するための資源の存在の確認、ゲストドメインの追加、ドメインに必要な資源およびデバイスの追加、起動時の動作をシステムに指示する起動パラメータの設定、ドメインへの資源のバインド、バックアップ用のXMLファイルへのゲストドメインの構成の保存を実行します。また、primaryドメインおよびゲストドメインの構成をSCに保存する場合があります。その後、ドメインを起動し、ドメインのTCPポートを検出し、デフォルトの仮想コンソールサービスを介してそのTCPポートに接続することができます。

```

# ldm ls-devices
# ldm add-dom ldg1
# ldm add-vcpu 4 ldg1
# ldm add-mem 1g ldg1
# ldm add-vnet vnet1 primary-vsw0 ldg1
# ldm add-vdsdev /dev/dsk/c0t1d0s2 vol1@primary-vds0
# ldm add-vdisk vdisk1 vol1@primary-vds0 ldg1
# ldm set-var auto-boot\?=false ldg1
# ldm set-var boot-device=vdisk1 ldg1
# ldm bind-dom ldg1
# ldm ls-constraints -x ldg1 > ldg1.xml
# ldm add-config ldg1_4cpu_1G
# ldm start ldg1
# ldm ls -l ldg1
# telnet localhost 5000

```

例6 多数のゲストドメインに対する1つの端末の使用

通常、作成した各ゲストドメインには、そのドメイン専用のTCPポートおよびコンソールがあります。1つめのゲストドメイン(この例では、ldg1)を作成したあと

例6 多数のゲストドメインに対する1つの端末の使用 (続き)

は、`ldm set-vcons` コマンドを使用して、その他すべてのドメイン (この例での2つめのドメインは `ldg2`) を同じコンソールポートに接続できます。`set-vcons` サブコマンドは、アクティブでないドメインでのみ機能します。

```
# ldm set-vcons group=ldg1 service=primary-vcc0 ldg2
```

1つめ以外のすべてのゲストドメインで `set-vcons` コマンドを実行したあとに、`ldm ls -l` コマンドを実行すると、すべてのドメインが同じポートに接続していることを確認できます。コンソールの使用法については、[vntsd\(1M\)](#) マニュアルページを参照してください。

例7 論理ドメインへの仮想PCIバスの追加

I/O ドメインは、物理 I/O デバイスの直接所有権を持ち、これらに直接アクセスできるサービスドメインの一種です。I/O ドメインは、仮想 I/O デバイスの形式でゲストドメインにサービスを提供します。この例では、論理ドメインに仮想 PCI バスを追加する方法について示します。

```
# ldm add-io pci@7c0 ldg1
```

例8 仮想データプレーンのチャンネル機能の追加 (Netra のみ)

使用しているサーバーに Netra Data Plane Software (NDPS) 環境がある場合、仮想データプレーンのチャンネル機能を追加することがあります。まず、サービスドメインに仮想データプレーンのチャンネルサービス (`primary-vdpcs0` など) を追加します。この場合のサービスドメインは、`primary` ドメインです。

```
# ldm add-vdpcs primary-vdpcs0 primary
```

サービスドメイン (`primary`) にサービスを追加したら、ゲストドメイン (`ldg1`) に仮想データプレーンのチャンネルクライアント (`vdpccl1`) を追加できます。

```
# add-vdpccl vdpccl1 primary-vdpcs0 ldg1
```

例9 制御ドメインの遅延再構成処理の取り消し

1つの遅延再構成処理によって、その他すべてのドメインの構成処理がブロックされます。制御ドメインの遅延再構成処理の取り消しが必要になる場合があります。たとえば、目的のドメインまたは他のドメインでほかの構成コマンドを実行できるように、処理を取り消す場合があります。このコマンドを使用すると、遅延再構成処理を取り消して、目的のドメインまたは他のドメインでほかの構成処理を行うことができます。

```
# ldm cancel-op reconf primary
```

例10 ドメインの移行

論理ドメインは、別のマシンに移行することができます。次に、成功する移行の例を示します。

例10 ドメインの移行 (続き)

```
# ldm migrate ldg1 root@dt90-187:ldg
Target password:
```

例11 構成の一覧表示

次の例は、構成を表示する方法を示しています。1つめのコマンドは、SPに格納されている構成を表示します。2つめのコマンドは、SP上の構成と、制御ドメイン上の自動保存構成に関する情報を表示します。

```
# ldm ls-config
factory-default
3guests [current]
data1
reconfig_primary
split1
# ldm ls-config -r
3guests [newer]
data1 [newer]
reconfig_primary
split1
unit
```

現在の3guests構成とdata1構成の両方で、SPに保存されていない変更が自動保存されています。この状態のときにシステムの電源が再投入されると、Logical Domains Managerは指定の手順に基づき、自動保存された3guestsの構成を回復します。3guestsがcurrentとしてマークされているため、自動保存の回復処理はこのシステムに対して実行されます。

reconfig_primaryおよびsplit1自動保存構成は、SP上のバージョンと同一で、より新しいバージョンではありません。

unit構成は、自動保存構成として制御ドメインにのみ存在します。unitに対応する構成は、SP上には存在しません。この状況は、SPで構成が失われると発生することがあります。SPを交換した場合、またはSP上の持続的なバージョンの構成に問題が発生した場合、構成が失われる可能性があります。rm-configコマンドを使用して構成を明示的に削除すると、制御ドメインの自動保存構成も削除されます。その結果、制御ドメインにもSPにも構成は残りません。

例12 I/Oデバイスの一覧表示

次の例では、システム上のI/Oデバイスを一覧表示します。最初のセクションには、primaryドメイン上にあるPCIeバスに関する情報が表示されます。2つ目のセクションには、ダイレクトI/O割り当て可能デバイスに関する情報が表示されます。

```
# ldm ls-io
IO                PSEUDONYM        DOMAIN
```

例 12 I/O デバイスの一覧表示 (続き)

```
--
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      ldg1
pci@400/pci@/pci@d PCIE3      OCC      ldg2
pci@400/pci@/pci@8 MB/SASHBA  OCC      primary
pci@500/pci@/pci@9 PCIE0      EMP      -
pci@500/pci@/pci@d PCIE4      OCC      ldg2
pci@500/pci@/pci@c PCIE5      OCC      ldg1
pci@500/pci@/pci@8 MB/NET0    OCC      primary
```

終了ステータス 次の終了値が返されます。

- 0 正常に完了しました。
- >0 エラーが発生しました。

属性 次の属性の説明については、[attributes\(5\)](#) マニュアルページを参照してください。

属性型	属性値
使用条件	SUNWldm
インタフェースの安定性	未確定

関連項目 [dumpadm\(1M\)](#)、[ifconfig\(1M\)](#)、[shutdown\(1M\)](#)、[vntsd\(1M\)](#)、[attributes\(5\)](#)

『Oracle VM Server for SPARC 2.0 管理ガイド』

名前	ldmconfig – Oracle VM Server for SPARC Configuration Assistant
形式	ldmconfig [-cdh]
機能説明	<p>Oracle VM Server for SPARC Configuration Assistant である ldmconfig ユーティリティーは、Oracle VM Server for SPARC を実行可能なシステムを効率良く設定する端末ベースのアプリケーションです。Oracle VM Server for SPARC ソフトウェアの実行に使用できるのは、チップマルチスレッド (Chip Multi-Threaded-based、CMT) ベースのシステムのみです。</p> <p>ldmconfig は、システムを検査して、有効な構成を生成するためのデフォルトの一連の選択肢をユーザーに提供します。設定プロパティー値の収集後、ldmconfig は、論理ドメインの設定に適した構成を作成します。</p> <p>ldmconfig ユーティリティーは、コンソール接続、リモート端末エミュレータ、または ssh セッションを使用して実行できます。</p> <p>Configuration Assistant は、次のオプションを使用します。</p> <ul style="list-style-type: none">-c Oracle Solaris OS メディアに有効なパッケージがあるかどうかを確認します。-d デバッグモードを指定します。完了後、実行ログおよびエラーログが保持されます。-h 使用法メッセージを表示します。 <p>終了ステータス 次の終了値が返されます。</p> <ul style="list-style-type: none">0 正常に完了しました。>0 エラーが発生しました。 <p>属性 次の属性の説明については、attributes(5) マニュアルページを参照してください。</p>

属性型	属性値
使用条件	SUNWconfig
インタフェースの安定性	未確定

関連項目 [ldm\(1M\)](#)、[attributes\(5\)](#)

『Oracle VM Server for SPARC 2.0 管理ガイド』

名前	ldmd – Logical Domains Manager デーモン
形式	/opt/SUNWldm/bin/ldmd
機能説明	<p>ldmd デーモンは Logical Domains Manager と呼ばれます。これは、論理ドメインの作成および管理に使用される、ldm コマンド用のデーモンプログラムです。Logical Domains Manager は、サーバーごとに 1 つだけ存在できます。ldmd デーモンは、サービスプロセッサ (Service Processor, SP) により作成される初期ドメインである、制御ドメイン上で実行されます。制御ドメインの名前は primary です。</p> <p>論理ドメインとは、独自のオペレーティングシステム、リソース、識別情報を単一システム内に持つ個別の論理グループです。各論理ドメインは、サーバーの電源の再投入を必要とせずに、作成、削除、再構成、および再起動を単独で行うことができます。セキュリティ上の理由から、論理ドメインを使用してさまざまなアプリケーションを異なるドメインで動作させて、アプリケーションの独立性を維持することができます。</p>
SMF プロパティー	<p>svcadm コマンドで、次のプロパティーを変更できます。</p> <p>ldmd/autorecovery_policy</p> <p>自動回復ポリシーを指定します。このプロパティーには、次の値のいずれかを指定できます。</p> <ul style="list-style-type: none"> ■ autorecovery_policy=1 – 自動保存構成が、対応する実行中の構成よりも新しい場合に、警告メッセージをログに記録します。これらのメッセージは、ldmd SMF ログファイルに記録されます。ユーザーは、構成の回復を手動で実行する必要があります。これはデフォルトのポリシーです。 ■ autorecovery_policy=2 – 自動保存構成が、対応する実行中の構成よりも新しい場合に、通知メッセージを表示します。この通知メッセージは、毎回の Logical Domains Manager の再起動後に最初に ldm コマンドが発行されたときに、いずれかの ldm コマンドの出力になります。ユーザーは、構成の回復を手動で実行する必要があります。 ■ autorecovery_policy=3 – 自動保存構成が、対応する実行中の構成よりも新しい場合に、構成を自動的に更新します。この処理により、次の電源再投入時に使用される SP 構成が書き換えられます。この構成は、制御ドメインに保存されている、より新しい構成で更新されます。この処理は、現在実行中の構成には影響を与えません。この処理は、次の電源再投入時に使用される構成にのみ影響します。新しい構成が SP 上で保存され、次回システムで電源再投入を実行したときに起動されるというメッセージも記録されます。これらのメッセージは、ldmd SMF ログファイルに記録されます。 <p>ldmd/incoming_migration_enabled</p> <p>xmpp_enabled も true に設定されている場合、別のシステムからこのシステムへのゲストドメインの移行を有効にします。デフォルト値は true です。</p> <p>ldmd/outgoing_migration_enabled</p> <p>xmpp_enabled も true に設定されている場合、このシステムから別のシステムへのゲストドメインの移行を有効にします。デフォルト値は true です。</p>

`ldmd/xmpp_enabled`
ldmd XMPP サーバーが他社製の管理アプリケーションからの構成要求を待機できるようにします。また、2つのシステム間の移行を調整するために、ldmd デーモンが別のシステム上の ldmd デーモンと通信できるようにします。デフォルト値は true です。

属性 次の属性の説明については、[attributes\(5\)](#) マニュアルページを参照してください。

属性型	属性値
使用条件	SUNWldm
インタフェースの安定性	未確定

関連項目 [svcs\(1\)](#)、[drd\(1M\)](#)、[ldm\(1M\)](#)、[ldmad\(1M\)](#)、[svcadm\(1M\)](#)、[vntsd\(1M\)](#)、[attributes\(5\)](#)、[smf\(5\)](#)

注意事項 ldmd サービスは、サービス管理機能 (Service Management Facility、SMF) により管理され、`svc:/ldoms/ldmd:default` サービス識別子を使用します。[smf\(5\)](#) マニュアルページを参照してください。

`svcadm` コマンドを使用して、再起動の有効化、無効化、または要求などの管理操作をこのサービスで実行します。`svcs` コマンドを使用してサービスの状態をクエリーします。

ldmd SMF プロパティについては、『[Oracle VM Server for SPARC 2.0 管理ガイド](#)』を参照してください。

名前	ldmp2v – Oracle VM Server for SPARC Physical-to-Virtual (P2V) 移行ツール用のコマンド行インタフェース
形式	<pre>ldmp2v collect [-a flash none] [-O "flarcreate-options"] [-v] [-x mount-point [-x ...]] -d data-dir ldmp2v prepare [-b zvol file disk] [-B backend:volume:vdisk [-B ...]] [-c cpu] [-m mount-point:size [-m ...]] [-M memsize] [-o keep-hostid] [-o keep-mac] [-p prefix] [-s] [-v] [-x no-auto-adjust-fs] [-x remove-unused-slices] -d data-dir domain ldmp2v prepare -R guest-root [-c cpu] [-M memsize] [-o keep-hostid] [-o keep-mac] [-v] -d data-dir domain ldmp2v prepare -C domain ldmp2v convert -i install-image -d data-dir [-v] domain ldmp2v convert [-j] -n interface -d data-dir [-v] domain</pre>
機能説明	<p>Oracle VM Server for SPARC Physical-to-Virtual (P2V) 移行ツールのバージョン 1.0 は、既存の物理システムを、チップマルチスレッディング (Chip Multi-Threading, CMT) システム上の論理ドメインで実行する仮想システムに自動的に変換します。ソースシステムには、Solaris 8 以降のオペレーティングシステムを実行する sun4u SPARC システム、または Solaris 10 OS を実行する非 Logical Domains sun4u システムを指定できます。</p> <p>物理システムから仮想システムへの変換は、次のフェーズで実行されます。</p> <ul style="list-style-type: none"> ■ 収集フェーズ。物理ソースシステムで実行されます。collect は、ソースシステムに関して収集した構成情報に基づいて、ソースシステムのファイルシステムイメージを作成します。 ■ 準備フェーズ。ターゲットシステムの制御ドメインで実行されます。prepare は、collect フェーズで収集された構成情報に基づいて、ターゲットシステムに論理ドメインを作成します。ファイルシステムイメージは、1 つ以上の仮想ディスクに復元されます。このイメージは、論理ドメインとして動作できるように変更されます。 ■ 変換フェーズ。ターゲットシステムの制御ドメインで実行されます。convert フェーズで、標準の Solaris アップグレードプロセスを使用することにより、作成された論理ドメインが、Solaris 10 OS を実行する論理ドメインに変換されます。 <p>次の節からは、物理システムから仮想システムへの変換が各フェーズで実行される方法について説明します。</p>
収集フェーズ	<pre>ldmp2v collect [-a flash none] [-O "flarcreate-options"] [-v] [-x mount-point [-x ...]] -d data-dir</pre> <p>ldmp2v collect コマンドは、次のオプションを使用します。</p> <p>-a flash none 使用するアーカイブ方法を指定します。有効な値は、flash または none です。デフォルトは flash です。</p>

- O "flarcreate-options"** flarcreate コマンドに渡すオプションの引用符付きのリストを指定します。flarcreate のオプションのうち許可されているものは、**-c** と **-x** のみです。**-c** オプションはアーカイブを圧縮し、**-x** オプションはアーカイブからファイルまたはディレクトリを除外します。複数の flarcreate オプションを指定できます。**-O** オプションは、**-a flash** を使用してフラッシュアーカイブ方法を指定する場合にのみ使用できます。
- v** 詳細モードを使用します。このモードでは、ldmp2v によって発行されるメッセージがより詳細になります。
- x mount-point** *mount-point* でマウントされるファイルシステムをアーカイブから除外します。
- d data-dir** P2V ファイルを格納するディレクトリをシステムごとに指定します。収集フェーズでは、このディレクトリは、**root** による書き込みが可能である必要があります。中間ディレクトリは、自動的に作成されます。

準備フェーズ

```
ldmp2v prepare [-b zvol|file|disk] [-B backend:volume:vdisk [-B ...]] [-c cpu]
  [-m mount-point:size [-m ...]] [-M memsize] [-o keep-hostid] [-o keep-mac]
  [-p prefix] [-s] [-v] [-x no-auto-adjust-fs] [-x remove-unused-slices]
  -d data-dir domain
ldmp2v prepare -R guest-root [-c cpu] [-M memsize] [-o keep-hostid] [-o keep-mac]
  [-v] -d data-dir domain
ldmp2v prepare -C domain
```

ldmp2v prepare コマンドは、次のオペランドおよびオプションを使用します。

- domain** 操作の対象となる論理ドメインを指定します。
- b zvol|file|disk** 使用するバックエンドの種類を指定します。仮想ディスクは、ZFS ボリューム (**zvol**)、プレーンファイル (**file**)、物理ディスクまたはボリュームマネージャーボリューム (**disk**) で構成されます。このオプションは、**/etc/ldmp2v.conf** 内の **BACKEND_TYPE** の設定を上書きします。
- B backend:volume:vdisk** バックエンドデバイスの名前を指定し、任意で、作成するボリュームおよび仮想ディスクの名前を指定します。**volume** または **vdisk** 値が省略されている場合、デフォルトの名前が使用されます。省略する各値にコロン文字 (:) を指定することにより、値を省略できます。たとえば、**-B ::vdisk01** と **-B :volume001** が挙げられます。

このオプションは、disk バックエンドに必要であり、`/dev/dsk/c0t2d0s2` または `/dev/md/dsk/d100` のように少なくともバックエンドデバイスを指定する必要があります。disk バックエンドについては、物理システムのマニフェスト内に存在する各ディスクに対して `-B` オプションを 1 つ指定します。

zvol および file バックエンドについては、*backend* を使用して、ldmp2v が仮想ディスク用に作成するファイルまたは ZFS データセットを指定できます。たとえば、`-B data/ldom1/disk0` のように指定します。`-B` オプションを使用して、バックエンド名を指定し、デフォルト名を上書きします。デフォルト名は、`-p` オプションによって生成されるか、`/etc/ldmp2v.config` 内の `BACKEND_PREFIX` 設定とドメイン名によって生成されます。

- `-c cpu` VCPU の数を論理ドメインに割り当てます。デフォルトでは、ldmp2v は、物理システム上の各 CPU に 1 つの VCPU を割り当てます。
- `-C` 指定したドメインをクリーンアップします。
- `-d data-dir` P2V に必要なファイルが配置されるディレクトリをシステムごとに指定します。
- `-m mount-point:size` *mount-point* で、ファイルシステムのベースとなるスライスとディスクのサイズを変更します。サイズは *numunit* として指定されます。*num* は容量であり、*unit* はブロックの場合は `b`、K バイトの場合は `k`、M バイトの場合は `m`、ギガバイトの場合は `g` です。このオプションは、複数回指定できます。このオプションは、`/`、`/usr`、および `/var` の自動サイズ変更を無効にします。*mount-point* が `swap` である場合、最初に構成されたスワップデバイスが *size* の値にサイズ変更されます。
- `-M memsize` 論理ドメインに割り当てるメモリー量を M バイト単位で指定します。デフォルトでは、ldmp2v は、物理システムと同じ量を割り当てます。
- `-o keep-hostid` 物理システムのホスト ID を論理ドメインに転送します。デフォルトで、Logical Domains Manager は新しい一意のホスト ID を割り当てます。
- `-o keep-mac` 物理システムの MAC アドレスを論理ドメインに転送します。デフォルトで、Logical Domains Manager は新しい一意の MAC アドレスを割り当てます。

- p *prefix* バックエンドデバイスを作成する場所を指定します。zvol バックエンドの場合は ZFS データセット、file バックエンドの場合は / からの相対ディレクトリを指定します。このオプションは、/etc/ldmp2v.conf の BACKEND_PREFIX パラメータより優先されます。
- R *guest-root* 非自動モードを選択します。OS イメージの変更手順は、*guest-root* をルートとするファイルシステムに適用されます。論理ドメインの /etc/vfstab を更新して、*guest-root* 配下のファイルシステムのレイアウトに一致させます。
- s スパースバックエンドデバイスを作成します。このオプションは、/etc/ldmp2v.conf の BACKEND_SPARSE パラメータより優先されます。
- v 詳細モードを使用します。このモードでは、ldmp2v によって発行されるメッセージがより詳細になります。
- x no-auto-adjust-fs /、/usr、および/var ファイルシステムが自動サイズ調整で合計 10G バイトにならないようにします。既存のファイルシステムのサイズは新しい Solaris リリースにアップグレードするには十分でない場合があるため、このオプションは慎重に使用してください。

-m オプションを使用すると、ファイルシステムのサイズを手動で変更できます。
- x remove-unused-slices ファイルシステムまたはスワップデバイスを保持しないスライスを作成しないことにより、仮想ディスクのサイズを減らします。

変換フェーズ

```
ldmp2v convert -i install-image -d data-dir [-v] domain
ldmp2v convert [-j] -n interface -d data-dir [-v] domain
```

ldmp2v convert コマンドは、次のオプションを使用します。

- d *data-dir* P2V に必要なファイルが配置されるディレクトリをシステムごとに指定します。
- i *install-image* アップグレードに使用する Solaris 10 OS DVD ISO イメージへのパスを指定します。
- j Custom JumpStart を使用します。この場合、JumpStart サーバーおよび JumpStart クライアントが適切に構成されている必要があります。
- n *interface* ネットワークインストールサーバーを使用する場合に起動する仮想ネットワークインタフェースを指定します。

- v 詳細モードを使用します。このモードでは、ldmp2v によって発行されるメッセージがより詳細になります。

注意-変換フェーズを開始する前に、元の物理システムを停止してください。これは、論理ドメインが、物理システムと同じ IP アドレスを使用し、場合によっては同じ MAC アドレスを使用するためです。

物理システムのいずれかの IP アドレスがアクティブな場合、ldmp2v convert コマンドはエラーメッセージを表示して終了します。

使用例

この節では、3つのフェーズの例が含まれます。

例1 収集フェーズの例

ldmp2v collect コマンドの使用方法的例を次に示します。

- NFS マウント済みファイルシステムを共有する。次の例は、collect フェーズの簡単な実行方法を示しています。この場合、ソースシステムとターゲットシステムは、1つの NFS マウント済みファイルシステムを共有します。

```
# ldmp2v collect -d /home/dana/p2v/volumia
```

- NFS マウント済みファイルシステムを共有しない。ソースシステムとターゲットシステムが1つの NFS マウント済みファイルシステムを共有しない場合、ファイルシステムイメージをローカル記憶領域に書き込んだあとで制御ドメインにコピーできます。ldmp2v により提供されるフラッシュアーカイブ方法を使用します。フラッシュツールは、作成したアーカイブを自動的に除外します。

```
# ldmp2v collect -d /home/dana/p2v/volumia -a flash
```

- ファイルシステムのバックアップステップをスキップする。NetBackup など、他社のバックアップツールを使用することでシステムのバックアップをすでに利用できる場合は、none アーカイブ方式を使用してファイルシステムのバックアップステップをスキップできます。このオプションを使用する場合、システム構成マニフェストのみが作成されます。

```
# ldmp2v collect -d /home/dana/p2v/volumia -a none
```

注--d で指定するディレクトリが、ソースシステムとターゲットシステムによって共有されていない場合は、そのディレクトリの内容を制御ドメインにコピーします。準備フェーズを開始する前に、ディレクトリの内容を制御ドメインにコピーする必要があります。

- フラッシュアーカイブからファイルまたはディレクトリを除外する。フラッシュアーカイブ方法を使用する場合、flarcreate コマンドにオプションを渡すことにより、ファイルまたはディレクトリをアーカイブから除外できます。この機能では、少なくとも次のパッチバージョンがソースシステムにインストールされている必要があります。

例1 収集フェーズの例 (続き)

- **Solaris 8 OS:** パッチ ID 109318-34
 - **Solaris 9 OS:** パッチ ID 113434-06
- ```
ldmp2v collect -d /home/dana/p2v/volumia -a flash
-O "-x /path/to/file -x /some/dir"
```

## 例2 準備フェーズの例

ldmp2v prepare コマンドの使用方法的例を次に示します。

- 次の例は、物理システムの MAC アドレスを保持しながら、/etc/ldmp2v.conf に構成されているデフォルトを使用することで、volumia という論理ドメインを作成します。

```
ldmp2v prepare -d /home/dana/p2v/volumia -o keep-mac volumia
```

- 次の例は、-c オプションを使用して、ドメインとそのバックエンドデバイスを完全に削除する方法を示しています。

```
ldmp2v prepare -C volumia
```

- 次の例は、-m オプションを使用して、P2V 中のファイルシステムとスワップデバイスのサイズ変更を行う方法を示しています。

```
ldmp2v prepare -d /home/dana/p2v/normaal -m /:8g -m swap:4g normaal
```

- 次の例は、Solaris ボリュームマネージャメタデバイス d100 と d101 をゲストドメインのバックエンドデバイスとして使用し、最初の仮想ディスクの名前を vdisk100 に設定する方法を示しています。

```
ldmp2v prepare -b disk -B /dev/md/dsk/d100::vdisk100 -B /dev/md/dsk/d101
-d /p2v/volumia volumia
```

- 次の例は、デフォルトでない ZFS ボリューム名が付いた ZFS ボリュームを使用する方法を示しています。

```
ldmp2v prepare -b zvol -B tank/ldom1/zvol1 -B tank/ldom1/zvol2 -d /p2v/volumia
volumia
```

## 例3 変換フェーズの例

ldmp2v convert コマンドの使用方法的例を次に示します。

- ネットワークインストールサーバーの使用。ldmp2v convert コマンドは、指定された仮想ネットワークインタフェースを使用して、ネットワークを通じて論理ドメインを起動します。インストールサーバーで setup\_install\_server および add\_install\_client スクリプトを実行する必要があります。

Custom JumpStart 機能を使用し、完全に操作不要の変換を実行することもできます。



例3 変換フェーズの例 (続き)

次の例は、ネットワークインストールサーバーを使用してシステムをアップグレードする方法を示しています。

```
ldmp2v convert -n vnet0 -d /p2v/volumia volumia
```

次の例は、Custom JumpStart を使用してシステムをアップグレードする方法を示しています。

```
ldmp2v convert -j -n vnet0 -d /p2v/volumia volumia
```

- ISO イメージの使用。ldmp2v convert コマンドは、Solaris DVD ISO イメージを論理ドメインに接続し、そこから ISO イメージを起動します。アップグレードを行うには、sysid のすべての質問に回答し、「Upgrade」を選択します。

注 - sysid の質問への回答は、アップグレード処理時にのみ使用されるため、もっとも単純なオプション(ネットワーク接続なし、ネームサービスなし、など)を選択できます。システムの元の ID は、アップグレードによって維持され、アップグレードの完了後に再起動すると有効になります。アップグレードの実行に要する時間は、元のシステムにインストールされている Solaris クラスタによって決まります。

```
ldmp2v convert -i /tank/iso/s10s_u5.iso -d /home/dana/p2v/volumia volumia
```

終了ステータス 次の終了値が返されます。

- 0           正常に完了しました。
- >0         エラーが発生しました。

属性           次の属性の説明については、[attributes\(5\)](#) マニュアルページを参照してください。

|             |            |
|-------------|------------|
| 属性型         | 属性値        |
| 使用条件        | SUNWldmp2v |
| インタフェースの安定性 | 未確定        |

関連項目      [ldm\(1M\)](#)、[attributes\(5\)](#)

『Oracle VM Server for SPARC 2.0 管理ガイド』

名前 virtinfo – 仮想化ドメイン情報

形式 virtinfo [ -acdpstu ]

機能説明 virtinfo ユーティリティーは、libv12n(3LIB) により返される、現在のドメインに関する仮想化情報を提供します。オプションが指定されている場合、1 つ以上の仮想化ドメイン特性を表す記号が標準出力に書き出されます。オプションが指定されていない場合、virtinfo は仮想化ドメインの種類のみを示します。

オプション virtinfo コマンドは次のオプションをサポートしています。

- a ドメインについて現在分かっている情報をすべて表示します。
- c 制御ドメインのネットワークノード名を表示します。この名前は、制御ドメイン上で `uname -n` コマンドを実行することにより表示される文字列と同一です。
- d この仮想ドメインのドメイン名を表示します。
- p 解析可能な形式で情報を表示します。

Logical Domains (LDoms) の解析可能な出力は次の形式になります。

```
VERSION 1.0
DOMAINROLE|impl=LDoms|control={true|false}|
 io={true|false}|service={true|false}|
 root={true|false}
DOMAINNAME|name=domain-name
DOMAINUUID|uuid=uuid
DOMAINCONTROL|name=control-domain-nodename
DOMAINCHASSIS|serialno=serial-no
```

- s プラットフォームシリアル番号を表示します。
- t ドメインの種類を表示します。

Logical Domains のドメインの種類は、次の空白文字で区切った値の 1 つまたは複数の文字列です。

|              |              |
|--------------|--------------|
| LDoms[LDoms] | LDoms 実装     |
| 制御           | 制御ドメイン       |
| ゲスト          | ゲストドメイン      |
| I/O          | I/O ドメイン     |
| service      | サービスドメイン     |
| ルート (root)   | ルート I/O ドメイン |

control 値と guest 値は相互排他的です。

-u      ドメインの汎用一意識別子 (Universally Unique Identifier、UUID) を表示します。[libuuid\(3LIB\)](#) マニュアルページを参照してください。

終了ステータス      次の終了値が返されます。

0              正常に完了しました。  
>0              エラーが発生しました。

## 使用例

例1      ドメインに関するデフォルト情報の表示

次の例は、ドメインに関するデフォルト情報を示しています。

```
$ virtinfo
Domain role: LDoms control I/O service root

The current domain is the control domain, which is also an
I/O domain, the service domain and a root I/O domain.
```

例2      ドメインに関するすべての情報の表示

次の例は、ドメインに関するすべての使用可能な情報を示しています。

```
$ virtinfo -a
Domain role: LDoms control I/O service root
Domain name: primary
Domain UUID: 8e0d6ec5-cd55-e57f-ae9f-b4cc050999a4
Control domain: san-t2k-6
Chassis serial#: 0704RB0280
```

例3      解析可能な形式によるドメインに関するすべての情報の表示

次の例は、解析可能な形式で、ドメインに関するすべての使用可能な情報を示しています。

```
$ virtinfo -ap
VERSION 1.0
DOMAINROLE|impl=LDoms|control=true|io=true|service=true|root=true
DOMAINNAME|name=primary
DOMAINUUID|uuid=8e0d6ec5-cd55-e57f-ae9f-b4cc050999a4
DOMAINCONTROL|name=san-t2k-6
DOMAINCHASSIS|serialno=0704RB0280
```

## 属性

次の属性の説明については、[attributes\(5\)](#) を参照してください。

| 属性型         | 属性値     |
|-------------|---------|
| 使用条件        | SUNWcsu |
| インタフェースの安定性 | 未確定     |

関連項目      [libuuid\(3LIB\)](#)、[libv12n\(3LIB\)](#)

参 照

## 拡張ライブラリ関数

|      |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
|------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 名前   | v12n, v12n_capabilities, v12n_domain_roles, v12n_domain_name, v12n_domain_uuid, v12n_ctrl_domain, v12n_chassis_serialno – 仮想化環境のドメインパラメータを返します。                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| 形式   | <pre>cc [ flag... ] file... -lv12n [ library... ] #include &lt;libv12n.h&gt;  int v12n_capabilities();  int v12n_domain_roles();  int v12n_domain_uuid(uuid_t uuid);  size_t v12n_domain_name(char *buf, size_t buflen);  size_t v12n_ctrl_domain(char *buf, size_t buflen);  size_t v12n_chassis_serialno(char *buf, size_t buflen);</pre>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| 機能説明 | <p>v12n_capabilities() 関数は、現在のドメインの仮想化機能マスクを返します。仮想化機能ビットマスクは次の値で構成されます。</p> <p>V12N_CAP_SUPPORTED      仮想化がこのドメインでサポートされています。</p> <p>V12N_CAP_ENABLED        仮想化がこのドメインで有効になっています。</p> <p>V12N_CAP_IMPL_LDOMS     Logical Domains はサポートされている仮想化実装です。</p> <p>v12n_domain_roles() 関数は仮想化ドメイン役割マスクを返します。仮想化ドメイン役割マスクは次の値で構成されます。</p> <p>V12N_ROLE_CONTROL      仮想化実装が Logical Domains であり、このビットが1の場合、現在のドメインが制御ドメインです。このビットが0の場合、現在のドメインがゲストドメインです。</p> <p>V12N_ROLE_IO            現在のドメインが I/O ドメインです。</p> <p>V12N_ROLE_SERVICE      現在のドメインがサービスドメインです。</p> <p>V12N_ROLE_ROOT          現在のドメインがルート I/O ドメインです。</p> <p>v12n_domain_uuid() 関数は、uuid 引数の現在の仮想化ドメインの汎用一意識別子 (Universally Unique Identifier, UUID) を格納します。<a href="#">libuuid(3LIB)</a> マニュアルページを参照してください。</p> <p>v12n_domain_name() 関数は、bufによって指定された場所にある現在の仮想化ドメインの名前を格納します。buflen は、バッファのサイズ(バイト単位)を指定します。完全な NULL で終わる名前を保持するにはバッファが小さすぎる場合は、名前の最初の buflen バイトがバッファに格納されます。V12N_NAME_MAX サイズのバッファは、ドメイン名を保持するには十分です。bufが NULL であるか、buflen が 0 である場合、名前はバッファにコピーされません。</p> |

`v12n_ctrl_domain()` 関数は、*buf*により指定された場所にある現在のドメインの制御ドメインまたは`dom0` ネットワークノード名を格納します。ドメインの制御ドメインは、ドメインの移行時には流動的になります。ドメインが移行の処理中だった場合、この関数により返される情報は失効している可能性があります。*buflen* は、バッファのサイズ(バイト単位)を指定します。完全な NULL で終わる名前を保持するにはバッファが小さすぎる場合は、名前の最初の *buflen* バイトがバッファに格納されます。`V12N_NAME_MAX` サイズのバッファは、制御ドメインノード名の文字列を保持するには十分です。*buf* が NULL であるか、*buflen* が 0 である場合、名前はバッファにコピーされません。

`v12n_chassis_serialno()` 関数は、現在のドメインが *buf* により指定された場所で実行されているプラットフォームのシャーシのシリアル番号を格納します。シャーシのシリアル番号は、ドメインの移行時には流動的になります。ドメインが移行の処理中だった場合、この関数により返される情報は失効している可能性があります。*buflen* は、バッファのサイズ(バイト単位)を指定します。完全な NULL で終わる名前を保持するにはバッファが小さすぎる場合は、名前の最初の *buflen* バイトがバッファに格納されます。`V12N_NAME_MAX` サイズのバッファは、シャーシのシリアル番号の文字列を保持するには十分です。*buf* が NULL であるか、*buflen* が 0 である場合、名前はバッファにコピーされません。

## 戻り値

正常に完了した場合、`v12n_capabilities()` および `v12n_domain_roles()` 関数は負でないビットマスクを返します。そうでない場合は、`v12n_domain_roles()` 関数が -1 を返し、エラーを示す `errno` を設定します。

正常に完了した場合、`v12n_domain_uuid()` 関数が 0 を返します。そうでない場合は、`v12n_domain_uuid()` 関数が -1 を返し、エラーを示す `errno` を設定します。

正常に完了した場合、`v12n_domain_name()`、`v12n_ctrl_domain()`、および `v12n_chassis_serialno()` 関数が完全な終端処理されていない文字列を保持するために必要なバッファサイズを返します。そうでない場合は、これらの関数が -1 を返し、エラーを示す `errno` を設定します。

## エラー

`v12n_domain_roles()` 関数は、呼び出し元のプロセスが特権ユーザー以外の ID を持つ場合には `EPERM` で失敗します。

次の場合に `v12n_domain_name()` 関数は失敗します。

- |                      |                                                                  |
|----------------------|------------------------------------------------------------------|
| <code>EPERM</code>   | 呼び出し元のプロセスが特権ユーザー以外の ID を持っています。                                 |
| <code>ENOTSUP</code> | 仮想化はこのドメインでサポートされていないか、有効になっていません。                               |
| <code>EFAULT</code>  | <i>buf</i> は不正なアドレスを指しています。                                      |
| <code>ENOENT</code>  | <code>sun4v</code> マシン記述にアクセスできないか、 <code>uuid</code> ノードがありません。 |

次の場合に `v12n_domain_uuid()` 関数は失敗します。

- EPERM 呼び出し元のプロセスが特権ユーザー以外の ID を持っています。
- ENOTSUP 仮想化はこのドメインでサポートされていないか、有効になっていません。
- EFAULT *buf*は不正なアドレスを指しています。
- ENOENT sun4v マシン記述にアクセスできないか、*uuid* ノードがありません。

次の場合に `v12n_ctrl_domain()` 関数は失敗します。

- EPERM 呼び出し元のプロセスが特権ユーザー以外の ID を持っています。
- ENOTSUP 仮想化はこのドメインでサポートされていないか、有効になっていません。
- EFAULT *buf*は不正なアドレスを指しています。
- ETIME 制御ドメイン上のドメインサービスがタイムアウト値以内に応答しませんでした。

次の場合に `v12n_chassis_serialno()` 関数は失敗します。

- EPERM 呼び出し元のプロセスが特権ユーザー以外の ID を持っています。
- ENOTSUP 仮想化はこのドメインでサポートされていないか、有効になっていません。
- EFAULT *buf*は不正なアドレスを指しています。
- ETIME 制御ドメイン上のドメインサービスがタイムアウト値以内に応答しませんでした。

属性 次の属性の説明については、[attributes\(5\)](#) を参照してください。

| 属性型         | 属性値   |
|-------------|-------|
| インタフェースの安定性 | 「確実」  |
| MT-レベル      | MT-安全 |

関連項目 `virtinfo(1M)`、[libuuid\(3LIB\)](#)、`libv12n(3LIB)`、[attributes\(5\)](#)



参 照

## ライブラリのインタフェースと ヘッダー

|                   |                                                                                                                                                                                                                                                                                                                         |                   |                       |                  |                  |                   |                  |
|-------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------|-----------------------|------------------|------------------|-------------------|------------------|
| 名前                | libv12n – 仮想化ドメイン情報インタフェースライブラリ                                                                                                                                                                                                                                                                                         |                   |                       |                  |                  |                   |                  |
| 形式                | <pre>cc [ flag... ] file... -lv12n [ library... ] #include &lt;libv12n.h&gt;</pre>                                                                                                                                                                                                                                      |                   |                       |                  |                  |                   |                  |
| 機能説明              | <p>このライブラリ内の関数は特定の仮想化ドメイン情報を抽出します。Logical Domains の場合、この情報は次のいずれかからのものです。</p> <ul style="list-style-type: none"><li>■ ドメインのマシン記述</li><li>■ Logical Domains エージェントデーモン (ldmad) により提供される制御ドメインのドメインサービス</li></ul>                                                                                                         |                   |                       |                  |                  |                   |                  |
| インタフェース           | <p>libv12n.so.1 共有オブジェクトは、次に定義されている公開インタフェースを提供します。共有オブジェクトインタフェースについては、<a href="#">Intro(3)</a> を参照してください。</p> <table><tr><td>v12n_capabilities</td><td>v12n_chassis_serialno</td></tr><tr><td>v12n_ctrl_domain</td><td>v12n_domain_name</td></tr><tr><td>v12n_domain_roles</td><td>v12n_domain_uuid</td></tr></table> | v12n_capabilities | v12n_chassis_serialno | v12n_ctrl_domain | v12n_domain_name | v12n_domain_roles | v12n_domain_uuid |
| v12n_capabilities | v12n_chassis_serialno                                                                                                                                                                                                                                                                                                   |                   |                       |                  |                  |                   |                  |
| v12n_ctrl_domain  | v12n_domain_name                                                                                                                                                                                                                                                                                                        |                   |                       |                  |                  |                   |                  |
| v12n_domain_roles | v12n_domain_uuid                                                                                                                                                                                                                                                                                                        |                   |                       |                  |                  |                   |                  |
| ファイル              | <p>/usr/lib/libv12n.so.1      共有オブジェクト</p> <p>/usr/lib/64/libv12n.so.1    64 ビット共有オブジェクト</p>                                                                                                                                                                                                                            |                   |                       |                  |                  |                   |                  |
| 属性                | 次の属性の説明については、 <a href="#">attributes(5)</a> を参照してください。                                                                                                                                                                                                                                                                  |                   |                       |                  |                  |                   |                  |

| 属性型         | 属性値                                               |
|-------------|---------------------------------------------------|
| 使用条件        | system/library (32 ビット)、SUNWcs1x (64 ビット)、SUNWhea |
| インタフェースの安定性 | 「確実」                                              |
| MT-レベル      | MT-安全                                             |

関連項目      virtinfo(1M)、[Intro\(3\)](#)、v12n(3EXT)、[attributes\(5\)](#)