

**Oracle® VM Server for SPARC 3.4 リファ
レンスマニュアル**

ORACLE®

Part No: E71826
2016年9月

目次

I システム管理	5
ldm	7
ldmd	99
ldmp2v	103
ldmpower	111
ovmtadm	119
ovmtconfig	123
ovmtcreate	129
ovmtdeploy	133
ovmtlibrary	139
ovmtprop	143

システム管理

Name

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

```
ldm or ldm --help [subcommand]
ldm -v
ldm add-domain -i file
ldm add-domain [cpu-arch=generic|native|migration-class1|sparc64-class1] [hostid=num]
    [mac-addr=MAC-address] [failure-policy=ignore|panic|reset|stop]
    [extended-mapin-space=off] [boot-policy=enforce|none|warning]
    [master=master-ldom1, ..., master-ldom4] [max-cores=[num|unlimited]] [uuid=uuid]
    [shutdown-group=num] [rc-add-policy=[iovs]] [perf-counters=counter-set] domain-name
ldm add-domain domain-name...
ldm set-domain -i file
ldm set-domain [cpu-arch=generic|native|migration-class1|sparc64-class1] [hostid=num]
    [mac-addr=MAC-address] [failure-policy=ignore|panic|reset|stop]
    [extended-mapin-space=[on|off]] [boot-policy=enforce|none|warning]
    [master=[master-ldom1, ..., master-ldom4]] [max-cores=[num|unlimited]] [shutdown-group=num]
    [rc-add-policy=[iovs]] [perf-counters=[counter-set]] domain-name
ldm remove-domain -a
ldm remove-domain domain-name...
ldm list-domain [-e] [-l] [-o format] [-p] [-S] [domain-name...]
ldm migrate-domain [-f] [-n] [-p filename] source-ldom [user@]target-host[:target-ldom]
ldm migrate-domain [-f] [-n] -c source-ldom target-host[:target-ldom]
ldm add-vcpu CPU-count domain-name
ldm set-vcpu CPU-count domain-name
ldm remove-vcpu [-f] CPU-count domain-name
ldm add-core num domain-name
ldm add-core cid=core-ID[, core-ID[, ...]] domain-name
ldm set-core num domain-name
ldm set-core cid=[core-ID[, core-ID[, ...]]] domain-name
ldm remove-core [-f] num domain-name
ldm remove-core cid=core-ID[, core-ID[, ...]] domain-name
ldm remove-core -g resource-group [-n number-of-cores] domain-name
ldm add-crypto number domain-name
ldm set-crypto [-f] number domain-name
ldm remove-crypto [-f] number domain-name
ldm add-memory [--auto-adj] size[unit] domain-name
ldm add-memory mblock=PA-start:size[, PA-start:size[, ...]] domain-name
ldm set-memory [--auto-adj] size[unit] domain-name
ldm set-memory mblock=[PA-start:size[, PA-start:size[, ...]]] domain-name
ldm remove-memory [--auto-adj] size[unit] domain-name
ldm remove-memory mblock=PA-start:size[, PA-start:size[, ...]] domain-name
ldm remove-memory -g resource-group [-s size[unit]] domain-name
ldm start-reconf domain-name
ldm cancel-reconf domain-name
ldm cancel-operation (migration | reconf | memdr) domain-name
ldm add-io (device | vf-name) domain-name
ldm add-io bus domain-name
ldm add-io iov=on|off domain-name
ldm set-io name=value [name=value...] pf-name
ldm set-io iov=on|off bus
ldm set-io [mac-addr=MAC-address] [alt-mac-addr=[auto|MAC-address, [auto|MAC-address, ...]]]
    [pvid=[pvid]] [vid=[vid1, vid2, ...]] [mtu=size] [name=value...] net-vf-name
ldm set-io name=[value...] ib-pf-name
ldm set-io [bw-percent=[value]] [port-wwn=value node-wwn=value] fc-vf-name
ldm remove-io [-n] (bus | device | vf-name) domain-name
ldm list-io [-l] [-p] [bus | device | pf-name]
ldm list-io -d pf-name
ldm add-vsw [-q] [default-vlan-id=VLAN-ID] [pvid=port-VLAN-ID] [vid=VLAN-ID1,VLAN-ID2,...]
```

```

[linkprop=phys-state] [mac-addr=MAC-address] [net-dev=device] [mode=sc] [mtu=size]
[id=switch-ID] [inter-vnet-link=[auto|on|off]] vswitch-name domain-name
ldm set-vsw [-q] [pvid=port-VLAN-ID] [vid=[+|-]VLAN-ID1,VLAN-ID2,...]
[mac-addr=MAC-address] [net-dev=device] [linkprop=[phys-state]] [mode=[sc]] [mtu=size]
[inter-vnet-link=[auto|on|off]] vswitch-name
ldm remove-vsw [-f] vswitch-name
ldm add-vnet [mac-addr=MAC-address] [pvid=port-VLAN-ID] [pvlan=secondary-vid, pvlan-type]
[protection=protection-type[, protection-type], ...] [allowed-ips=IP-address[, IP-address]...]
[priority=high|medium|low] [cos=0-7]
[allowed-dhcp-cids=[MAC-address|hostname, MAC-address|hostname, ...]]
[alt-mac-addr=auto|MAC-address[, auto|MAC-address, ...]] [vid=VLAN-ID1,VLAN-ID2,...]
[linkprop=phys-state] [id=network-ID] [mtu=size] [auto-alt-mac-addr=num]
[custom=[enable|disable]] [custom/max-vlans=num] [custom/max-mac-addr=num]
[maxbw=value] if-name vswitch-name domain-name
ldm set-vnet [mac-addr=MAC-address] [vswitch=vswitch-name] [mode=] [pvid=port-VLAN-ID]
[pvlan=[secondary-vid, pvlan-type]] [protection=[+|-]protection-type[, protection-type], ...]
[allowed-ips=[+|-]IP-address[, IP-address]...] [priority=high|medium|low] [cos=0-7]
[allowed-dhcp-cids=[+|-]MAC-address|hostname, MAC-address|hostname, ...]]
[alt-mac-addr=[+|-]auto|MAC-address[, auto|MAC-address, ...]]]
[vid=[+|-]VLAN-ID1,VLAN-ID2,...] [linkprop=[phys-state]] [mtu=size]
[auto-alt-mac-addr=[+]num] [custom=[enable|disable]] [custom/max-vlans=[num]]
[custom/max-mac-addr=[num]] [maxbw=[value]] if-name domain-name
ldm remove-vnet [-f] if-name domain-name
ldm add-vds service-name domain-name
ldm remove-vds [-f] service-name
ldm add-vdsdev [-f] [-q] [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 domain-name
ldm set-vdisk [timeout=seconds] [volume=volume-name@service-name] disk-name domain-name
ldm remove-vdisk [-f] disk-name domain-name
ldm list-hba [-d] [-l] [-p] [-t] domain-name
ldm add-vhba [id=vHBA-ID] vHBA-name vSAN-name domain-name
ldm rescan-vhba vHBA-name domain-name
ldm remove-vhba vHBA-name domain-name
ldm set-vhba timeout=seconds vHBA-name domain-name
ldm add-vsan [-q] iport-path vSAN-name domain-name
ldm remove-vsant vSAN-name
ldm add-vcc port-range=x-y vcc-name domain-name
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] [log=[on|off]] domain-name
ldm create-vf -n number | max pf-name
ldm create-vf [mac-addr=MAC-address] [vid=vid1,vid2,...] [mtu=size]
[alt-mac-addr=[auto|MAC-address, [auto|MAC-address, ...]]] [pvid=pvid] [name=value...]
net-pf-name
ldm create-vf [name=value...] ib-pf-name
ldm create-vf [port-wwn=value node-wwn=value] [bw-percent=[value]] fc-pf-name
ldm destroy-vf vf-name
ldm destroy-vf -n number | max pf-name
ldm add-variable var-name=[value]... domain-name
ldm set-variable var-name=[value]... domain-name
ldm remove-variable var-name... domain-name
ldm list-variable [var-name...] domain-name
ldm start-domain (-a | -i file | domain-name...)
ldm stop-domain [[-f | -q] | [[-h | -r | -t sec] [-m msg]]] (-a | domain-name...)
ldm panic-domain domain-name
ldm bind-domain [-f] [-q] (-i file | domain-name)
ldm unbind-domain domain-name
ldm list-bindings [-e] [-o [network|net]] [-p] [domain-name...]
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]) [domain-name...]
ldm list-devices [-a] [-p] [-S] [cmi] [core] [cpu] [crypto] [io] [memory]
ldm list-hvdump
ldm list-permits
ldm list-services [-e] [-p] [domain-name...]
ldm set-hvdump [hvdump=on|off] [hvdump-reboot=on|off]
ldm start-hvdump
ldm add-policy [attack=value] [decay=value] [elastic-margin=value]
[enable=yes|no] [priority=value] [sample-rate=value] [tod-begin=hh:mm[:ss]]
[tod-end=hh:mm[:ss]] [util-lower=percent] [util-upper=percent]
[vcpu-max=value] [vcpu-min=value] name=policy-name domain-name...
ldm set-policy [attack=value] [decay=value] [elastic-margin=value]
[enable=yes|no] [priority=value] [sample-rate=value] [tod-begin=[hh:mm:ss]]
[tod-end=[hh:mm:ss]] [util-lower=[percent]] [util-upper=[percent]] [vcpu-max=[value]]
[vcpu-min=[value]] name=policy-name domain-name...
ldm remove-policy [name=]policy-name... domain-name
ldm init-system [-frs] -i file
ldm list-netdev [-b] [-l] [-o net-device] [-p] [domain-name...]
ldm list-netstat [-o net-device] [-p] [-t interval [-c count]] [-u unit] [domain-name]
ldm list-dependencies [-l] [-p] [-r] [domain-name]
ldm list-rsrc-group [-a] [-d domain-name] [-l] [-o core|memory|io] [-p] [resource-group]
ldm add-cmi num domain-name
ldm add-cmi cmi_id=ID[,ID[,...]] domain-name
ldm set-cmi [-f] num domain-name
ldm set-cmi [-f] cmi_id=[ID[,ID[,...]]] domain-name
ldm remove-cmi num domain-name
ldm remove-cmi cmi_id=ID[,ID[,...]] domain-name
ldm grow-cmi vcpus=num cmi_id=ID domain-name
ldm grow-cmi cores=num cmi_id=ID domain-name
ldm shrink-cmi vcpus=num cmi_id=ID domain-name
ldm shrink-cmi cores=num cmi_id=ID domain-name
ldm evict-cmi vcpus=num cmi_id=ID domain-name
ldm evict-cmi cores=num cmi_id=ID domain-name
ldm list-cmi [-l] [-p] [cmi_id=ID[,ID[,...]]] [domain-name...]
ldm grow-socket vcpus=num socket_id=ID domain-name
ldm grow-socket cores=num socket_id=ID domain-name
ldm grow-socket memory=size[unit] socket_id=ID domain-name
ldm shrink-socket vcpus=num socket_id=ID domain-name
ldm shrink-socket cores=num socket_id=ID domain-name
ldm shrink-socket memory=size[unit] socket_id=ID domain-name
ldm set-socket [-f] [--remap] socket_id=[ID[,ID[,...]]] domain-name
ldm set-socket [-f] [--remap] --restore-degraded domain-name
ldm list-socket [--free] [-l] [-o format] [-p] [socket_id=ID[,ID[,...]]] [domain-name...]

```

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。物理ドメインがあるプラットフォームの場合、Logical Domains Manager は、各物理ドメインの制御ドメインでのみ実行されます。制御ドメインの名前は **primary** です。

論理ドメインは、個別のオペレーティングシステム、リソース、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、破棄、再構成、およびリブート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 SR-IOV (Single-Root I/O Virtualization) 仮想機能や PCIe (PCI EXPRESS) コントローラ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します。I/O ドメインは PCIe ルートコンプレックスを所有するか、ダイレクト I/O 機能を使用して PCIe スロットまたはシステムボード上の PCIe デバイス、および SR-IOV 機能を使用して SR-IOV 仮想機能を所有することができます。
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。

ルートドメイン 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、仮想機能を作成して I/O ドメインに割り当てることのできる SR-IOV 物理機能をすべて所有します。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。
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によって決まります。詳細は、プラットフォームのドキュメントを参照してください。
デフォルトのルートドメインは、primary ドメインです。

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。スレーブドメインは、マスタードメインに障害が発生したときの障害ポリシーを制定します。たとえば、マスタードメインに障害が発生した場合、スレーブドメインに対して、放置、パニックの発生、リブート、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。1つのドメインには最大4つのマスタードメインを指定できます。1つ以上のマスタードメインに障害が発生すると、障害ポリシーによってスレーブドメインの動作が指示されます。

サブコマンドのサマリー

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 3.4 管理ガイド](#)の“[権利プロファイルと役割の使用](#)”を参照してください。

サブコマンド	説明	承認
<code>add-resource</code>	既存の論理ドメインにリソースを追加します。	<code>solaris.ldoms.write</code>
<code>add-domain</code>	論理ドメインを作成します。	<code>solaris.ldoms.write</code>
<code>add-policy</code>	既存の論理ドメインにリソース管理ポリシーを追加します。	<code>solaris.ldoms.write</code>
<code>add-spconfig</code>	論理ドメイン構成をサービスプロセッサ (Service Processor, SP) に追加します。	<code>solaris.ldoms.write</code>
<code>add-variable</code>	論理ドメインに1つ以上の変数を追加します。	<code>solaris.ldoms.write</code>
<code>bind-domain</code>	作成された論理ドメインにリソースをバインドします。	<code>solaris.ldoms.write</code>
<code>cancel-operation</code>	遅延再構成 (<code>reconf</code>)、メモリー動的再構成の削除 (<code>memdr</code>)、またはドメイン移行 (<code>migration</code>) などの処理を取り消します。	<code>solaris.ldoms.write</code>
<code>cancel-reconf</code>	<code>primary</code> ドメインの遅延再構成処理を取り消します。	<code>solaris.ldoms.write</code>
<code>create-vf</code>	1つ以上の仮想機能を作成します。	<code>solaris.ldoms.write</code>
<code>destroy-vf</code>	1つ以上の仮想機能を破棄します。	<code>solaris.ldoms.write</code>
<code>evict-cmi</code>	特定の CMI デバイスに関連付けられた仮想 CPU または仮想 CPU コアを、そのデバイスを所有する論理ドメインから削除します。	<code>solaris.ldoms.write</code>
<code>grow-cmi</code>	特定の CMI デバイスに関連付けられた仮想 CPU または仮想 CPU コアを、そのデバイスを所有する論理ドメインに追加します。	<code>solaris.ldoms.write</code>
<code>grow-socket</code>	特定の CPU ソケットに関連付けられた仮想 CPU、仮想 CPU コア、または仮想メモリーを既存の論理ドメインに追加します。	<code>solaris.ldoms.write</code>
<code>init-system</code>	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、あるいはその両方を構成します。	<code>solaris.ldoms.write</code>
<code>list-bindings</code>	論理ドメインのサーバーバインドを一覧表示します。	<code>solaris.ldoms.read</code>

サブコマンド	説明	承認
list-cmi	論理ドメインのデバイスを一覧表示します。	solaris.ldoms.read
list-constraints	論理ドメインのリソース制約を一覧表示します。	solaris.ldoms.read
list-dependencies	依存関係を一覧表示します。	solaris.ldoms.read
list-devices	論理ドメインのデバイスを一覧表示します。	solaris.ldoms.read
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-hba	論理ドメインの SCSI ホストバスアダプタ (HBA) を一覧表示します。	solaris.ldoms.read
list-hvdump	ハイパーバイザデータ収集のプロパティ値を一覧表示します。	solaris.ldoms.read
list-io	論理ドメインの I/O デバイスを一覧表示します。	solaris.ldoms.read
list-netdev	論理ドメインのネットワークデバイスを一覧表示します。	solaris.ldoms.read
list-netstat	論理ドメインのネットワークデバイス統計情報を一覧表示します。	solaris.ldoms.read
list-permits	CPU コアアクティベーションの情報を一覧表示します。	solaris.ldoms.read
list-rsrc-group	リソースグループ情報を一覧表示します。	solaris.ldoms.read
list-services	論理ドメインのサービスを一覧表示します。	solaris.ldoms.read
list-socket	CPU ソケット情報を一覧表示します。	solaris.ldoms.read
list-spconfig	論理ドメインの構成を一覧表示します。	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
remove-resource	既存の論理ドメインからリソースを削除します。	solaris.ldoms.write
remove-domain	論理ドメインを削除します。	solaris.ldoms.write
remove-policy	既存の論理ドメインからリソース管理ポリシーを削除します。	solaris.ldoms.write
remove-spconfig	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。	solaris.ldoms.write
remove-variable	既存の論理ドメインから 1 つ以上の変数を削除します。	solaris.ldoms.write
rescan-vhba	仮想 SCSI HBA および仮想 SAN によって認識される SCSI デバイスのセットを同期します。	solaris.ldoms.read
set-resource	既存の論理ドメインに対してリソースを指定します。これは、プロパティの変更	solaris.ldoms.write

サブコマンド	説明	承認
	または数量の変更のいずれかに使用できません。リソース <code>cmi</code> 、 <code>core</code> 、 <code>vcpu</code> 、 <code>memory</code> 、または <code>crypto</code> に適用される場合、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、指定したリソースの数量が、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リソースの数がこのサブコマンドの指定よりも多い場合は、いくつかのリソースが削除されます。論理ドメインに割り当てられているリソースの数がこのサブコマンドの指定よりも少ない場合は、いくつかのリソースが追加されます。リソースの定義については、「リソース」を参照してください。	
<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-hvdump</code>	ハイパーバイザデータ収集プロセスのプロパティ値を設定します。	<code>solaris.ldoms.write</code>
<code>set-io</code>	物理機能または仮想機能を変更します。	<code>solaris.ldoms.write</code>
<code>set-policy</code>	既存の論理ドメインにリソース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-socket</code>	指定された CPU ソケットに関連付けられた仮想 CPU、仮想 CPU コア、および仮想メモリーリソースを使用するように既存の論理ドメインを制約します。	<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-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>shrink-cmi</code>	特定の CMI デバイスに関連付けられた仮想 CPU または仮想 CPU コアを、そのデバイスを所有する論理ドメインから削除します。	<code>solaris.ldoms.write</code>
<code>shrink-socket</code>	特定の CPU ソケットに関連付けられた仮想 CPU、仮想 CPU コア、または仮想メモリーを既存の論理ドメインから削除します。	<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-hvdump</code>	ハイパーバイザデータ収集プロセスを手動で起動します。	<code>solaris.ldoms.write</code>
<code>start-reconf</code>	ルートドメイン上で遅延再構成モードに入ります。	<code>solaris.ldoms.write</code>
<code>stop-domain</code>	1 つ以上の実行中のドメインを停止します。	<code>solaris.ldoms.write</code>
<code>unbind-domain</code>	論理ドメインから、リソースのバインドを解除 (リソースを解放) します。	<code>solaris.ldoms.write</code>

Note - すべてのサブコマンドが、すべてのリソースタイプでサポートされているわけではありません。

別名

このセクションには、**ldm** サブコマンドのアクション (動詞)、リソース名 (名詞)、および完全なサブコマンドの短い形式と長い形式を示す表が含まれています。

次の表は、サブコマンドのアクションの短い形式と長い形式を示しています。

省略形式	長文形式
ls	list
rm	remove

次の表は、リソース名の短い形式と長い形式を示しています。

省略形式	長文形式
config	spconfig
crypto	mau
dep	dependencies
dom	domain
group	rsrc-group
mem	memory
var	variable
vcc	vconscon
vcons	vconsole
vds	vdiskserver
vdsdev	vdiskserverdevice
vsw	vswitch

次の表は、サブコマンドの短い形式と長い形式を示しています。

省略形式	長文形式
bind	bind-domain
cancel-op	cancel-operation
create	add-domain
destroy	remove-domain

省略形式	長文形式
list	list-domain
migrate	migrate-domain
modify	set-domain
panic	panic-domain
start	start-domain
stop	stop-domain
unbind	unbind-domain

Note - 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リソースの別名を使用します。

リソース

次のリソースがサポートされています。

core	CPU コア。
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 (Modular Arithmetic Unit, 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。
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。または、ダイレクト I/O 割り当て可能デバイスおよび PCIe SR-IOV 仮想機能。
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。プラットフォームのドキュメントを参照してください。

<code>vdisk</code>	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 (<code>tN</code>) は除外されます。論理ドメインの仮想ディスクの形式は、 <code>cNdNsN</code> です。 <code>cN</code> は仮想コントローラ、 <code>dN</code> は仮想ディスク番号、および <code>sN</code> はスライスを示します。
<code>vds</code> 、 <code>vdiskserver</code>	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。
<code>vdsdev</code> 、 <code>vdiskserverdev</code>	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。
<code>vhba</code>	Sun Common SCSI Architecture (SCSA) インタフェースをサポートする仮想 SCSI ホストバスアダプタ (HBA)。
<code>vnet</code>	仮想 Ethernet デバイスを実装し、仮想ネットワークスイッチ (<code>vsw</code>) を使用するシステム内のほかの <code>vnet</code> デバイスと通信する仮想ネットワークデバイス。
<code>vsan</code>	指定された SCSI HBA イニシエータポートの配下にある物理 SCSI デバイスのセットをエクスポートする仮想ストレージエリアネットワーク (SAN) サービス。
<code>vsw</code> 、 <code>vswitch</code>	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。

サブコマンドの使用法

このセクションでは、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リソース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

`add-domain` 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、マスタートドメインのリスト、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。

構文:

```
ldm add-domain -i file
```

```
ldm add-domain [cpu-arch=generic|native|migration-class1|sparc64-class1] [hostid=num]
[mac-addr=MAC-address] [failure-policy=ignore|panic|reset|stop] [extended-mapin-space=off]
[boot-policy=enforce|none|warning] [master=master-ldom1, ..., master-ldom4]
[max-cores=[num|unlimited]] [uuid=uuid] [shutdown-group=num] [rc-add-policy=[iovm]]
[perf-counters=counter-set] domain-name
```

```
ldm add-domain domain-name...
```

ここでは:

- `-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。
- `cpu-arch=generic|native|migration-class1|sparc64-class1` には、次のいずれかの値を指定します。
 - `generic` は、CPU タイプに依存しない移行を実行できるようにゲストドメインを構成します。
 - `native` は、同じ CPU タイプを持つプラットフォーム間でのみ移行できるようにゲストドメインを構成します。`native` はデフォルト値です。
 - `migration-class1` は、SPARC T4 以降の SPARC プラットフォーム用の CPU 間移行ファミリです。これらのプラットフォームでは、移行中と移行後のハードウェア暗号化をサポートするため、サポートされる CPU には下限があります。
この値は、UltraSPARC T2、UltraSPARC T2 Plus、または SPARC T3 プラットフォームや Fujitsu M10 サーバーとは互換性がありません。
 - `sparc64-class1` は、SPARC64 プラットフォーム用の CPU 間移行ファミリです。`sparc64-class1` 値は SPARC64 の命令に基づいているため、`generic` 値より多い命令数が含まれています。そのため、`sparc64-class1` 値は、`generic` 値に比べてパフォーマンスに影響しません。
この値は、UltraSPARC T2、UltraSPARC T2 Plus、SPARC T3、SPARC T4、SPARC T5、SPARC M5、または SPARC M6 プラットフォームとは互換性ありません。
- `boot-policy=enforce|none|warning` は検証済みブートポリシーを指定します。値が `enforce` の場合、ブートブロックとカーネルモジュールが検証されます。不正に署名されたブートブロックとモジュールはロードされず、ゲストドメインがブートしない可能性があります。ただし、値が `none` の場合は、検証が実行されず、ゲストドメインがブートします。デフォルト値は `warning` で、不正に署名されたブートブロックとカーネルモジュールに関する警告メッセージを発行しますが、モジュールのロードとゲストドメインのブートを続行します。
- `mac-addr=MAC-address` 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` には、マスタートメインに障害が発生したときのスレーブドメインの動作を制御する障害ポリシーを指定します。このプロパティーは、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、有効なプロパティー値を示します。
 - `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 (`ldm panic` コマンドを実行することと似ています)。
 - `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停止して再起動します (`ldm stop -f` コマンド、`ldm start` コマンドの順に実行することと似ています)。
 - `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ただちに停止します (`ldm stop -f` コマンドを実行することと似ています)。
 - `extended-mapin-space=off` は、指定されたドメインで拡張されたマップイン領域を無効にします。デフォルトでは、拡張されたマップイン領域は有効になっています。
 - `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。このプロパティーは、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。

Note - 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。

- `rc-add-policy` は、特定のドメインに追加される可能性のあるルートコンプレックスで、ダイレクト I/O および SR-IOV I/O 仮想化操作を有効または無効にするかどうかを指定します。有効な値は、`iov` および値なし (`rc-add-policy=`) です。`rc-add-policy=iov` の場合、追加されるルートコンプレックスで、ダイレクト I/O および SR-IOV の機能が有効になります。`rc-add-policy=` の場合、`iov` プロパティー値がクリアされ、ルートコンプレックスの I/O 仮想化機能が無効になります (`add-io` コマンドを使用して `iov=on` を明示的に設定していない場合)。デフォルト値は値なしです。
- `perf-counters=counter-set` は、パフォーマンスカウンタに許可するアクセスのタイプを指定します。`perf-counters` 値が指定されていない場合、この値は `htstrand` になります。`perf-counters` プロパティーには次の値を指定できます。

`global` ドメインに対し、そのドメインに割り当てられたリソースがアクセスできるグローバルパフォーマンスカウンタへのアクセス

を許可します。グローバルパフォーマンスカウンタに同時にアクセスできるドメインは1つだけです。この値は単独で指定することも、strand または htstrand 値と一緒に指定することもできます。

strand	ドメインに対し、そのドメインに割り当てられた CPU 上に存在するストランドパフォーマンスカウンタへのアクセスを許可します。この値と htstrand 値と一緒に指定することはできません。
htstrand	strand 値と同様に動作し、さらにそのドメインに割り当てられた CPU 上のハイパー権限モードイベントの計測を可能にします。この値と strand 値と一緒に指定することはできません。

すべてのパフォーマンスカウンタへのすべてのアクセスを無効にするには、perf-counters= を指定します。

- `uuid=uuid` には、ドメインの汎用一意識別子 (UUID) を指定します。`uuid` は、ダッシュで区切った5つの16進数で構成される16進数の文字列 (12345678-1234-abcd-1234-123456789abc など) です。各数字には、次のように、指定された桁数の16進数 (8、4、4、4、12) が必要です。

xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx

- `max-cores=[num|unlimited]` には、ドメインに割り当てることができるコアの最大数を指定します。値が `unlimited` の場合、割り当てることができる CPU コアの数に制限はありません。
- `shutdown-group=num` には、ドメインの停止グループ番号を指定します。この値は、順序付きシャットダウンが実行されるときに、Fujitsu M10 サーバー上の SP で使用されます。

SP によって、順序付きシャットダウンが開始されると、停止グループ番号の降順でドメインが停止されます。つまり、もっとも大きい番号を持つドメインが最初に停止され、もっとも小さい番号を持つドメインが最後に停止されます。複数のドメインで1つの停止グループ番号を共有している場合、それらのドメインは同時に停止します。マスタードメインとスレーブドメインで1つの停止グループ番号を共有している場合、それらのドメインはマスター-スレーブ関係が存在していても、同時に停止します。したがって、マスタードメインとスレーブドメインの間に依存関係を確立するときは、各ドメインに異なる停止グループ番号を割り当てます。

有効な値は、1 から 15 までです。制御ドメインの停止グループ番号はゼロ (0) で、変更できません。その他のドメインのデフォルト値は 15 です。

新しい `shutdown-group` プロパティ値を有効にするには、`ldm add-spconfig` コマンドを使用して構成を SP に保存する必要があります。

このプロパティは、Fujitsu M10 プラットフォームにのみ適用されます。

- `domain-name` は、追加され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

set-domain サブコマンドでは、各ドメインの boot-policy、mac-addr、hostid、failure-policy、extended-mapin-space、master、および max-cores プロパティーのみを変更できます。このコマンドをリソースのプロパティーを更新するために使用することはできません。

Note - 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ドメインもバインドしておく必要があります。

構文:

```
ldm set-domain -i file
```

```
ldm set-domain [cpu-arch=generic|native|migration-class1|sparc64-class1] [hostid=num]
[mac-addr=MAC-address] [failure-policy=ignore|panic|reset|stop]
[extended-mapin-space=[on|off]] [boot-policy=enforce|none|warning]
[master=[master-ldom1,...,master-ldom4]] [max-cores=[num|unlimited]] [shutdown-group=num]
[rc-add-policy=[ioV]] [perf-counters=[counter-set]] domain-name
```

ここでは:

- -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。
XML ファイルで指定された ldom_info ノードのみが解析されます。vcpu、mau、memory などのリソースノードは無視されます。
- cpu-arch=generic|native|migration-class1|sparc64-class1 には、次のいずれかの値を指定します。
 - generic は、CPU タイプに依存しない移行を実行できるようにゲストドメインを構成します。
 - native は、同じ CPU タイプを持つプラットフォーム間でのみ移行できるようにゲストドメインを構成します。native はデフォルト値です。
 - migration-class1 は、SPARC T4 以降の SPARC プラットフォーム用の CPU 間移行ファミリです。これらのプラットフォームでは、移行中と移行後のハードウェア暗号化をサポートするため、サポートされる CPU には下限がありません。
この値は、UltraSPARC T2、UltraSPARC T2 Plus、または SPARC T3 プラットフォームや Fujitsu M10 サーバーとは互換性がありません。
 - sparc64-class1 は、SPARC64 プラットフォーム用の CPU 間移行ファミリです。sparc64-class1 値は SPARC64 の命令に基づいているため、generic 値より多い命令数が含まれています。そのため、sparc64-class1 値は、generic 値に比べてパフォーマンスに影響しません。
この値は、UltraSPARC T2、UltraSPARC T2 Plus、SPARC T3、SPARC T4、SPARC T5、SPARC M5、または SPARC M6 プラットフォームとは互換性がありません。

-
- `boot-policy=enforce|none|warning` は検証済みブートポリシーを指定します。値が `enforce` の場合、ブートブロックとカーネルモジュールが検証されます。不正に署名されたブートブロックとモジュールはロードされず、ゲストドメインがブートしない可能性があります。ただし、値が `none` の場合は、検証が実行されず、ゲストドメインがブートします。デフォルト値は `warning` で、不正に署名されたブートブロックとカーネルモジュールに関する警告メッセージを発行しますが、モジュールのロードとゲストドメインのブートを続行します。
`boot-policy` 値を変更するときにドメインがアクティブな場合は、ドメインをリブートして変更を有効にする必要があります。
 - `mac-addr=MAC-address` 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` には、マスタードメインに障害が発生したときのスレーブドメインの動作を制御する障害ポリシーを指定します。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 `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。
 - `panic` 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。
 - `reset` 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て再起動します。
 - `stop` 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。
 - `extended-mapin-space` は、指定したドメインで拡張されたマップイン領域を有効または無効にすることができます。デフォルトでは、`extended-mapin-space=on` であり、これは `extended-mapin-space=` と同等の設定です。
 - `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。

Note - 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。

- `rc-add-policy` は、特定のドメインに追加される可能性のあるルートコンプレックスで、ダイレクト I/O および SR-IOV I/O 仮想化操作を有効または無効にするかどうかを指定します。有効な値は、`iov` および値なし (`rc-add-policy=`) です。`rc-add-policy=iov` の場合、追加されるルートコンプレックスで、ダイレク

ト I/O および SR-IOV の機能が有効になります。rc-add-policy= の場合、iov プロパティ値がクリアされ、ルートコンプレックスの I/O 仮想化機能が無効になります (add-io コマンドを使用して iov=on を明示的に設定していない場合)。デフォルト値は値なしです。

- perf-counters=counter-set は、パフォーマンスカウンタに許可するアクセスのタイプを指定します。perf-counters プロパティには次の値を指定できます。

global	ドメインに対し、そのドメインに割り当てられたリソースがアクセスできるグローバルパフォーマンスカウンタへのアクセスを許可します。グローバルパフォーマンスカウンタに同時にアクセスできるドメインは1つだけです。この値は単独で指定することも、strand または htstrand 値と一緒に指定することもできます。
strand	ドメインに対し、そのドメインに割り当てられた CPU 上に存在するストランドパフォーマンスカウンタへのアクセスを許可します。この値と htstrand 値と一緒に指定することはできません。
htstrand	strand 値と同様に動作し、さらにそのドメインに割り当てられた CPU 上のハイパー権限モードイベントの計測を可能にします。この値と strand 値と一緒に指定することはできません。

すべてのパフォーマンスカウンタへのすべてのアクセスを無効にするには、perf-counters= を指定します。

- max-cores=[num|unlimited] には、ドメインに割り当てることができるコアの最大数を指定します。値が unlimited の場合、割り当てることができる CPU コアの数に制限はありません。
- shutdown-group=num には、ドメインの停止グループ番号を指定します。この値は、順序付きシャットダウンが実行される時に、Fujitsu M10 サーバー上の SP で使用されます。

SP によって、順序付きシャットダウンが開始されると、停止グループ番号の降順でドメインが停止されます。つまり、もっとも大きい番号を持つドメインが最初に停止され、もっとも小さい番号を持つドメインが最後に停止されます。複数のドメインで1つの停止グループ番号を共有している場合、それらのドメインは同時に停止します。マスタードメインとスレーブドメインで1つの停止グループ番号を共有している場合、それらのドメインはマスター-スレーブ関係が存在していても、同時に停止します。したがって、マスタードメインとスレーブドメインの間に依存関係を確立するときは、各ドメインに異なる停止グループ番号を割り当てます。

有効な値は、1 から 15 までです。制御ドメインの停止グループ番号はゼロ (0) で、変更できません。その他のドメインのデフォルト値は 15 です。

新しい shutdown-group プロパティ値を有効にするには、ldm add-spconfig コマンドを使用して構成を SP に保存する必要があります。

このプロパティは、Fujitsu M10 プラットフォームにのみ適用されます。

- `domain-name` 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

`remove-domain` 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 remove-domain -a
ldm remove-domain domain-name...
```

ここでは:

- `-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。
- `domain-name` は、削除され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

`migrate-domain` 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]
ldm migrate-domain [-f] [-n] -c source-ldom target-host[:target-ldom]
```

ここでは:

- `-f` は、ドメインの移行を強制的に行おうとします。
- `-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、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ない非対話型のマイグレーションを実行できます。

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。

このオプションは `-c` オプションと一緒に使用できません。

- `-c` は、信頼できる SSL 証明書を使用してドメイン移行を実行します。このオプションは `-p filename` オプションと一緒に使用できません。`-c` オプションを使用する場合は、ユーザー名を指定できません。

このオプションを使用するには、まず証明書がソースマシンとターゲットマシンにインストールされ、構成されていることを確認する必要があります。`-c` オプションが指定されている場合、ソースマシンはパスワードの入力を求めません。ターゲットの証明書を確認できない場合は、移行のリクエストが拒否されます。

信頼できる SSL 証明書が正常にアクセスされると、それらの証明書は `ldmd` インスタンスの有効期間中キャッシュされます。証明書を変更または削除した場合は、その変更を有効にするために `ldmd` デーモンを再起動する必要があります。

- `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。

- **動的再構成処理。** 動的再構成は、アクティブなドメインに対してリソースを追加、設定、または削除する機能です。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よって依存します。制御ドメインで動的再構成を実行できない場合、遅延再構成処理を開始します。自動的に遅延再構成が開始されることもあります。
- **遅延再構成処理。** すぐに有効になる動的再構成処理とは異なり、遅延再構成処理が有効になるのは、次に OS をリブートしたあとや、OS が実行中でない場合に論理ドメインを停止して起動したあとです。 `ldm start-reconf primary` コマンドを実行すると、ルート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ができます。 `primary` 以外のルートドメインで遅延再構成を開始した場合は、限られた一連の I/O 操作 (`add-io`、`set-io`、`remove-io`、`create-vf`、および `destroy-vf`) のみを実行できます。動的に構成できないリソースを変更する前に、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の詳細については、[Oracle VM Server for SPARC 3.4 管理ガイドの“リソースの再構成”](#)を参照してください。

CPU の操作

CPU スレッドまたは CPU コアをドメインに割り当てることができます。CPU スレッドを割り当てるには、`add-vcpu`、`set-vcpu`、および `remove-vcpu` サブコマンドを使用します。CPU コアを割り当てるには、`add-core`、`set-core`、および `remove-core` サブコマンドを使用します。

CPU スレッドの追加

`add-vcpu` サブコマンドは、指定された数の CPU スレッドまたは CPU コアを論理ドメインに追加します。ドメインでは CPU コア全体と CPU スレッドを同時に構成することはできないことに注意してください。CPU コアの構成と CPU スレッドの構成は相互排他的です。

構文:

```
ldm add-vcpu CPU-count domain-name
```

ここでは:

- *CPU-count* は、論理ドメインに追加する CPU スレッドの数です。
- *domain-name* は、CPU スレッドが追加される論理ドメインを指定します。

CPU スレッドの設定

`set-vcpu` サブコマンドは、論理ドメインで設定される CPU スレッドまたは CPU コア
の数を指定します。ドメインでは CPU コア全体と CPU スレッドを同時に構成するこ
とはできないことに注意してください。CPU コアの構成と CPU スレッドの構成は相
互排他的です。

構文:

```
ldm set-vcpu CPU-count domain-name
```

ここでは:

- *CPU-count* は、論理ドメインに追加する CPU スレッドの数です。
- *domain-name* は、CPU スレッドの数が設定される論理ドメインです。

CPU スレッドの削除

`remove-vcpu` サブコマンドは、指定された数の CPU スレッドまたは CPU コアを論理
ドメインから削除します。ドメインでは CPU コア全体と CPU スレッドを同時に構成
することはできないことに注意してください。CPU コアの構成と CPU スレッドの構
成は相互排他的です。

構文:

```
ldm remove-vcpu [-f] CPU-count domain-name
```

ここでは:

- `-f` は、アクティブなドメインから 1 つ以上の仮想 CPU スレッドを強制的に削除し
ようと試みます。
- *CPU-count* は、論理ドメインに追加する CPU スレッドの数です。
- *domain-name* は、CPU スレッドが削除される論理ドメインを指定します。

CPU コアの追加

`add-core` サブコマンドは、指定された数の CPU コアをドメインに追加します。CPU
コアの数を指定すると、割り当てられるコアが自動的に選択されます。ただし、`cid`
プロパティに *core-ID* 値を指定すると、指定されたコアが明示的に割り当てられま
す。

`cid` プロパティは、構成するシステムのトポロジについて知識のある管理者のみが
使用するようにしてください。この高度な構成機能は、特定の割り当て規則を強制的

に適用します。これは、システムのパフォーマンス全体に影響を与える可能性があります。

`ldm add-core` コマンドを使用して、自動的に割り当てられた (匿名の) コアリソースをすでに使用しているドメインに名前付きのコアリソースを追加することはできないことに注意してください。

構文:

```
ldm add-core num domain-name
```

```
ldm add-core cid=core-ID[,core-ID[,...]] domain-name
```

ここでは:

- `num` には、ドメインに割り当てる CPU コアの数指定します。
- `cid=core-ID[,...]` には、ドメインに割り当てる物理 CPU コアを 1 つ以上指定します。
- `domain-name` は、CPU コアが割り当てられるドメインを指定します。

CPU コアの設定

`set-core` サブコマンドは、ドメインに割り当てる CPU コアの数指定します。CPU コアの数指定すると、割り当てられるコアが自動的に選択されます。ただし、`cid` プロパティに `core-ID` 値を指定すると、指定されたコアが明示的に割り当てられません。

構文:

```
ldm set-core num domain-name
```

```
ldm set-core cid=[core-ID[,core-ID[,...]]] domain-name
```

ここでは:

- `num` には、ドメインに割り当てる CPU コアの数指定します。
- `cid=core-ID[,...]` には、ドメインに割り当てる物理 CPU コアを 1 つ以上指定します。cid= は、名前付き CPU コアをすべて削除します。
- `domain-name` は、CPU コアが割り当てられるドメインを指定します。

CPU コアの削除

`remove-core` サブコマンドは、ドメインから削除する CPU コアの数指定します。CPU コアの数指定すると、削除されるコアが自動的に選択されます。ただし、`cid` プロパティに `core-ID` 値を指定すると、指定されたコアが明示的に削除されます。

-g オプションを使用してリソースグループを指定した場合、削除対象として選択されるコアはすべて、そのリソースグループのものになります。

構文:

```
ldm remove-core [-f] num domain-name
ldm remove-core cid=[core-ID[, core-ID[, ...]]] domain-name
ldm remove-core -g resource-group [-n number-of-cores] domain-name
```

ここでは:

- `-f` は、アクティブなドメインから 1 つ以上のコアを強制的に削除しようと試みません。
- `-g resource-group` は、指定されたりソースグループ内のリソースに対して操作が行われるように指定します。
- `-n number-of-cores` は、削除するコアの数を指定します。このオプションが指定されていない場合は、指定されたドメインに属する指定されたりソースグループからすべてのコアが削除されます。このオプションは、`-g` オプションが指定されている場合にのみ使用できます。
- `num` には、ドメインから削除する CPU コアの数を指定します。
- `cid=core-ID[, ...]` には、ドメインから削除する物理 CPU コアを 1 つ以上指定します。
- `domain-name` は、CPU コアが削除されるドメインを指定します。

暗号化装置の操作

暗号化装置のサブコマンドは、個別の暗号化装置を持つ SPARC プラットフォームにのみ適用されます。SPARC T4 プラットフォーム以降の新しいプラットフォームには、統合された暗号化手順があるため、個別の暗号化装置は使用されません。

暗号化装置の追加

`add-crypto` サブコマンドは、論理ドメインに追加され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。

構文:

```
ldm add-crypto number domain-name
```

ここでは:

- `number` は、論理ドメインに追加する暗号化装置の数です。
- `domain-name` は、暗号化装置が追加される論理ドメインを指定します。

暗号化装置の設定

`set-crypto` サブコマンドは、論理ドメインで設定される暗号化装置の数を指定します。アクティブなドメインからすべての暗号化装置を削除する場合は、`-f` オプションを指定する必要があります。

ドメインがアクティブなときに **primary** ドメインから最後の暗号化装置を削除する場合は、次のいずれかの操作を実行します。

- 動的再構成を使用して、**-f** オプションを指定します
- 遅延再構成を使用します

構文:

```
ldm set-crypto [-f] number domain-name
```

ここでは:

- **number** が 0 の場合に **-f** を指定すると、ドメイン内の最後の暗号化装置が強制的に削除されます。
-f オプションは次の状況でのみ必要となります。
 - ゲストドメインがアクティブな場合
 - **primary** ドメイン上 (ただし、システム上にアクティブなゲストドメインが1つ以上存在する場合のみ)
- **number** は、論理ドメインで設定する暗号化装置の数です。
- **domain-name** は、暗号化装置の数が設定される論理ドメインを指定します。

暗号化装置の削除

remove-crypto サブコマンドは、指定された数の暗号化装置を論理ドメインから削除します。アクティブなドメインからすべての暗号化装置を削除する場合は、**-f** オプションを指定する必要があります。

ドメインがアクティブなときに **primary** ドメインから最後の暗号化装置を削除する場合は、次のいずれかの操作を実行します。

- 動的再構成を使用して、**-f** オプションを指定します
- 遅延再構成を使用します

構文:

```
ldm remove-crypto [-f] number domain-name
```

ここでは:

- **-f** は、**number** がドメイン内の暗号化装置の数と等しい場合に、ドメイン内の最後の暗号化装置を強制的に削除します。
-f オプションは次の状況でのみ必要となります。
 - ゲストドメインがアクティブな場合
 - **primary** ドメイン上 (ただし、システム上にアクティブなゲストドメインが1つ以上存在する場合のみ)
- **number** は、論理ドメインから削除する暗号化装置の数です。
- **domain-name** は、暗号化装置が削除され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

`add-memory` サブコマンドは、指定された量のメモリーをドメインに追加します。メモリーブロックのサイズを指定すると、割り当てられるメモリーブロックが自動的に選択されます。ただし、`mblock` プロパティに `PA-start:size` 値を指定すると、指定されたメモリーブロックが明示的に割り当てられます。

`mblock` プロパティは、構成するシステムのトポロジについて知識のある管理者のみが使用するようになっています。この高度な構成機能は、特定の割り当て規則を強制的に適用します。これは、システムのパフォーマンス全体に影響を与える可能性があります。

構文:

```
ldm add-memory [--auto-adj] size[unit] domain-name
```

```
ldm add-memory mblock=PA-start:size[, PA-start:size[,...]] domain-name
```

ここでは:

- `--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、論理ドメインで設定されるメモリーのサイズ (バイト単位) です。異なるサイズの単位が必要な場合は、`unit` を、大文字または小文字を使用して次のいずれかの値として指定します。
 - G バイトの場合は G
 - K バイトの場合は K
 - M バイトの場合は M
- `mblock=PA-start:size` には、ドメインに割り当てる物理メモリーブロックを 1 つ以上指定します。`PA-start` には、メモリーブロックの開始物理アドレスを 16 進形式で指定します。`size` は、ドメインに割り当てられるメモリーブロックのサイズ (単位を含む) です。このプロパティを使用して DIMM の物理アドレスを指定することはできないことに注意してください。
- `domain-name` は、メモリーが追加される論理ドメインを指定します。

メモリーの設定

`set-memory` サブコマンドは、ドメイン内に特定の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。

メモリーブロックのサイズを指定すると、割り当てられるメモリーブロックが自動的に選択されます。ただし、`mblock` プロパティに `PA-start:size` 値を指定すると、指定されたメモリーブロックが明示的に割り当てられます。

構文:

```
ldm set-memory [--auto-adj] size[unit] domain-name
```

```
ldm set-memory mblock=PA-start:size[,PA-start:size[,...]] domain-name
```

ここでは:

- `--auto-adj` は、アクティブドメインで追加または削除されるメモリーの量を自動的に 256M バイトに揃えることを指定し、これにより、リクエスト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、論理ドメインで設定されるメモリーのサイズ (バイト単位) です。異なるサイズの単位が必要な場合は、`unit` を、大文字または小文字を使用して次のいずれかの値として指定します。
 - G バイトの場合は G
 - K バイトの場合は K
 - M バイトの場合は M
- `mblock=PA-start:size` には、ドメインに割り当てる物理メモリーブロックを 1 つ以上指定します。`PA-start` には、メモリーブロックの開始物理アドレスを 16 進形式で指定します。`size` は、ドメインに割り当てられるメモリーブロックのサイズ (単位を含む) です。このプロパティを使用して DIMM の物理アドレスを指定することはできないことに注意してください。
- `domain-name` は、メモリーが変更される論理ドメインを指定します。

メモリーの削除

`remove-memory` 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。メモリーブロックのサイズを指定すると、削除されるメモリーブロックが自動的に選択されます。ただし、`mblock` プロパティに `PA-start:size` 値を指定すると、指定されたメモリーブロックが明示的に削除されます。

`-g` オプションを使用してリソースグループを指定した場合、削除対象として選択されるメモリーはすべて、そのリソースグループのものになります。

構文:

```
ldm remove-memory [--auto-adj] size[unit] domain-name
```

```
ldm remove-memory mblock=PA-start:size[,PA-start:size[,...]] domain-name
```

```
ldm remove-memory -g resource-group [-s size[unit]] domain-name
```

ここでは:

- `--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されるメモリーのサイズ (バイト単位) です。異なるサイズの単位が必要な場合は、`unit` を、大文字または小文字を使用して次のいずれかの値として指定します。
 - G バイトの場合は G
 - K バイトの場合は K
 - M バイトの場合は M
- `mblock=PA-start:size` には、ドメインから削除される物理メモリーブロックを 1 つ以上指定します。`PA-start` には、メモリーブロックの開始物理アドレスを 16 進形式で指定します。`size` は、ドメインから削除されるメモリーブロックのサイズ (単位を含む) です。このプロパティを使用して DIMM の物理アドレスを指定することはできないことに注意してください。
- `-g resource-group` は、指定されたりソースグループ内のリソースに対して操作が実行されるように指定します。
- `-s size[unit]` は、削除するメモリーの量を指定します。このオプションが指定されていない場合、このコマンドは、指定されたドメインにバインドされている指定されたりソースグループからすべてのメモリーを削除しようとします。このオプションは、`-g` オプションが指定されている場合にのみ使用できます。
- `domain-name` は、メモリーが削除される論理ドメインを指定します。

遅延再構成モードの入力

`start-reconf` サブコマンドにより、ドメインが遅延再構成モードに入ることができます。遅延再構成は、ルートドメインでのみサポートされています。

Note - `primary` 以外のルートドメインが遅延再構成の状態にある場合は、`add-io`、`set-io`、`remove-io`、`create-vf`、および `destroy-vf` 操作のみを実行できます。

構文:

```
ldm start-reconf domain-name
```

遅延再構成処理の取り消し

`cancel-reconf` サブコマンドは、遅延再構成を取り消します。遅延再構成は、ルートドメインでのみサポートされています。

構文:

```
ldm cancel-reconf domain-name
```

操作の取り消し

`cancel-operation` サブコマンドは、論理ドメインに対する遅延再構成 (`reconf`)、メモリーの動的再構成の削除 (`memdr`)、またはドメイン移行 (`migration`) を取り消します。`reconf` 操作は、ルートドメインでのみサポートされています。

構文:

```
ldm cancel-operation migration domain-name
```

```
ldm cancel-operation reconf domain-name
```

```
ldm cancel-operation memdr domain-name
```

I/O デバイス

I/O デバイスの追加

`add-io` サブコマンドは、指定された論理ドメインに PCIe バス、デバイス、または仮想機能を動的に追加しようとします。ドメインで動的構成がサポートされていない場合は、コマンドに失敗し、遅延再構成を開始するか、ドメインを停止してからデバイスを追加する必要があります。

`iovs=off` のときにルートドメインにルートコンプレックスを追加する場合は、`create-vf`、`destroy-vf`、`add-io`、または `remove-io` サブコマンドを使用してダイレクト I/O および SR-IOV デバイスを正常に割り当てることはできません。

構文:

```
ldm add-io bus domain-name
```

```
ldm add-io iovs=on|off domain-name
```

```
ldm add-io (device | vf-name) domain-name
```

ここでは:

- `iovs=on|off` は、PCIe バス (ルートコンプレックス) 上の I/O 仮想化 (ダイレクト I/O および SR-IOV) 操作を有効または無効にします。有効にすると、該当するバスのデバイスで I/O 仮想化がサポートされます。`ldm add-io` コマンドは、指定された PCIe バスをルートドメインに再度バインドします。デフォルト値は `off` です。追加しようとする PCIe バスがすでにドメインにバインドされている場合は、コマンドに失敗することに注意してください。
- `bus`、`device`、および `vf-name` は、それぞれ PCIe バス、ダイレクト I/O 割り当て可能デバイス、および PCIe SR-IOV 仮想機能です。オペランドはデバイスパスとしても、仮名としても指定できますが、デバイスの仮名を使用することが推奨されています。

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。

デバイス名に関連付けられた仮名の例を次に示します。

- **PCIe バス。** pci_0 という仮名は pci@400 というデバイスパスと一致します。
- **ダイレクト I/O 割り当て可能デバイス。** PCIE1 という仮名は pci@400/pci@0/pci@c というデバイスパスと一致します。
- **PCIe SR-IOV 仮想機能。** /SYS/MB/NET0/IOVNET.PF0.VF0 という仮名は pci@400/pci@2/pci@0/pci@6/network@0 というデバイスパスと一致します。

指定されたゲストドメインは、アクティブでない状態またはバインドされた状態である必要があります。primary ドメインを指定した場合、このコマンドは遅延再構成を開始します。

- *domain-name* は、バスまたはデバイスが追加される論理ドメインを指定します。

仮想機能のプロパティの設定

set-io サブコマンドは、プロパティ値を変更するか、または新しいプロパティを渡すことによって、仮想機能の現在の構成を変更します。このコマンドは、クラス固有のプロパティとデバイス固有のプロパティの両方を変更できます。

ほとんどのネットワーククラス固有のプロパティは、ルートドメインをリブートしなくても変更できます。ただし、ドメインにバインドされている仮想機能の mtu および mac-addresses プロパティを変更するには、まず該当するドメインを停止するか、ルートドメインで遅延再構成を開始する必要があります。

- すべてのデバイス固有のプロパティでは、物理機能デバイスドライバの接続操作中に該当するプロパティを更新できるように、遅延再構成が開始されます。結果として、ルートドメインのリブートが必要になります。
- このコマンドは、物理機能ドライバが結果として得られる構成を正常に検証できた場合にのみ成功します。

構文:

```
ldm set-io name=value [name=value...] pf-name

ldm set-io iov=on|off bus

ldm set-io [mac-addr=MAC-address] [alt-mac-addr=[auto|MAC-address, [auto|MAC-address, ...]]]
[pvid=[pvid]] [vid=[vid1, vid2, ...]] [mtu=size] [name=value...] net-vf-name

ldm set-io name=[value...] ib-pf-name

ldm set-io [bw-percent=[value]] [port-wwn=value node-wwn=value] fc-vf-name
```

ここでは:

- *alt-mac-addr=auto|MAC-address, [auto|MAC-address, ...]* は、代替 MAC アドレスのコンマ区切りリストです。有効な値は、数値の MAC アドレスと、システムで代替 MAC アドレスが生成されるようにリクエストするために 1 回以上使用できる auto キーワードです。auto キーワードと数値の MAC アドレスは混在できます。

数値の MAC アドレスは、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。

このプロパティ値は、バインドされたドメイン内の仮想ネットワークデバイス上では変更できません。まず、そのドメインを停止するか、またはルートドメインで遅延再構成を開始する必要があります。

1つ以上の代替 MAC アドレスを割り当てると、このデバイス上に1つ以上の仮想 NIC (vNIC) を作成できます。vNIC ごとに1つの代替 MAC アドレスが使用されるため、割り当てられた MAC アドレスの数によって、デバイス上に作成できる vNIC の数が決まります。代替 MAC アドレスが指定されていない場合は、このデバイス上に vNIC を作成しようとしても失敗します。詳細は、Oracle Solaris 11 ネットワークドキュメントおよび [Oracle VM Server for SPARC 3.4 管理ガイド](#) の第 13 章、“仮想ネットワークの使用”を参照してください。

- `iov=on|off` は、PCIe バス (ルートコンプレックス) 上の I/O 仮想化 (ダイレクト I/O および SR-IOV) 操作を有効または無効にします。有効にすると、該当するバスのデバイスで I/O 仮想化がサポートされます。デフォルト値は `off` です。

`iov` プロパティ値を変更するには、ルートコンプレックスをドメインにバインドし、ドメインを遅延再構成の状態にする必要があります。

- `bw-percent=[value]` は、ファイバチャネル仮想機能に割り当てられる帯域幅の割合を指定します。有効な値は、0 から 100 までです。ファイバチャネル物理機能の仮想機能に割り当てられた帯域幅の合計値が 100 を超えることはできません。その仮想機能が、同じ物理機能を共有するほかの仮想機能によってまだ予約されていない帯域幅の正当な割当量を受け取れるように、デフォルト値は 0 になっています。
- `node-wwn=value` は、ファイバチャネル仮想機能のノードの World Wide Name を指定します。ゼロ以外の値が有効です。デフォルトでは、この値は自動的に割り当てられます。この値を手動で指定する場合は、`port-wwn` プロパティの値も指定する必要があります。

IEEE 形式では、2 バイトのヘッダーのあとに、OUI を含む埋め込みの MAC-48 または EUI-48 アドレスが続きます。最初の 2 バイトは 16 進数の 10:00 または 2x:xx (x はベンダー固有) であり、そのあとに 3 バイトの OUI と 3 バイトのベンダー固有のシリアル番号が続きます。

- `port-wwn=value` は、ファイバチャネル仮想機能のポートの World Wide Name を指定します。ゼロ以外の値が有効です。デフォルトでは、この値は自動的に割り当てられます。この値を手動で指定する場合は、`node-wwn` プロパティの値も指定する必要があります。

IEEE 形式では、2 バイトのヘッダーのあとに、OUI を含む埋め込みの MAC-48 または EUI-48 アドレスが続きます。最初の 2 バイトは 16 進数の 10:00 または 2x:xx (x はベンダー固有) であり、そのあとに 3 バイトの OUI と 3 バイトのベンダー固有のシリアル番号が続きます。

- `name=value` は、設定するプロパティの名前と値の組み合わせです。
- `pf-name` は、物理機能の名前です。
- `bus` は、PCIe バスの名前です。
- `net-vf-name` は、ネットワーク仮想機能の名前です。

-
- *ib-pf-name* は、InfiniBand 物理機能の名前です。
 - *fc-vf-name* は、ファイバチャネル仮想機能の名前です。

物理機能のプロパティの設定

`set-io` サブコマンドは、物理機能の構成を変更します。物理機能デバイス固有のプロパティのみがサポートされています。物理機能デバイスドライバの接続操作中にプロパティが適用されるため、プロパティを変更すると遅延再構成が発生します。

プロパティ値は、整数または文字列である必要があります。プロパティ値のタイプおよび特定のプロパティが設定可能かどうかを判定するには、`ldm list-io -d` コマンドを実行します。

`ldm set-io` コマンドは、結果として得られる構成を物理機能ドライバが正常に検証した場合にのみ成功することに注意してください。

構文:

```
ldm set-io name=value [name=value...] pf-name
```

ここでは:

- *name=value* は、設定するプロパティの名前と値の組み合わせです。
- *pf-name* は、物理機能の名前です。

I/O デバイスの削除

`remove-io` サブコマンドは、指定されたドメインから PCIe バス、デバイス、または仮想機能を削除します。

構文:

```
ldm remove-io [-n] (bus | device | vf-name) domain-name
```

ここでは:

- `-n` は、コマンドが成功するかどうかを判定するために予行演習を行います。実際には I/O デバイスは削除されません。
- *bus*、*device*、および *vf-name* は、それぞれ PCIe バス、ダイレクト I/O 割り当て可能デバイス、および PCIe SR-IOV 仮想機能です。オペランドはデバイスパスとしても、仮名としても指定できますが、デバイスの仮名を使用することが推奨されています。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。

デバイス名に関連付けられた仮名の例を次に示します。

- **PCIe バス。** `pci_0` という仮名は `pci@400` というデバイスパスと一致します。
- **ダイレクト I/O 割り当て可能デバイス。** `PCIE1` という仮名は `pci@400/pci@0/pci@c` というデバイスパスと一致します。
- **PCIe SR-IOV 仮想機能。** `/SYS/MB/NET0/IOVNET.PF0.VF0` という仮名は `pci@400/pci@2/pci@0/pci@6/network@0` というデバイスパスと一致します。

指定されたゲストドメインは、アクティブでない状態またはバインドされた状態である必要があります。primary ドメインを指定した場合、このコマンドは遅延再構成を開始します。

- `domain-name` は、バスまたはデバイスが削除され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

`add-vsw` サブコマンドは、指定された論理ドメインに仮想スイッチを追加します。

構文:

```
ldm add-vsw [-q] [default-vlan-id=VLAN-ID] [pvid=port-VLAN-ID] [vid=VLAN-ID1,VLAN-ID2,...]
[linkprop=phys-state] [mac-addr=MAC-address] [net-dev=device] [mode=sc] [mtu=size]
[id=switch-ID] [inter-vnet-link=auto|on|off]] vswitch-name domain-name
```

ここでは:

- `-q` は、`net-dev` プロパティで指定されたネットワークデバイスのパスの検証を無効にします。特に論理ドメインがフル構成ではない場合に、このオプションによってコマンドをより迅速に実行できるようになります。
- `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属するデフォルトの VLAN をタグなしモードで指定します。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 3.4 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、値は `phys-state` であり、これは基盤の物理デバイスがそのリンクステータスを報告する場合にのみ有効になります。
- `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。詳細については、[Oracle VM Server for SPARC 3.4 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=MAC-address` 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またはアグリゲーションへのパスです。システムは、`-q` オプションが指定されない場合、パスが実際のネットワークデバイスを参照していることを検証します。

VLAN が含まれるパス上でこのプロパティを設定する場合は、VLAN タグを含むパス名を使用しないでください。

- `mode=sc` は、論理ドメイン環境で Oracle Solaris Cluster 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 Cluster 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 Cluster 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 Cluster を論理ドメイン環境で実行し、ゲストドメインを Oracle Solaris Cluster 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 Cluster 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。有効な値は、1500-16000 の範囲です。
- `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。
- `inter-vnet-link=auto|on|off` は、同じ仮想スイッチに接続された仮想ネットワークデバイスの各ペア間にチャンネルを割り当てるかどうかを指定します。この動作は、ゲスト間のパフォーマンスを向上させます。

値が `on` の場合、`inter-vnet` LDC チャンネルが割り当てられます。値が `off` の場合、`inter-vnet` LDC チャンネルは割り当てられません。値が `auto` の場合、仮想スイッチ内の仮想ネットワーク数が 8 つを超えていないかぎり、`inter-vnet` LDC チャンネルが割り当てられます。デフォルトの 8 つの仮想ネットワーク数を超えると、`inter-vnet` LDC チャンネルが無効になります。デフォルトの仮想ネットワーク数を変更するには、`ldmd/auto_inter_vnet_link_limit` SMF プロパティ値を変更します。デフォルト値は `auto` 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的な名前です。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。
- `domain-name` 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

`set-vsw` 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。

構文:

```
ldm set-vsw [-q] [pvid=port-VLAN-ID] [vid=[[+|-]VLAN-ID1,VLAN-ID2,...]] [mac-addr=MAC-address]
[net-dev=device] [linkprop=[phys-state]] [mode=[sc]] [mtu=size]
[inter-vnet-link=[auto|on|off]] vswitch-name
```

ここでは:

- `-q` は、`net-dev` プロパティで指定されたネットワークデバイスのパスの検証を無効にします。特に論理ドメインがフル構成ではない場合に、このオプションによってコマンドをより迅速に実行できるようになります。
- `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 3.4 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-ID1,VLAN-ID2,...]]` 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。オプションの `+` 文字を使用して、リストに 1 つまたは複数の VLAN ID を追加します。オプションの `-` 文字を使用して、リストから 1 つまたは複数の VLAN ID を削除します。[Oracle VM Server for SPARC 3.4 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=MAC-address` 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、このスイッチが動作するネットワークデバイスまたはアグリゲーションへのパスです。システムは、`-q` オプションが指定されない場合、パスが実際のネットワークデバイスを参照していることを検証します。

VLAN が含まれるパス上でこのプロパティを設定する場合は、VLAN タグを含むパス名を使用しないでください。

`ldm set-vsw` コマンドを使用して `net-dev` プロパティ値を指定または更新すると、`primary` ドメインが遅延再構成モードに入ることにご注意してください。

- `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、値は `phys-state` であり、これは基盤の物理デバイスがそのリンクステータスを報告する場合にのみ有効になります。`linkprop=` を設定して、`linkprop` プロパティ値をクリアできます。
- `mode=sc` は、論理ドメイン環境で Oracle Solaris Cluster 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 Cluster 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 Cluster 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 Cluster を論理ドメイン環境で実行し、ゲストドメインを Oracle Solaris Cluster 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 Cluster 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。有効な値は、1500-16000 の範囲です。
- `inter-vnet-link=auto|on|off` は、同じ仮想スイッチに接続された仮想ネットワークデバイスの各ペア間にチャンネルを割り当てるかどうかを指定します。この動作は、ゲスト間のパフォーマンスを向上させます。
値が `on` の場合、`inter-vnet` LDC チャンネルが割り当てられます。値が `off` の場合、`inter-vnet` LDC チャンネルは割り当てられません。値が `auto` の場合、仮想スイッチ内の仮想ネットワーク数が 8 つを超えていないかぎり、`inter-vnet` LDC チャンネルが割り当てられます。8 つの仮想ネットワークを超えると、`inter-vnet` LDC チャンネルが無効になります。デフォルト値は `auto` 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

`remove-vsw` サブコマンドは、仮想スイッチを削除します。

構文:

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

ここでは:

- `-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

`add-vnet` サブコマンドは、指定され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=MAC-address] [pvid=port-VLAN-ID] [pvlan=secondary-vid, pvlan-type]
[protection=protection-type[, protection-type], ...] [auto-alt-mac-addr=num]
[custom=[enable|disable]] [custom/max-vlans=num] [custom/max-mac-addr=num]
[allowed-ips=IP-address[, IP-address]...] [priority=high|medium|low] [cos=0-7]
[allowed-dhcp-cids=[MAC-address|hostname, MAC-address|hostname, ...]]
[alt-mac-addr=auto|MAC-address[, auto|MAC-address, ...]] [vid=VLAN-ID1,VLAN-ID2,...]
[linkprop=phys-state] [id=network-ID] [mtu=size] [maxbw=value] if-name vswitch-name
domain-name
```

ここでは:

- `custom=[enable|disable]` は、信頼されるホストから仮想ネットワークデバイスに割り当てることができる VLAN および MAC アドレスの最大数のカスタム設

定を有効または無効にします。custom=enabled の場合、alt-mac-addr プロパティを使用して代替 MAC アドレスを、または vid プロパティを使用して VID を指定することはできません。VLAN ID または代替 MAC アドレスが構成されている場合は custom=enable を設定できません。そのため、カスタム設定を有効にするには、最初に alt-mac-addr および vid プロパティ値をクリアします。PVLAN を同時に有効にすることはできません。有効な値は enable および disable です。デフォルト値は disable です。

custom=enable は動的に設定できますが、custom=disable を設定するには、ドメインを停止する必要があります。

- custom/max-vlans=num は、信頼されるホストから仮想ネットワークデバイスに割り当てることができる VLAN の最大数を指定します。デフォルト値は 4096 です。custom/max-vlans プロパティの値は動的に削減できないことに注意してください。
- custom/max-mac-addr=num は、信頼されるホストから仮想ネットワークデバイスに割り当てることができる MAC アドレスの最大数を指定します。デフォルト値は 4096 です。
- mac-addr=MAC-address 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。
- alt-mac-addr=auto|MAC-address, [auto|MAC-address, ...] は、代替 MAC アドレスのコンマ区切りリストです。有効な値は、数値の MAC アドレスと、システムで代替 MAC アドレスが生成されるようにリクエストするために 1 回以上使用できる auto キーワードです。auto キーワードと数値の MAC アドレスは混在できます。数値の MAC アドレスは、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。

1 つ以上の代替 MAC アドレスを割り当てると、このデバイス上に 1 つ以上の仮想 NIC (vNIC) を作成できます。vNIC ごとに 1 つの代替 MAC アドレスが使用されるため、割り当てられた MAC アドレスの数によって、デバイス上に作成できる vNIC の数が決まります。代替 MAC アドレスが指定されていない場合は、このデバイス上に vNIC を作成しようとしても失敗します。詳細は、Oracle Solaris 11 ネットワークドキュメントおよび [Oracle VM Server for SPARC 3.4 管理ガイドの第 13 章、「仮想ネットワークの使用」](#) を参照してください。
- auto-alt-mac-addr=[+]num は仮想ネットワーク用に構成する自動代替 MAC アドレスの最大数を指定します。オプションの + 文字を使用して、1 つ以上かつ最大数までの代替 MAC アドレスを追加します。
- 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 3.4 管理ガイドの「VLAN のタグ付けの使用」](#) を参照してください。
- pvlan=secondary-vid, pvlan-type は、プライベート VLAN (PVLAN) を構成します。プライマリ VLAN は、トラフィックを下流のセカンダリ VLAN に転送します。この VLAN は、隔離 VLAN とコミュニティ VLAN のどちらでもかまいません。pvid プロパティも指定する必要があります。pvlan プロパティは、

PVLAN の *secondary-vid* (1 から 4094 までの値) と、次のいずれかの値である *pvlan-type* を指定します。

- **isolated** – 隔離 PVLAN に関連付けられたポートは、バックエンドネットワークデバイス上のすべてのピア仮想ネットワークおよび Oracle Solaris 仮想 NIC から隔離されます。パケットは、その PVLAN に指定された値に基づく外部ネットワークにのみ到達します。
- **community** – コミュニティー PVLAN に関連付けられたポートは、同じコミュニティ PVLAN 内に存在するほかのポートと通信できますが、その他のすべてのポートから隔離されます。パケットは、その PVLAN に指定された値に基づく外部ネットワークに到達します。
- **[vid=[+|-]VLAN-ID1,VLAN-ID2,...]** は、仮想ネットワークデバイスがメンバーとして属する必要がある 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 3.4 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。有効な値は、1500-16000 の範囲です。
- **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、値は **phys-state** であり、これは基盤の物理デバイスがそのリンクステータスを報告する場合にのみ有効になります。
- **maxbw=value** は、指定されたポートの最大帯域幅制限を M ビット/秒で指定します。この制限は、外部ネットワーク (具体的には、仮想スイッチを通じて送信されるトラフィック) からの帯域幅が指定された値を超えないことを保証します。この帯域幅制限は、**inter-vnet** リンク上のトラフィックには適用されません。帯域幅制限は、任意の高い値に設定できます。その値が、ネットワークバックエンドデバイスによってサポートされている帯域幅よりも大きい場合は無視されます。
- **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。
- **allowed-dhcp-cids=MAC-address|hostname,MAC-address|hostname,...**
MAC アドレスまたはホスト名のコンマ区切りリストを指定します。*hostname* にはホスト名か、またはドメイン名を含む完全修飾ホスト名を指定できます。この名前は英字で始まる必要があります。*MAC-address* は標準のオクテット記述法での数値の MAC アドレス (80:00:33:55:22:66 など) です。詳細は、**dhcp_nospoof** を参照してください。
- **allowed-ips=IP-address[,IP-address,...]**
IP アドレスのコンマ区切りリストを指定します。詳細は、**ip_nospoof** を参照してください。
- **cos=0-7**

リンク上のアウトバウンドパケットに関連付けられたサービスクラス (802.1p) 優先度を指定します。このプロパティを設定すると、リンク上のすべてのアウトバウンドパケットに VLAN タグが付けられ、その優先度フィールドがこのプロパティ値に設定されます。有効な値は 0-7 です。7 はもっとも高いサービスクラス、0 はもっとも低いサービスクラスです。デフォルト値は 0 です。

- **priority=value**

リンクの相対的な優先度を指定します。これは、システム内でのパケット処理のスケジューリングに使用されます。有効な値は **high**、**medium**、および **low** です。デフォルト値は **medium** です。

- **protection=protection-type[, protection-type]...**

保護のタイプ (*protection-type*) を、保護タイプのビット単位の OR の形式で指定します。デフォルトでは、保護タイプは使用されません。次の値をコマンドで区切ります。

- **mac_nospoof** は、MAC アドレスのアンチスプーフィングを有効にします。送信パケットの発信元 MAC アドレスが、そのリンクの構成済み MAC アドレスに一致している必要があります。一致していないパケットは削除されます。この値には、データリンク MAC 構成の保護が含まれています。

- **ip_nospoof** は、IP アドレスのアンチスプーフィングを有効にします。この保護タイプは、1 つ以上の IP アドレス (IPv4 または IPv6) を指定する **allowed-ips** リンクプロパティと連動します。アウトバウンド IP パケットは、その発信元アドレスが **allowed-ips** リストで指定されている場合に通過できます。アウトバウンド ARP パケットは、その送信側プロトコルアドレスが **allowed-ips** リストに含まれている場合に通過できます。この値には、IP アドレス構成の保護が含まれています。

- **dhcp_nospoof** は、DHCP クライアント ID (CID) とハードウェアアドレスのアンチスプーフィングを有効にします。デフォルトでは、この値により、デバイスポートノードの構成済み MAC アドレスのアンチスプーフィングが有効になります。 **allowed-dhcp-cids** プロパティが指定されている場合は、そのノードの DHCP クライアント ID に対する DHCP アンチスプーフィングが有効になります。

- **restricted** は、送信パケットのタイプを IPv4、IPv6、および ARP パケットのみに制限するパケット制限を有効にします。

- **if-name** は、後続の **set-vnet** または **remove-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。

- **domain-name** 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

`set-vnet` サブコマンドは、指定され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=MAC-address] [vswitch=vswitch-name] [mode=] [pvid=port-VLAN-ID]
[pvlan=[secondary-vid, pvlan-type]] [protection=[[+|-]protection-type[, protection-type], ...]]
[allowed-ips=[[+|-]IP-address[, IP-address]...]] [priority=high|medium|low] [cos=0-7]
[allowed-dhcp-cids=[[+|-]MAC-address|hostname, MAC-address|hostname, ...]]
[alt-mac-addr=[[+|-]auto|MAC-address[, auto|MAC-address, ...]]] [linkprop=[phys-state]]
[mtu=size] [vid=[[+|-]VLAN-ID1,VLAN-ID2,...]] [auto-alt-mac-addr=[[+]num]
[custom=[enable|disable]] [custom/max-vlans=[num]] [custom/max-mac-addr=[[+]num]
[maxbw=[value]] if-name domain-name
```

ここでは:

- `custom=[enable|disable]` は、信頼されるホストから仮想ネットワークデバイスに割り当てることができる VLAN および MAC アドレスの最大数のカスタム設定を有効または無効にします。`custom=enable` の場合、`alt-mac-addr` プロパティを使用して代替 MAC アドレスを、または `vid` プロパティを使用して VID を指定することはできません。VLAN ID または代替 MAC アドレスが構成されている場合は `custom=enable` を設定できません。そのため、カスタム設定を有効にするには、最初に `alt-mac-addr` および `vid` プロパティ値をクリアします。PVLAN を同時に有効にするにはできません。有効な値は `enable` および `disable` です。デフォルト値は `disable` です。
`custom=enable` は動的に設定できますが、`custom=disable` を設定するには、ドメインを停止する必要があります。
- `custom/max-vlans=num` は、信頼されるホストから仮想ネットワークデバイスに割り当てることができる VLAN の最大数を指定します。デフォルト値は 4096 です。
`custom/max-vlans` プロパティの値は動的に削減できないことに注意してください。
- `custom/max-mac-addr=num` は、信頼されるホストから仮想ネットワークデバイスに割り当てることができる MAC アドレスの最大数を指定します。デフォルト値は 4096 です。
- `mac-addr=MAC-address` 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。
- `auto-alt-mac-addr=[[+]num` は仮想ネットワーク用に構成する自動代替 MAC アドレスの最大数を指定します。オプションの + 文字を使用して、1 つ以上かつ最大数までの代替 MAC アドレスを追加します。
- `alt-mac-addr=[[+|-]auto|MAC-address, [auto|MAC-address, ...]]` は、代替 MAC アドレスのコンマ区切りリストです。有効な値は、数値の MAC アドレスと、システムで代替 MAC アドレスが生成されるようにリクエストするために 1 回以上使用できる `auto` キーワードです。`auto` キーワードと数値の MAC アドレスは混在できます。数値の MAC アドレスは、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。

1つ以上の代替 MAC アドレスを割り当てると、このデバイス上に1つ以上の仮想 NIC (vNIC) を作成できます。vNIC ごとに1つの代替 MAC アドレスが使用されるため、割り当てられた MAC アドレスの数によって、デバイス上に作成できる vNIC の数が決まります。代替 MAC アドレスが指定されていない場合は、このデバイス上に vNIC を作成しようとしても失敗します。オプションの + 文字を使用して、リストに1つまたは複数の代替 MAC アドレスを追加します。オプションの - 文字を使用して、リストから1つまたは複数の代替 MAC アドレスを削除します。詳細は、Oracle Solaris 11 ネットワークドキュメントおよび[Oracle VM Server for SPARC 3.4 管理ガイドの第13章](#)、“仮想ネットワークの使用”を参照してください。

- `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=` は、非推奨のハイブリッドモードを有効にするために使用されるプロパティです。ハイブリッド I/O 構成を作成できなくなります。このプロパティの値のみをクリアできます。
- `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 3.4 管理ガイドの“VLAN のタグ付けの使用”](#)を参照してください。
- `pvlan=secondary-vid, pvlan-type` は、PVLAN を構成します。PVLAN は、トラフィックを下流のセカンダリ VLAN に転送します。この VLAN は、隔離された VLAN とコミュニティ VLAN のどちらでもかまいません。少なくとも1つの `pvid` が指定されている必要があります。`pvlan` プロパティは、PVLAN の `secondary-vid` (1 から 4094 までの値) と、次のいずれかの値である `pvlan-type` を指定します。
 - `isolated` – 隔離 PVLAN に関連付けられたポートは、バックエンドネットワークデバイス上のすべてのピア仮想ネットワークおよび Oracle Solaris 仮想 NIC から隔離されます。パケットは、その PVLAN に指定された値に基づく外部ネットワークにのみ到達します。
 - `community` – コミュニティ PVLAN に関連付けられたポートは、同じコミュニティ PVLAN 内に存在するほかのポートと通信できますが、その他のすべてのポートから隔離されます。パケットは、その PVLAN に指定された値に基づく外部ネットワークに到達します。
- `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、値は `phys-state` であり、これは基盤の物理デバイスがそのリンクステータスを報告する場合にのみ有効になります。`linkprop=` を設定して、`linkprop` プロパティ値をクリアできます。
- `[vid=[[+|-]VLAN-ID1,VLAN-ID2,...]]` は、仮想ネットワークデバイスがメンバーとして属する必要のある1つ以上の VLAN をタグ付きモードで指定します。オプションの + 文字を使用して、リストに1つまたは複数の VLAN ID を追加します。オプションの - 文字を使用して、リストから1つまたは複数の VLAN ID を削除し

ます。Oracle VM Server for SPARC 3.4 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。有効な値は、1500-16000 の範囲です。
- `maxbw=value` は、指定されたポートの最大帯域幅制限を M ビット/秒で指定します。この制限は、外部ネットワーク (具体的には、仮想スイッチを通じて送信されるトラフィック) からの帯域幅が指定された値を超えないことを保証します。この帯域幅制限は、`inter-vnet` リンク上のトラフィックには適用されません。帯域幅制限は、任意の高い値に設定できます。その値が、ネットワークバックエンドデバイスによってサポートされている帯域幅よりも大きい場合は無視されます。
- `allowed-dhcp-cids=[[+|-]MAC-address|hostname,MAC-address|hostname,...]`
MAC アドレスまたはホスト名のコンマ区切りリストを指定します。`hostname` にはホスト名か、またはドメイン名を含む完全修飾ホスト名を指定できます。この名前は英字で始まる必要があります。`MAC-address` は標準のオクテット記述法での数値の MAC アドレス (80:00:33:55:22:66 など) です。オプションの + 文字を使用して、リストに 1 つまたは複数の MAC アドレスまたはホスト名を追加します。オプションの - 文字を使用して、リストから 1 つまたは複数の MAC アドレスまたはホスト名を削除します。詳細は、`dhcp_nospoof` を参照してください。
- `allowed-ips=[[+|-]IP-address[,IP-address,...]]`
IP アドレスのコンマ区切りリストを指定します。オプションの + 文字を使用して、1 つまたは複数の IP アドレスをリストに追加します。オプションの - 文字を使用して、1 つまたは複数の IP アドレスをリストから削除します。詳細は、`ip_nospoof` を参照してください。
- `cos=0-7`
リンク上のアウトバウンドパケットに関連付けられたサービスクラス (802.1p) 優先度を指定します。このプロパティを設定すると、リンク上のすべてのアウトバウンドパケットに VLAN タグが付けられ、その優先度フィールドがこのプロパティ値に設定されます。有効な値は 0-7 です。7 はもっとも高いサービスクラス、0 はもっとも低いサービスクラスです。デフォルト値は 0 です。
- `priority=value`
リンクの相対的な優先度を指定します。これは、システム内でのパケット処理のスケジューリングに使用されます。有効な値は `high`、`medium`、および `low` です。デフォルト値は `medium` です。
- `protection=[[+|-]protection-type[,protection-type]...]`
保護のタイプ (`protection-type`) を、保護タイプのビット単位の OR の形式で指定します。デフォルトでは、保護タイプは使用されません。オプションの + 文字を使用して、1 つまたは複数の保護タイプをリストに追加します。オプションの - 文字を使用して、1 つまたは複数の保護タイプをリストから削除します。次の値をコマンドで区切ります。
 - `mac_nospoof` は、MAC アドレスのアンチスプーフィングを有効にします。送信パケットの発信元 MAC アドレスが、そのリンクの構成済み MAC アドレス

に一致している必要があります。一致していないパケットは削除されます。この値には、データリンク MAC 構成の保護が含まれています。

- `ip_nospoof` は、IP アドレスのアンチスプーフィングを有効にします。この保護タイプは、1 つ以上の IP アドレス (IPv4 または IPv6) を指定する `allowed-ips` リンクプロパティと連動します。アウトバウンド IP パケットは、その発信元アドレスが `allowed-ips` リストで指定されている場合に通過できます。アウトバウンド ARP パケットは、その送信側プロトコルアドレスが `allowed-ips` リストに含まれている場合に通過できます。この値には、IP アドレス構成の保護が含まれています。
- `dhcp_nospoof` は、DHCP クライアント ID (CID) とハードウェアアドレスのアンチスプーフィングを有効にします。デフォルトでは、この値により、デバイスポートノードの構成済み MAC アドレスのアンチスプーフィングが有効になります。`allowed-dhcp-cids` プロパティが指定されている場合は、そのノードの DHCP クライアント ID に対する DHCP アンチスプーフィングが有効になります。
- `restricted` は、送信パケットのタイプを IPv4、IPv6、および ARP パケットのみに制限するパケット制限を有効にします。
- `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ーフェイス名です。
- `domain-name` 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

`remove-vnet` サブコマンドは、指定され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。

構文:

```
ldm remove-vnet [-f] if-name domain-name
```

ここでは:

- `-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ーフェイス名です。
- `domain-name` 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

`add-vds` サブコマンドは、指定された論理ドメインに仮想ディスクサーバーを追加します。

構文:

```
ldm add-vds service-name domain-name
```

ここでは:

- *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。
- *domain-name* 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

`remove-vds` サブコマンドは、仮想ディスクサーバーを削除します。

構文:

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

ここでは:

- `-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。



Caution `-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

`add-vdsdev` 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 3.4 管理ガイドの第 11 章, “仮想ディスクの使用”](#)を参照してください。

構文:

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

ここでは:

- `-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。
- `-q` は、*backend* オペラントで指定された仮想ディスクのバックエンドの検証を無効にします。特に論理ドメインまたはバックエンドがフル構成ではない場合に、このオプションによってコマンドをより迅速に実行できるようになります。

- `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マンドで区切って、`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。
- `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、Solaris Volume Manager、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。システムは、`-q` オプションが指定されない場合、`backend` で指定された場所が存在し、仮想ディスクのバックエンドとして使用できることを検証します。
- `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

`set-vdsdev` 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 3.4 管理ガイド](#)を参照してください。

構文:

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

ここでは:

- `-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。
- `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

`remove-vdsdev` 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。

構文:

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

ここでは:

- `-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的な名前です。
- `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。



Caution - `-f` オプションを指定しないと、仮想ディスクサーバーデバイスがビジー状態である場合、`remove-vdsdev` サブコマンドはそ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

`add-vdisk` サブコマンドは、指定され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。

`disk-name` が `mpgroup` ディスクである場合、`ldm add-vdisk` コマンドは次を実行します。

- 指定されたドメインに仮想ディスクを追加します。
- その仮想ディスクにアクセスするための最初のパスとして `volume-name@service-name` を選択します。

構文:

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

ここでは:

- `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。
- `domain-name` 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

`set-vdisk` サブコマンドは、指定され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。

`mpgroup` ディスクに使用する場合を除き、このコマンドは、ドメインがバインドされているか、またはアクティブでない場合のみ使用できます。

disk-name が *mpgroup* ディスクである場合は、`ldm set-vdisk` コマンドを使用して、*volume* プロパティの値として仮想ディスクへの最初のパスを指定できます。選択されたパスとして指定するパスが *mpgroup* にすでに属している必要があります。

更新された仮想ディスクドライバが実行されている場合は、動的なパス選択を使用できます。これらの更新されたドライバを含む Oracle Solaris OS のバージョンを確認するには、[Oracle VM Server for SPARC 3.4 管理ガイド](#)を参照してください。

動的なパス選択は、`ldm set-vdisk` コマンドを使用して *volume* プロパティに *volume-name@service-name* という形式の値を設定することによって *mpgroup* ディスク内の最初のパスが変更された場合に発生します。選択されたパスに切り替えることができるのは、動的なパス選択をサポートするアクティブドメインだけです。更新されたドライバが実行されていない場合は、Oracle Solaris OS がディスクインスタンスをリロードしたとき、または次のドメインリブート時にこのパスが選択されます。

構文:

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

ここでは:

- *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。
- *domain-name* は、以前に仮想ディスクが追加された既存の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

`remove-vdisk` サブコマンドは、指定された論理ドメインから仮想ディスクを削除します。

構文:

```
ldm remove-vdisk [-f] disk-name domain-name
```

ここでは:

- *-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。

- *domain-name* 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 SCSI ホストバスアダプタ – クライアント

次のセクションで説明するサブコマンドは、vHBA リソースに影響します。vHBA とは、Sun Common SCSI Architecture (SCSA) インタフェースをサポートする仮想 SCSI ホストバスアダプタ (HBA) のことです。vHBA は、SCSI コマンドをクライアントの SCSI ターゲットドライバから仮想ストレージエリアネットワーク (SAN) vSAN サービスによって管理される SCSI デバイスに送信します。

仮想 SCSI ホストバスアダプタの作成

`add-vhba` サブコマンドは、指定された論理ドメインで仮想 SCSI HBA を作成します。

```
ldm add-vhba [id=vHBA-ID] vHBA-name vSAN-name domain-name
```

ここでは:

- *id=vHBA-ID* は、新しい仮想 SCSI HBA デバイスの ID を指定します。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のみ、このプロパティを設定します。
- *vHBA-name* は、仮想 SCSI HBA の名前を指定します。
- *vSAN-name* は、接続する既存の仮想 SAN サーバーデバイスの名前を指定します。
- *domain-name* は、論理ドメインの名前を指定します。

仮想 SCSI ホストバスアダプタの変更

`set-vhba` サブコマンドによって、指定した論理ドメインの仮想 SCSI HBA のタイムアウト値を指定できます。

```
ldm set-vhba [timeout=seconds] vHBA-name domain-name
```

ここでは:

- *timeout=seconds* は、SCSA コマンドがタイムアウトして失敗するまで、指定された仮想 SCSI HBA インスタンスが待機する時間を指定します。
- *vHBA-name* は、仮想 SCSI HBA の名前を指定します。
- *domain-name* は、論理ドメインの名前を指定します。

仮想 SCSI ホストバスアダプタの再スキャン

`rescan-vhba` サブコマンドを使用すると、指定された仮想 SCSI HBA が、関連付けられた仮想 SAN に対して、仮想 SAN が認識している SCSI デバイスの現在のセットをクエリーします。

vSAN によって管理されている物理 SCSI HBA で SCSI デバイスを作成または削除する場合にこのサブコマンドを使用します。`ldm rescan-vhba` コマンドは、仮想 SCSI HBA および仮想 SAN によって認識される SCSI デバイスのセットを同期します。

```
ldm rescan-vhba vHBA-name domain-name
```

仮想 SCSI ホストバスアダプタの削除

`remove-vhba` サブコマンドは、論理ドメインから仮想 SCSI HBA を削除します。

```
ldm remove-vhba vHBA-name domain-name
```

仮想 SCSI ホストバスアダプタ – サービス

次のセクションで説明するサブコマンドは、vSAN リソースに影響します。vSAN とは、指定された SCSI HBA イニシエータポートの配下にある物理 SCSI デバイスのセットをエクスポートする仮想ストレージエリアネットワーク (SAN) サービスのことです。vSAN の物理デバイスは、vHBA デバイスを介してクライアントからアクセスされます。

仮想ストレージエリアネットワークの作成

`add-vsan` サブコマンドは、論理ドメインで仮想 SAN サーバーを作成します。仮想 SAN は、指定された SCSI HBA イニシエータポートの子であるすべての SCSI デバイスを管理します。

```
ldm add-vsan [-q] iport-path vSAN-name domain-name
```

ここでは:

- `-q` は、サービスドメインの I/O デバイスを検証しません。
- `iport-path` は、仮想 SAN に関連付ける SCSI HBA イニシエータポートを指定します。
`ldm list-hba` コマンドは、`iport-path` 値を取得するときに使用します。
- `vSAN-name` は、仮想 SAN の名前を指定します。
- `domain-name` は、指定された SCSI HBA イニシエータポートが存在する論理ドメインを指定します。

仮想ストレージエリアネットワークの削除

`remove-vsan` サブコマンドは、論理ドメインから仮想 SAN を削除します。

```
ldm remove-vsan vSAN-name
```

ここで、`vSAN-name` は、仮想 SAN の名前を指定します。

仮想コンソール

仮想コンソール端末集配信装置の追加

`add-vcc` サブコマンドは、指定され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。

構文:

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

ここでは:

- `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。
- `domain-name` 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

`set-vcc` 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。

構文:

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

ここでは:

- `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

`remove-vcc` サブコマンドは、指定され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。

構文:

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

ここでは:

- `-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。



Caution `--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。

仮想コンソールのオプションの設定

`set-vcons` サブコマンドは、指定され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。

構文:

```
ldm set-vcons [port=[port-num]] [group=group] [service=vcc-server] [log=[on|off]] domain-name
```

ここでは:

- `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。
- `log=[on|off]` は、仮想コンソールのロギングを有効または無効にします。有効な値は、ロギングを有効にする `on`、ロギングを無効にする `off`、およびデフォルト値にリセットする `NULL` 値 (`log=`) です。デフォルト値は `on` です。
ログデータは、仮想コンソール端末集配信装置サービスを提供するサービスドメイン上の `/var/log/vntsd/domain-name/console-log` というファイルに保存されます。`logadm` コマンドを使用すると、コンソールログファイルがローテーションされます。[logadm\(1M\)](#) および [logadm.conf\(4\)](#) のマニュアルページを参照してください。
- `domain-name` は、仮想コンソールを設定する論理ドメインを指定します。
Oracle Solaris 10 OS または Oracle Solaris 11 OS を実行している任意のゲストドメインに対して、仮想コンソールのロギングを有効にできます。サービスドメインは、Oracle Solaris 11.1 OS を実行する必要があります。

物理機能および仮想機能

仮想機能

PCIe Single-Root I/O Virtualization (SR-IOV) 標準を使用すると、I/O ドメイン間で PCIe デバイスを効率的に共有できます。この標準は、ハードウェアでネイティブに近い I/O パフォーマンスを実現するために実装されます。SR-IOV では、物理デバイスまたは物理機能の仮想化されたインスタンスである仮想機能が数多く作成されます。関連付けられた物理デバイスを共有し、CPU やハイパーバイザのオーバーヘッドを発生させずに I/O を実行できるように、仮想機能は直接 I/O ドメインに割り当てられます。

PCIe 物理機能は、ハードウェアに完全にアクセスでき、仮想機能を作成、構成、および管理する SR-IOV 機能を提供します。システムボード上の PCIe コンポーネントまたは PCIe 差し込み式カードは、1 つ以上の物理機能を提供できます。Oracle Solaris ドライバは、SR-IOV 機能へのアクセスを提供する物理機能と相互に作用します。

PCIe 仮想機能には、データの移動に必要なリソースが含まれています。仮想機能を持つ I/O ドメインはハードウェアにアクセスし、Oracle Solaris 仮想機能ドライバ経由で直接 I/O を実行できます。この動作では、I/O ドメインおよびルートドメイン内の

物理 I/O デバイスで実行されるアプリケーション間の通信パスにあるボトルネックを取り除くことによって、仮想 I/O 機能に関連するオーバーヘッドや待ち時間が回避されます。

これらのコマンドの一部では、次のように物理機能または仮想機能の識別子を指定する必要があります。

```
pf-name ::= pf-pseudonym | pf-path
vf-name ::= vf-pseudonym | vf-path
```

対応するデバイスを参照するときは、仮名形式を使用します。これは、`ldm list-io` 出力の **NAME** 列に表示される名前の形式です。`ldm list-io -l` コマンドを実行すると、出力に名前パス形式が表示されます。`ldm list-io -p` 出力には、**alias=** トークンの値として仮名形式が表示され、**dev=** トークンの値としてパス形式が表示されます。

仮想機能の作成

`create-vf` サブコマンドは、指定された物理機能の仮想機能数を 1 つ増分することによって、指定された物理機能から仮想機能を作成します。新しい仮想機能には、仮想機能番号の順序でもっとも大きい番号が割り当てられます。

仮想機能を動的に作成するには、必ず親ルートコンプレックスに `ioV` プロパティを設定してください。

ネットワーククラス仮想機能には、デフォルトで割り当てられている MAC アドレスを割り当てる必要があります。デフォルトの MAC アドレス値をオーバーライドするには、`mac-addr` プロパティに別の値を指定します。

仮想機能を作成するときに、クラス固有のプロパティとデバイス固有のプロパティを設定することもできます。このコマンドは、結果として得られる構成を物理機能ドライバが正常に検証した場合にのみ成功します。デフォルトでは、新しい仮想機能はどのドメインにも割り当てられていません。ルートドメインがリポートし、ハードウェアで仮想機能がインスタンス化されたあとでなければ、仮想機能を I/O ドメインに割り当てることはできません。前もって計画して、複数の仮想機能を作成するかどうかを決定します。その場合、複数回リポートを実行することを回避するために、続けて作成します。

デバイス固有のプロパティは、物理機能ドライバによってエクスポートされるプロパティによって異なります。詳細については、`ldm list-io -d` コマンドを使用してください。コマンドが正常に完了すると、遅延再構成に関するメッセージが表示されます。

構文:

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

ldm create-vf [alt-mac-addr=[auto|MAC-address, [auto|MAC-address, ...]]] [pvid=pvid]
               [mac-addr=MAC-address] [vid=vid1, vid2, ...] [mtu=size] [name=value...] net-pf-name

ldm create-vf [name=value...] ib-pf-name
```

```
ldm create-vf [port-wnn=value node-wnn=value] [bw-percent=[value]] fc-pf-name
```

ここでは:

- `-n` は、*number* 個の仮想機能を作成します。*number* の代わりに `max` を指定した場合は、指定された物理機能に対して最大数の仮想機能が作成されます。
- `mac-addr=MAC-address` は、Ethernet 仮想機能のプライマリ MAC アドレスです
- `alt-mac-addr=auto|MAC-address, [auto|MAC-address, ...]` は、Ethernet 仮想機能の代替 MAC アドレスのコンマ区切りリストです。有効な値は、数値の MAC アドレスと、システムで代替 MAC アドレスが生成されるようにリクエストするために 1 回以上使用できる `auto` キーワードです。`auto` キーワードと数値の MAC アドレスは混在できます。数値の MAC アドレスは、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。

1 つ以上の代替 MAC アドレスを割り当てると、このデバイス上に 1 つ以上の仮想 NIC (VNIC) を作成できます。VNIC ごとに 1 つの代替 MAC アドレスが使用されるため、割り当てられた MAC アドレスの数によって、デバイス上に作成できる VNIC の数が決まります。代替 MAC アドレスが指定されていない場合は、このデバイス上に VNIC を作成しようとしても失敗します。詳細は、Oracle Solaris 11 ネットワークドキュメントおよび [Oracle VM Server for SPARC 3.4 管理ガイド](#) の第 13 章、“仮想ネットワークの使用”を参照してください。

- `pvid=port-VLAN-ID` は、Ethernet 仮想機能のポート VLAN ID (デフォルト値なし) です
- `vid=VLAN-ID1, VLAN-ID2...` は、Ethernet 仮想機能の整数の VLAN ID のコンマ区切りリストです。
- `mtu=size` は、Ethernet 仮想機能の最大転送単位 (バイト) です。
- `name=value` は、指定するプロパティの名前と値の組み合わせです。
- `bw-percent=[value]` は、ファイバチャネル仮想機能に割り当てられる帯域幅の割合を指定します。有効な値は、0 から 100 までです。ファイバチャネル物理機能の仮想機能に割り当てられた帯域幅の合計値が 100 を超えることはできません。その仮想機能が、同じ物理機能を共有するほかの仮想機能によってまだ予約されていない帯域幅の正当な割当量を受け取れるように、デフォルト値は 0 になっています。
- `node-wnn=value` は、ファイバチャネル仮想機能のノードの World Wide Name を指定します。ゼロ以外の値が有効です。デフォルトでは、この値は自動的に割り当てられます。この値を手動で指定する場合は、`port-wnn` プロパティの値も指定する必要があります。

IEEE 形式では、2 バイトのヘッダーのあとに、OUI を含む埋め込みの MAC-48 または EUI-48 アドレスが続きます。最初の 2 バイトは 16 進数の `10:00` または `2x:xx` (*x* はベンダー固有) であり、そのあとに 3 バイトの OUI と 3 バイトのベンダー固有のシリアル番号が続きます。

- `port-wnn=value` は、ファイバチャネル仮想機能のポートの World Wide Name を指定します。ゼロ以外の値が有効です。デフォルトでは、この値は自動的に割り当てられます。この値を手動で指定する場合は、`node-wnn` プロパティの値も指定する必要があります。

IEEE 形式では、2 バイトのヘッダーのあとに、OUI を含む埋め込みの MAC-48 または EUI-48 アドレスが続きます。最初の 2 バイトは 16 進数の 10:00 または 2x:xx (x はベンダー固有) であり、そのあとに 3 バイトの OUI と 3 バイトのベンダー固有のシリアル番号が続きます。

- *pf-name* は、物理機能の名前です。
- *net-pf-name* は、ネットワーク物理機能の名前です。
- *ib-pf-name* は、InfiniBand 物理機能の名前です。
- *fc-pf-name* は、ファイバチャネル物理機能の名前です。

仮想機能の破棄

`destroy-vf` サブコマンドは、指定された物理機能から仮想機能を破棄します。このコマンドは、次に該当する場合にのみ成功します。

- 現在、指定された仮想機能がどのドメインにも割り当てられていません。
- 指定された仮想機能に対応する物理機能の最後の仮想機能です。
- 結果として得られる構成が物理機能ドライバによって正常に検証されています。
- 仮想機能番号の変更はリブートの一部としてのみ実行できるため、正常な操作によって遅延再構成がトリガーされます。詳細については、`create-vf` サブコマンドを参照してください。

構文:

```
ldm destroy-vf vf-name
```

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

ここでは:

- *vf-name* は、仮想機能の名前です。
- `-n` は、*number* 個の仮想機能を破棄します。*number* の代わりに `max` を指定した場合は、指定された物理機能に対して最大数の仮想機能が破棄されます。
- *pf-name* は、物理機能の名前です。

変数

変数の追加

`add-variable` 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。

構文:

```
ldm add-variable var-name=[value]... domain-name
```

ここでは:

- *var-name=value* は、追加する変数の名前と値の組み合わせです。値はオプションです。

-
- *domain-name* 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

`set-variable` サブコマンドは、論理ドメインの変数を設定します。

構文:

```
ldm set-variable var-name=[value]... domain-name
```

ここでは:

- *var-name=value* は、設定する変数の名前と値の組み合わせです。値はオプションです。
- *domain-name* は、変数を設定する論理ドメインを指定します。

Note - *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

`remove-variable` サブコマンドは、論理ドメインの変数を削除します。

構文:

```
ldm remove-variable var-name... domain-name
```

ここでは:

- *var-name* は、削除する変数の名前です。
- *domain-name* 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

`start-domain` サブコマンドは、1つ以上の論理ドメインを起動します。

構文:

```
ldm start-domain -a
```

```
ldm start-domain -i file
```

```
ldm start-domain domain-name...
```

ここでは:

- `-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。
- *domain-name* 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

`stop-domain` サブコマンドは、次のいずれかを実行して、1つ以上の実行中のドメインを停止します。

- 適切な Logical Domains エージェントが実行されている場合、`shutdown` リクエストをドメインに送信する
- Oracle Solaris OS がブートされている場合は、`uadmin` リクエストをドメインに送信する

デフォルトでは、このコマンドはまず `shutdown` を使用してドメインを停止しようと試みます。ただし、適切な Logical Domains エージェントが使用できない場合、このコマンドは `uadmin` を使用してドメインを停止します。[shutdown\(1M\)](#) および [uadmin\(1M\)](#) のマニュアルページを参照してください。

このデフォルト動作は、`ldmd/default_quick_stop` SMF プロパティを設定することによって変更できます。[ldmd\(1M\) on page 99](#) マニュアルページを参照してください。

構文:

```
ldm stop-domain [[-f | -q] | [[-h | -r | -t sec] [-m msg]]] (-a | domain-name...)
```

ここでは:

- `-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。
- `-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。
- `-h` は `shutdown` コマンドのみを使用して、オペレーティングシステムを停止し、ドメインを停止します。このオプションは、`uadmin` コマンドの使用にフォールバックしません。
- `-m msg` には、停止またはリブートするドメインに送信するメッセージを指定します。

`msg` 文字列に空白が含まれる場合、この文字列を引用符で囲む必要があります。

- `-q` は `uadmin` コマンドを発行することによって、指定されたドメインをすばやく停止します。
- `-r` は `shutdown` コマンドを使用して、オペレーティングシステムを停止してリブートします。
- `-t sec` は、ドメインの停止処理が終了するまで少なくとも `sec` 秒間待機してから、`-q` オプション付きでコマンドを再発行して、まだ実行中の指定されたドメインをすべて停止します。このコマンドは、ドメインの停止リクエストが時間内に完了しない場合にのみ再発行されます。`sec` は、0 よりも大きい値にする必要があります。

特定のドメインで停止リクエストが実行できない場合、該当するドメインに対するコマンドはすぐに `-q` オプションにフォールバックします。

- `domain-name` は、停止する1つ以上の実行中の論理ドメインを指定します。

正常な Oracle Solaris の停止を、それをサポートする Logical Domains エージェントバージョンが実行されていないドメインで実行するには、該当するドメインで shutdown または init 操作を実行します。init(1M) のマニュアルページを参照してください。ldm stop-domain -h コマンドは、正常なシャットダウンを実行し、正常なシャットダウンを使用できないドメインについてエラーを報告します。

Oracle Solaris OS でのパニックの発生

panic-domain サブコマンドは、指定された論理ドメイン上の Oracle Solaris OS でパニックを発生させます。そのように Oracle Solaris OS を構成している場合は、これによってバックトレースおよびクラッシュダンプが提供されます。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。

構文:

```
ldm panic-domain domain-name
```

domain-name 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

ldm --help 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法を表示します。また、ldm コマンドを単独で使用することでも、すべてのサブコマンドの使用法を表示できます。

```
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

ldm --version コマンドは、バージョン情報を表示します。

```
ldm --version
```

```
ldm -v
```

ドメインへのリソースのバインド

bind-domain サブコマンドは、構成されたリソースを論理ドメインにバインド (接続) します。

構文:

```
ldm bind-domain [-f] [-q] -i file
```

```
ldm bind-domain [-f] [-q] domain-name
```

ここでは:

- -f は、無効なネットワークデバイスまたはディスクバックエンドデバイスが検出されても、ドメインのバインディングを強制的に実行しようとします。
- -q は、ネットワークデバイスまたはディスクバックエンドデバイスの検証を無効にするため、コマンドをより迅速に実行できるようになります。

-
- `-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。
 - `domain-name` は、リソースをバインドする論理ドメインを指定します。

ドメインからのリソースのバインド解除

`unbind-domain` 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リソースを解放します。

構文:

```
ldm unbind-domain domain-name
```

`domain-name` は、リソース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

`add-spconfig` 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、SP に格納されます。

構文:

```
ldm add-spconfig config-name
```

```
ldm add-spconfig -r autosave-name [new-config-name]
```

ここでは:

- `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。

指定された構成を自動保存情報に基づいて更新します。更新された構成をインスタンス化するには、このコマンドを使用したあとに電源再投入を実行する必要があります。ことに注意してください。

- `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

`set-spconfig` サブコマンドでは、次のシステム電源再投入時に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。

構文:

```
ldm set-spconfig 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。

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

論理ドメイン構成の削除

`remove-spconfig` サブコマンドは、SP 上に格納されている論理ドメイン構成を削除するとともに、対応するすべての自動保存構成を制御ドメインから削除します。

構文:

```
ldm remove-spconfig [-r] config-name
```

ここでは:

- `-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。

リスト操作

list サブコマンド出力内のフラグ

ドメインの出力 (`ldm list`) では、次のフラグを表示できます。コマンドに長形式および解析可能オプション (`-l -p`) を使用すると、`flags=normal,control,vio-service` のように、フラグが省略されずに表示されます。このオプションを使用しない場合は、`-n-cv-` のように略語が表示されます。リストフラグ値は位置に依存します。次に、左から順に 6 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。

列 1 – ドメインの起動または停止

- `s` 起動または停止

列 2 – ドメインのステータス

- `n` 通常
- `t` 切り替え
- `d` リソースがないために起動できない縮退ドメイン

列 3 – 再構成のステータス

- `d` 遅延再構成
- `r` メモリーの動的再構成

列 4 – 制御ドメイン

- `c` 制御ドメイン

列 5 – サービスドメイン

- v 仮想 I/O サービスドメイン

列 6 – 移行のステータス

- s 移行のソースドメイン
- t 移行のターゲットドメイン
- e 移行時に発生したエラー

ドメインおよび状態の一覧表示

`list-domain` サブコマンドは、論理ドメインおよびその状態を一覧表示します。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。

構文:

```
ldm list-domain [-e] [-l] [-o format] [-p] [-S] [domain-name...]
```

ここでは:

- `-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せん。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。
 - `cmi` – 出力には、CMI デバイスに関する情報が表示されます。これには、ドメインにバインドされている共有メモリーや、関連付けられた仮想 CPU およびコアが含まれます。
 - `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`) サービスが表示されます。
 - `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表示されます。
 - `cpu` – 出力には、CPU スレッド (`vcp`), 物理 CPU (`pcpu`)、およびコア ID (`cid`) に関する情報が表示されます。
 - `crypto` - 暗号化ユニットの出力には、モジュラー演算ユニット (Modular Arithmetic Unit、`mau`)、およびサポートされているその他の暗号化ユニット (Control Word Queue、`CWQ`) などが表示されます。
 - `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`) が表示されます。
 - `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 (UUID)、ソフトウェアの状態、利用率 (%)、正規化された利用率 (%)、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表示されます。
 - `hba` – 出力には、仮想 SCSI HBA、仮想 SAN (`vSAN`)、および仮想 SAN のドメインが表示されます。

- `memory` – 出力には、`memory` が表示されます。
- `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`) デバイスが表示されます。
- `physio` - 物理 I/O 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`) が表示されます。
- `resmgmt` – 出力には、リソース管理ポリシー情報が表示され、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。
- `san` – 出力には、仮想 SAN の名前、およびその仮想 SAN に関連付けられた SCSI HBA イニシエータポートのデバイスパスが表示されます。
- `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`) が表示されます。
- `status` - 出力には、移行中のドメインとメモリー動的再構成処理のステータスが表示されます。
 -o `status` オプションを使用すると、進行中の移行操作または DR 操作のステータスが表示されます。この情報は、`FLAGS` フィールドのフラグから入手できます。-o `status` オプションは `STATE` フィールドとは関係ありません。
- -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。
- -s は、CPU 関連およびメモリー関連のリソースに関するステータス情報を生成します。ステータスの値は、リソースが正常に動作していることを示す `ok` と、リソースに障害が発生していることを示す `fail` です。
 このステータスは、Fujitsu M10 プラットフォーム上の CPU およびメモリーリソースに対してのみ決定されます。その他のすべてのプラットフォームでは、ステータスフィールドは、-p オプションが使用されているときに解析可能な出力で表示されるだけです。これらのプラットフォーム上のステータスは、常に `status=NA` と表示されます。
- `domain-name` は、状態情報を一覧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

`list-bindings` サブコマンドは、論理ドメインのバインドを一覧表示します。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。

ドメインの名前を指定した場合、制御ドメインの MAC アドレスのあとに、仮想ネットワークデバイスの代替 MAC アドレスが表示されます。次のコマンドは、`ldg1` ドメイン上の `vnet1` の代替 MAC アドレスを 3 つ表示します。

```
primary# ldm list-bindings ldg1
...
NETWORK
NAME SERVICE ID DEVICE MAC MODE PVID VID MTU LINKPROP
vnet1 primary-vsw@primary 0 network@0 00:14:4f:f8:0c:80 1 1500
00:14:4f:fa:3a:f9
00:14:4f:f9:06:ab
00:14:4f:fb:3d:af
```

```

PEER          MAC          MODE PVID VID MTU LINKPROP
primary-vsw0@primary 00:14:4f:fa:94:60 1 1500
vnet2@ldg2    00:14:4f:f9:38:d1 1 1500
vnet3@ldg3    00:14:4f:fa:60:27 1 1500
vnet4@ldg4    00:14:4f:f8:0f:41 1 1500
...

```

次のコマンドは、ldg1 ドメイン上の vnet1 の代替 MAC アドレスを解析可能な出力で 3 つ表示します。

```

primary# ldm list-bindings -p ldg1
...
VNET|name=vnet1|dev=network@0|service=primary-vsw0@primary|mac-addr=00:14:4f:f8:0c:80
|mode=|pvid=1|vid=|mtu=1500|linkprop=|id=0
|alt-mac-addr=00:14:4f:fa:3a:f9,00:14:4f:f9:06:ab,00:14:4f:fb:3d:af
|peer=primary-vsw0@primary|mac-addr=00:14:4f:fa:94:60|mode=|pvid=1|vid=|mtu=1500
|peer=vnet2@ldg2|mac-addr=00:14:4f:f9:38:d1|mode=|pvid=1|vid=|mtu=1500|linkprop=
|peer=vnet3@ldg3|mac-addr=00:14:4f:fa:60:27|mode=|pvid=1|vid=|mtu=1500|linkprop=
|peer=vnet4@ldg4|mac-addr=00:14:4f:f8:0f:41|mode=|pvid=1|vid=|mtu=1500|linkprop=
...

```

ldm list-bindings コマンドは、mpgroup ディスクに関する次の情報を表示します。

- STATE 列には、各 mpgroup パスの次のいずれかの状態が表示されます。
 - active は、mpgroup の現在のアクティブなパスを示します
 - standby は、このパスが現在使用されていないことを示します
 - unknown は、ディスクが接続されていないか、または状態変更の最中であること、あるいは指定されたドメインが動的なパス選択をサポートする OS を実行していないことを示します
- パスのリストは、ドライバがアクティブなパスを選択するために使用する順序で表示されます (最初に表示されているパスが最初に選択されます)
- ディスクに関連付けられたボリュームは選択された mpgroup パスであるため、最初に表示されます。

この例では、選択されたパスは vol-ldg2@opath-ldg2 です。ldm list-bindings の出力は、アクティブなパスが選択されたパスではなく、vol-ldg1@opath-vdsであることを示しています。この状況は、選択されたパスが何らかの理由で失敗し、ドライバがリストにある 2 番目のパスをアクティブとして選択した場合に発生する可能性があります。選択されたパスがあとで使用可能になった場合でも、ドライバによって選択されたパスがアクティブなパスとして保持されます。最初のパスを再度アクティブにするには、ldm set-vdisk コマンドを再発行して、volume プロパティに必要なパスの名前 vol-ldg1@opath-vds を設定します。

```

primary# ldm list-bindings
DISK
NAME          VOLUME          TOUT ID DEVICE SERVER MPGROUP
disk          disk-ldg4@primary-vds0 0 disk@0 primary
tdiskgroup   vol-ldg2@opath-ldg2 1 disk@1 ldg2 testdiskgroup
PORT MPGROUP VOLUME          MPGROUP SERVER STATE
2    vol-ldg2@opath-ldg2 ldg2 standby
0    vol-ldg1@opath-vds ldg1 active
1    vol-prim@primary-vds0 primary standby

```

構文:

```
ldm list-bindings [-e] [-o [network|net]] [-p] [domain-name...]
```

ここでは:

- `-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。
- `-o [network|net]` は仮想スイッチおよび仮想ネットワークを含む仮想ネットワーク構成の出力を生成します。
- `-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。
- `domain-name` 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

`list-services` 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を一覧表示します。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。

構文:

```
ldm list-services [-e] [-p] [domain-name...]
```

ここでは:

- `-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。
- `domain-name` 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

`list-constraints` 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を一覧表示します。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。

復旧モードの操作によって物理ドメインから退避されたリソースはすべて、そのリソース識別子の前にアスタリスク (*) が付きます。

構文:

```
ldm list-constraints [-x] [domain-name...]
```

```
ldm list-constraints [-e] [-p] [domain-name...]
```

ここでは:

- `-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。
- `domain-name` は、制約を一覧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。

CPU コアアクティベーション情報の一覧表示

`list-permits` サブコマンドは、Fujitsu M10 プラットフォーム上の CPU コアアクティベーションの情報を一覧表示します。PERMITS 列には、発行された CPU コアアクティベーションの合計数が表示されます。この合計には、すべての常時 CPU コアアクティベーションおよび従量課金ベースの CPU コアアクティベーションが含まれます。常時 CPU コアアクティベーションとは、無制限に使用できるリソースの許可です。従量課金ベースの CPU コアアクティベーションとは、限られた時間だけ使用できるリソースの許可です。発行された常時 CPU コアアクティベーションの数は、PERMANENT 列に表示されます。IN USE 列には、使用中の発行済み CPU コアアクティベーションの数が表示されます。REST 列には、使用可能な CPU コアアクティベーションの数が表示されます。

構文:

```
ldm list-permits
```

デバイスの一覧表示

`list-devices` サブコマンドは、使用していない (バインドされていない) リソースまたはすべてのサーバーリソースのどちらかを一覧表示します。デフォルトでは、使用していないすべてのリソースを表示します。

構文:

```
ldm list-devices [-a] [-p] [-S] [cmi] [core] [cpu] [crypto] [io] [memory]
```

ここでは:

- `-a` は、すべてのサーバーリソース (バインドされたリソースおよびバインドされていないリソース) 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。
- `-S` は、CPU 関連およびメモリー関連のリソースに関するステータス情報を生成します。ステータスの値は、リソースが正常に動作していることを示す `ok` と、リソースに障害が発生していることを示す `fail` です。

このステータスは、Fujitsu M10 プラットフォーム上の CPU およびメモリーリソースに対してのみ決定されます。その他のすべてのプラットフォームでは、ステータスフィールドは、`-p` オプションが使用されているときに解析可能な出力で表示されるだけです。これらのプラットフォーム上のステータスは、常に `status=NA` と表示されます。

- `cmi` は、CMI デバイスに関する情報を一覧表示します。これには、これらのデバイスに関連付けられた共有メモリーや、割り当てられていないすべての仮想 CPU およびコアが含まれます。
- `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、CPU スレッドおよび物理 CPU リソースを一覧表示します。
- `crypto` は、モジュラー演算ユニットリソースのみを表示します。
- `memory` は、メモリーリソースのみを表示します。
- `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、入出力リソースのみを表示します。

リソース ID の番号には空きがあることがあります。次の例は、コア 2 が使用できないか、または無効になっている可能性があることを示しています。

```
primary# ldm list-devices -a core
CORE
ID      %FREE  CPUSSET
0       0      (0, 1, 2, 3, 4, 5, 6, 7)
1       100    (8, 9, 10, 11, 12, 13, 14, 15)
3       100    (24, 25, 26, 27, 28, 29, 30, 31)
4       100    (32, 33, 34, 35, 36, 37, 38, 39)
5       100    (40, 41, 42, 43, 44, 45, 46, 47)
6       100    (48, 49, 50, 51, 52, 53, 54, 55)
```

I/O デバイスの一覧表示

`list-io` サブコマンドは、システム上で構成されている I/O デバイスを一覧表示します。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一覧表示され、PSEUDONYM 列には、バスまたはネットワークデバイスに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一覧表示され、PSEUDONYM 列には、デバイスに関連付けられた仮名が表示されます。

STATUS 列は、差し込み式カードを受け入れるスロットおよびマザーボード上のデバイスに適用され、次のいずれかの値を指定できます。

- UNK – スロットのデバイスはファームウェアによって検出されましたが、OS では検出されませんでした。
- OCC – デバイスはマザーボード上で検出されたか、またはスロットの PCIe カードです。
- IOV – バスは IOV リソースを共有するために初期化されました。
- INV – スロット、仮想機能、または物理機能は無効な状態であり、使用できません。
- EMP – スロットは空です。

システムボード上のデバイスを示すスロットのステータスは常に、OCC になります。ルートドメインがダイレクト I/O をサポートしない場合、スロットのステータスは UNK になります。

構文:

```
ldm list-io [-l] [-p] [bus | device | pf-name]
```

```
ldm list-io -d pf-name
```

復旧モードの操作によって物理ドメインから退避されたリソースはすべて、そのリソース識別子の前にアスタリスク (*) が付きます。

ここでは:

- -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。
- -d *pf-name* には、指定された物理機能に関する情報が一覧表示されます。
- *bus*、*device*、および *pf-name* は、それぞれ PCIe バス、ダイレクト I/O 割り当て可能デバイス、および PCIe SR-IOV 物理機能です。

論理ドメイン構成の一覧表示

`list-spconfig` サブコマンドは、SP 上に格納されている論理ドメイン構成を一覧表示します。

構文:

```
ldm list-spconfig [-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とも示されません。

Note - 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 list-spconfig -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

`list-variable` サブコマンドは、論理ドメインの 1 つ以上の変数を一覧表示します。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。

構文:

```
ldm list-variable [var-name...] domain-name
```

ここでは:

- *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。

-
- *domain-name* は、1 つ以上の変数を一覧表示する論理ドメインの名前です。

物理 SCSI ホストバスアダプタの一覧表示

`list-hba` サブコマンドは、指定されたドメイン内の物理 SCSI HBA イニシエータポートを一覧表示します。論理ドメインの SCSI HBA イニシエータポートを確認したあと、`ldm add-vsan` コマンドを使用して、イニシエータポートの名前を指定することで仮想 SAN を作成できます。

```
ldm list-hba [-d] [-l] [-p] [-t] domain-name
```

ここでは:

- `-d` は、各イニシエータポートの配下にある SCSI デバイスを表示します。
- `-l` は、詳細な出力を表示します。
- `-p` は、解析可能な形式で出力を表示します。
- `-t` は、SCSI 転送メディアタイプを表示します (FiberChannel など)。
- *domain-name* は、指定する SCSI HBA イニシエータポートが存在する論理ドメインを指定します。

ネットワークデバイスの一覧表示

`list-netdev` サブコマンドは、システム上で構成されているネットワークデバイスを一覧表示します。デバイスに関して表示される情報には次のものがあります。

- CLASS – 次のネットワークデバイスタイプのいずれか
 - AGGR – ネットワークアグリゲーション
 - EOIB – Ethernet over InfiniBand
 - ESTUB – Ethernet スタブ
 - IPMP – IP ネットワークマルチパスグループ
 - PART – InfiniBand パーティション
 - PHYS – 物理ネットワークデバイス
 - VLAN – 仮想ローカルエリアネットワーク
 - VNET – 仮想ネットワークデバイス
 - VNIC – 仮想ネットワークインタフェースカード
 - VSW – 仮想スイッチデバイス
 - VXLAN – 仮想拡張 LAN
- MEDIA – ネットワークメディアのタイプ (Ethernet の場合は ETHER、InfiniBand の場合は IB)
- STATE – ネットワークデバイスの状態 (up、down、または unknown)
- SPEED – ネットワークデバイスの速度 (メガビット/秒)
- OVER – ネットワークデバイスがマップされる物理デバイス
- LOC – ネットワークデバイスの場所

構文:

```
ldm list-netdev [-b] [-l] [-p] [-o net-device] [domain-name]
```

ここでは:

- -b は、有効な仮想スイッチバックエンドデバイスのみを一覧表示できます。
- -l は、ネットワークデバイス、仮想スイッチデバイス、およびアグリゲーションに関する情報を一覧表示します。
- -p は、解析可能でマシンが読み取り可能な形式で出力を書き込みます。
- -o *net-device* は、指定されたネットワークデバイスに関する情報を一覧表示します。
- *domain-name* は、ネットワークデバイス情報を一覧表示する論理ドメインを指定します。

ネットワークデバイス統計情報の一覧表示

`list-netstat` サブコマンドは、システム上で構成されているネットワークデバイスの統計情報を一覧表示します。統計情報は、次のフィールドに表示されます。

- IPACKETS はインバウンドパケットを示します
- OPACKETS はアウトバウンドパケットを示します
- RBYTES は、受け取った (インバウンドの) バイト数を示します
- OBYTES は送信された (アウトバウンド) バイト数を示します

構文:

```
ldm list-netstat [-p] [-u unit] [-o net-device] [-t interval [-c count]] [domain-name]
```

ここでは:

- -c *count* は、統計情報を報告する回数を指定します。0 の値を指定すると、統計情報を無制限に報告します。-c を指定する場合は、-t で時間間隔を指定する必要があります。
- -o *net-device* は、指定されたネットワークデバイスに関する情報を一覧表示します。
- -p は、解析可能でマシンが読み取り可能な形式で出力を書き込みます。
- -t *interval* は、統計情報がリフレッシュされる間隔を秒単位で指定します。デフォルト値は 1 秒です。
- -u *unit* は、出力を表示する単位を指定します。有効な値は次のとおりです。
 - r は生のバイト数を指定します
 - k は K バイトを指定します
 - m は M バイトを指定します
 - g は G バイトを指定します
- *domain-name* は、ネットワークデバイス情報を一覧表示する論理ドメインを指定します。

依存関係の一覧表示

`list-dependencies` サブコマンドは、ドメイン内の依存関係を一覧表示します。オプションを指定しない場合、このコマンドはドメインと、そのドメインが依存するドメインのリストを出力します。

構文:

```
ldm list-dependencies [-l] [-p] [-r] [domain-name]
```

ここでは:

- `-l` は、依存関係に関する詳細情報を一覧表示します。
- `-p` は、解析可能でマシンが読み取り可能な形式で出力を書き込みます。
- `-r` は、依存関係ごとにグループ化された依存要素を表示します。
- `domain-name` は、依存関係情報を一覧表示する論理ドメインを指定します。`domain-name` が指定されていない場合は、すべてのドメインの依存関係情報が一覧表示されます。

リソースグループの一覧表示

`list-rsrc-group` サブコマンドは、リソースグループに関する情報を表示します。オプションを指定しない場合、このコマンドは、システム内のすべてのリソースグループの短いリストを生成します。

構文:

```
ldm list-rsrc-group [-a] [-d domain-name] [-l] [-o core|memory|io] [-p] [resource-group]
```

ここでは:

- `-a` は、各リソースグループのすべてのリソースに関する情報を一覧表示します。この出力には、どのドメインにもバインドされていないリソースが含まれます。
- `-d domain-name` は、指定されたドメインに関する情報のみを表示します。
- `-l` は、各リソースグループに関する詳細情報を一覧表示します。
- `-o core|memory|io` は、指定されたリソースタイプ(コア、メモリー、または I/O)に関する情報のみを一覧表示します。
- `-p` は、解析可能でマシンが読み取り可能な形式で出力を書き込みます。
- `resource-group` は、リソースグループを指定します。

CMI デバイスの一覧表示

`list-cmi` サブコマンドは、Fujitsu M10 サーバー上で構成されている CMI デバイスに関する情報を表示します。システム上で CMI デバイスが構成されていない場合、出力は表示されません。

出力は次のセクションに分かれます。

- CMI デバイスとそれに関連付けられた仮想 CPU およびコアの割り当て。

-
- **バインド済み。** CMI デバイスとそれに関連付けられた仮想 CPU およびコアがバインドされているドメインを表示します。これらのドメインは、`grow-cmi` および `shrink-cmi` サブコマンドのターゲットです。
 - **テナント。** バインドされた CMI デバイスとそれに関連付けられた仮想 CPU およびコアを所有していないドメインを表示します。これらのドメインは、`evict-cmi` サブコマンドのターゲットです。
 - **空き。** 割り当てられていない CMI デバイスと、これらのデバイスに関連付けられた割り当てられていないすべての仮想 CPU またはコアを表示します。
 - メッセージキュー情報。
 - 共有メモリー情報。

-l オプションを指定した場合は、CMI デバイスに関連付けられた仮想 CPU およびコアに関する追加情報が表示されます。さらに、*domain-name* が指定されていないかぎり、各 CMI デバイスに関連付けられたすべての仮想 CPU およびコアに関する情報が出力の前に付加されます。

構文:

```
ldm list-cmi [-l] [-p] [cmi_id=ID[,ID[,...]]] [domain-name...]
```

ここでは:

- -l は、物理 CPU セットとコア ID を一覧表示します。
- -p は、解析可能でマシンが読み取り可能な形式で出力を書き込みます。
- `cmi_id=ID[,...]` は、情報を一覧表示する 1 つ以上の CMI デバイスを指定します。
- *domain-name* は、情報を一覧表示する 1 つ以上の論理ドメインを指定します。

CPU ソケットの一覧表示

`list-socket` サブコマンドは、Fujitsu M10 サーバー上の CPU ソケットに関する情報を表示します。

出力は次のセクションに分かれます。

- 論理ドメインに対して指定された CPU ソケット制約。
- 各 CPU ソケットの仮想 CPU およびコアの割り当て。
 - **テナント。** 論理ドメインと、これらのドメインにバインドされている仮想 CPU およびコアを表示します。
 - **空き。** 割り当てられていないすべての仮想 CPU またはコアを表示します。
- 各 CPU ソケットのメモリーの割り当て。
- 各 CPU ソケットの I/O バスの割り当て。

-l オプションを指定した場合は、仮想 CPU およびコアに関する追加情報が表示されます。さらに、*domain-name* が指定されていないかぎり、各 CPU ソケット内のすべての仮想 CPU およびコアに関する情報が出力の前に付加されます。

構文:

```
ldm list-socket [--free] [-l] [-o format] [-p] [socket_id=ID[,ID[,...]]] [domain-name...]
```

ここでは:

- `--free` は、空きリソースのみを一覧表示します。
- `-l` は、物理 CPU セットとコア ID を一覧表示します。
- `-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。
 - `raw` – 出力には、CPU ソケット内のすべての物理 CPU およびコアの物理 CPU セットおよびコア ID に関する情報が表示されます。*domain-name* が指定されている場合、出力は表示されません。
 - `constraint` – 出力には、CPU ソケット制約が表示されます。`--free` オプションが指定されている場合、出力は表示されません。
 - `cpu` – 出力には、仮想 CPU およびコアに関する情報が表示されます。
 - `memory` – 出力には、メモリーに関する情報が表示されます。
 - `physio` – 出力には、I/O バスに関する情報が表示されます。
- `-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。
- `socket_id=ID[,...]` は、情報を一覧表示する 1 つ以上の CPU ソケットを指定します。
- *domain-name* は、情報を一覧表示する 1 つ以上の論理ドメインを指定します。

リソース管理ポリシーの追加、設定、および削除

リソース管理ポリシーの追加

`add-policy` サブコマンドでは、1 つ以上の論理ドメインのリソース管理ポリシーを追加できます。リソース管理ポリシーは、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。

構文:

```
ldm add-policy [enable=yes|no] [priority=value] [attack=value] [decay=value]
[elastic-margin=value] [sample-rate=value] [tod-begin=hh:mm[:ss]] [tod-end=hh:mm[:ss]]
[util-lower=percent] [util-upper=percent] [vcpu-min=value] [vcpu-max=value]
name=policy-name domain-name...
```

ここでは:

- `attack=value` には、いずれか 1 つのリソース制御サイクル中に追加するリソースの最大数を指定します。使用可能なリソースの数が指定された値よりも少ない場合、使用可能なリソースがすべて追加されます。使用可能な CPU スレッドをできるだけ多く追加できるように、デフォルトでは `attack` に制限はありません。ドメインにコア全体の制約が設定されている場合、値はコア全体 (通常 8) の倍数になる必要があります。有効な値は、1 から、システムで使用していない CPU スレッドの数から 1 を引いた数までです。

- `decay=value` には、いずれか 1 つのリソース制御サイクル中に削除するリソースの最大数を指定します。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 CPU スレッドの数から `vcpu-min` の値を引いた数だけが削除できます。デフォルトでは、値は無限です。ドメインにコア全体の制約が設定されている場合、値はコア全体 (通常 8) の倍数になる必要があります。有効な値は、1 から、システムの CPU スレッドの合計数から 1 を引いた数までです。
- `elastic-margin=value` では、`util-lower` と使用していない CPU スレッド数の間のバッファ領域量を指定して、CPU スレッドの数を減らした場合の変動を回避します。この値は、`util-upper` よりも大きくできません。有効な値は、0 から 100 までです。ドメインにコア全体の制約が設定されている場合、デフォルト値は 15 です。ドメインにコア全体の制約が設定されていない場合、デフォルト値は 5 です。
- `enable=yes|no` は、個々のドメインのリソース管理を有効または無効にします。デフォルトでは、`enable=yes` です。
- `name=policy-name` は、リソース管理ポリシー名を指定します。
- `priority=value` は、Dynamic Resource Management (DRM) ポリシーの優先順位を指定します。優先順位の値は、単一ドメイン内の DRM ポリシー間の関係、および単一システム内の DRM 対応ドメイン間の関係を決定するために使用されます。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。

`priority` プロパティの動作は、次に示すように、使用していない CPU リソースプールが使用可能であるかどうかによって異なります。

- **プール内の空き CPU リソースが使用可能です。** この場合、`priority` プロパティによって、1 つのドメインに複数の重複するポリシーが定義されている場合に、どの DRM ポリシーが有効になるのかが決まります。
- **プール内の空き CPU リソースが使用可能ではありません。** この場合、`priority` プロパティによって、同じシステム内の優先順位の低いドメインから優先順位の高いドメインにリソースを動的に移動できるかどうか指定されます。ドメインの優先順位は、そのドメインで有効になっている DRM ポリシーによって指定された優先順位です。
たとえば、優先順位の高いドメインは、優先順位の低い DRM ポリシーを持つ別のドメインから CPU リソースを取得できます。このリソース取得機能は、DRM ポリシーが有効になっているドメインのみに適用されます。`priority` 値が等しいドメインは、この機能の影響を受けません。そのため、すべてのポリシーに対してデフォルトの優先順位が使用されている場合、ドメインは優先順位の低いドメインからリソースを取得できません。この機能を活用するには、値が等しくならないように `priority` プロパティの値を調整します。
- `sample-rate=value` は、DRM のサンプリングレートとなるサイクル時間を秒で指定します。有効な値の範囲は、1 - 9999 です。推奨値はデフォルトの 10 です。
- `tod-begin=hh:mm[:ss]` は、ポリシーの有効開始時間を時、分、およびオプションの秒で指定します。この時間は、午前 0 時から 23:59:59 の範囲で、`tod-end` で指定された時間よりも早い時間に設定する必要があります。デフォルト値は 00:00:00 です。

- `tod-end=hh:mm[:ss]` は、ポリシーの有効停止時間を時、分、およびオプションの秒で指定します。この時間は、午前 0 時から 23:59:59 の範囲で、`tod-begin` で指定された時間よりも遅い時間に設定する必要があります。デフォルト値は 23:59:59 です。
- `util-lower=percent` は、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、`util-upper` より 1 少ない数までです。デフォルト値は 30 です。
- `util-upper=percent` は、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、`util-lower` に 1 を足した数から 99 までです。デフォルト値は 70 です。
- `vcpu-max=value` は、ドメインの CPU スレッドリソースの最大数を指定します。デフォルトでは、CPU スレッドの最大数に制限はありません。ドメインにコア全体の制約が設定されている場合、値はコア全体 (通常 8) の倍数になる必要があります。有効な値は、`vcpu-min` に 1 を足した数から、システムで使用していない CPU スレッドの合計数までです。
- `vcpu-min=value` は、ドメインの CPU スレッドリソースの最小数を指定します。ドメインにコア全体の制約が設定されている場合、値はコア全体 (通常 8) の倍数になる必要があります。有効な値は、1 から、`vcpu-max` より 1 少ない数までです。デフォルト値は 1 です。
- `domain-name` は、リソース管理ポリシーを追加する論理ドメインを指定します。

リソース管理ポリシーの変更

`set-policy` サブコマンドでは、オプションプロパティの値を指定することによって、1 つ以上の論理ドメインのリソース管理ポリシーを変更できます。

構文:

```
ldm set-policy [enable=[yes|no]] [priority=[value]] [attack=[value]] [decay=[value]]
[elastic-margin=[value]] [sample-rate=[value]] [tod-begin=[hh:mm:ss]]
[tod-end=[hh:mm:ss]] [util-lower=[percent]] [util-upper=[percent]] [vcpu-min=[value]]
[vcpu-max=[value]] name=policy-name domain-name...
```

ここでは:

- プロパティは、「リソース管理ポリシーの追加」セクションで説明します。
- `domain-name` は、リソース管理ポリシーを変更する論理ドメインを指定します。

リソース管理ポリシーの削除

`remove-policy` サブコマンドでは、1 つ以上のポリシー名を指定することによって、論理ドメインからリソース管理ポリシーを削除できます。

構文:

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

ここでは:

- `name` プロパティは、リソース管理ポリシーの名前である `policy-name` を指定します。
- `domain-name` は、リソース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

`init-system` 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 list-constraints -x` の出力など) を入力として取り込み、指定されたドメインを構成して、制御ドメインをリブートします。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。

構文:

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

ここでは:

- `-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。
- `-f` は、出荷時のデフォルト構成チェックをスキップし、システム上で何が構成済みであるかを問わず処理を次に進めます。
-f オプションの使用時には注意が必要です。 `ldm init-system` はシステムが出荷時のデフォルト構成であると想定するため、XML ファイルで指定された変更を直接適用します。システムが出荷時のデフォルト以外の構成の場合に `-f` を使用すると、システムが XML ファイルで指定された構成どおりにならない可能性が高くなります。XML ファイル内の変更と初期構成の組み合わせによっては、システムに1つまたは複数の変更が適用されない場合があります。
- `-r` は構成後にシステムをリブートします。
- `-s` は、仮想サービス構成 (`vds`、`vcc`、および `vsw`) のみを復元します。

ハイパーバイザダンプデータの収集

ハイパーバイザダンプデータ収集サブコマンドは、Fujitsu M10 プラットフォーム上のハイパーバイザダンプからデータを収集するプロセスにのみ適用されます。

ハイパーバイザの強制的な中止イベントが発生すると、ハイパーバイザメモリーの内容がファームウェアによって保持され、システムは工場出荷時のデフォルト構成でリブートされます。`ldmd` デーモンは、保持されたハイパーバイザメモリーの内容を、制御ドメイン上の `/var/opt/SUNWldm/hvdump.N.gz` というファイルにコピーします。`N` は、0-7 の範囲内の数値です。このファイルは、ハイパーバイザの強制的な中止が発生した時点のハイパーバイザメモリーの内容のバイナリダンプです。

ハイパーバイザダンプデータの一覧表示

`list-hvdump` サブコマンドは、Fujitsu M10 プラットフォーム上で使用可能なハイパーバイザデータ収集プロセスを制御する `hvdump` および `hvdump-reboot` プロパティの値を表示します。

ldm list-hvdump

ハイパーバイザデータ収集プロセスのプロパティ値の設定

`set-hvdump` サブコマンドは、Fujitsu M10 ハイパーバイザデータ収集のプロパティを変更します。自動ハイパーバイザデータ収集プロセスを有効または無効にするプロパティを設定できます。データの収集後に元の構成を復元するための自動リブートを有効または無効にするプロパティを設定することもできます。

構文:

```
ldm set-hvdump [hvdump=on|off] [hvdump-reboot=on|off]
```

ここでは:

- `hvdump=on|off` は、ハイパーバイザデータ収集プロセスを有効または無効にします。デフォルト値は `on` です。
- `hvdump-reboot=on|off` は、ハイパーバイザデータ収集プロセスが完了したあとの自動システムリブートを有効または無効にします。デフォルト値は `off` です。

手動によるハイパーバイザデータ収集プロセスの起動

`start-hvdump` サブコマンドは、自動収集に失敗した場合に、Fujitsu M10 ハイパーバイザデータ収集プロセスを手動で起動します。

ldm start-hvdump

CMI 操作の実行

CMI 関連のサブコマンドは、Fujitsu M10 プラットフォームにのみ適用されます。

CMI 関連のすべてのサブコマンドでは、CMI デバイスの数を指定すると、割り当てまたは削除の対象となる CMI リソースが自動的に選択されます。CMI リソースの明示的な割り当てまたは削除を行うには、`cmi_id` プロパティに CMI ID 値を指定します。

CMI 関連の操作を実行するとき、ドメインはアクティブでないドメインであるか、または `primary` ドメインである場合は遅延再構成モードになっている必要があります。

CMI デバイスの追加

`add-cmi` サブコマンドは、指定された数の CMI デバイスをドメインに追加します。

CMI 制約を使用してアクティブでないドメインをバインドすると、その CMI 制約および使用可能なリソースから未指定の仮想 CPU、コア、およびメモリ制約が自動的に生成されます。

構文:

```
ldm add-cmi num domain-name
```

```
ldm add-cmi cmi_id=ID[,ID[,...]] domain-name
```

ここでは:

- *num* は、ドメインに割り当てる CMI リソースの数を指定します。
- *cmi_id=ID[,...]* は、ドメインに追加する 1 つ以上の CMI デバイスを指定します。
- *domain-name* は、CMI デバイスを割り当てる 1 つ以上の論理ドメインを指定します。

CMI デバイスの変更

`set-cmi` サブコマンドは、ドメインに割り当てる CMI デバイスの数を指定します。

`-f` オプションを使用すると、既存の仮想 CPU、コア、およびメモリー制約をすべてクリアできます。CMI 制約を使用してアクティブでないドメインをバインドすると、その CMI 制約および使用可能なリソースからこれらの制約が自動的に生成されます。

構文:

```
ldm set-cmi [-f] num domain-name
```

```
ldm set-cmi [-f] cmi_id=[ID[,ID[,...]]] domain-name
```

ここでは:

- `-f` は、既存の仮想 CPU、コア、およびメモリー制約をすべてクリアします。
- *num* は、ドメインに割り当てる CMI リソースの数を指定します。
- *cmi_id=ID[,...]* は、ドメインに追加する 1 つ以上の CMI デバイスを指定します。*cmi_id=* は、指定された CMI デバイスをすべて削除します。
- *domain-name* は、CMI デバイスが割り当てられるドメインを指定します。

CMI デバイスの削除

`remove-cmi` サブコマンドは、ドメインから削除する CMI デバイスの数を指定します。

構文:

```
ldm remove-cmi [-f] num domain-name
```

```
ldm remove-cmi [-f] cmi_id=ID[,ID[,...]] domain-name
```

ここでは:

- *num* は、ドメインから削除する CMI リソースの数を指定します。
- *cmi_id=ID[,...]* は、ドメインから削除する 1 つ以上の CMI デバイスを指定します。
- *domain-name* は、CMI デバイスが削除されるドメインを指定します。

CMI デバイスの CPU スレッドまたは CPU コアの追加

`grow-cmi` サブコマンドでは、特定の CMI デバイスに関連付けられた仮想 CPU またはコアを、1 つ以上の CMI リソースを含むドメインに追加できます。指定された CMI

デバイスはそのドメインに割り当てられている必要があります、またそのドメインはバインドされているか、またはアクティブである必要があります。

構文:

```
ldm grow-cmi vcpus=num cmi_id=ID domain-name
```

```
ldm grow-cmi cores=num cmi_id=ID domain-name
```

ここでは:

- `vcpus=num` は、ドメインに追加する仮想 CPU の数を指定します。
- `cores=num` は、ドメインに追加するコアの数を指定します。
- `cmi_id=ID` は、ドメインによって所有されている CMI デバイスを指定します。
- `domain-name` は、仮想 CPU またはコアが追加されるドメインを指定します。

CMI デバイスの CPU スレッドまたは CPU コアの削除

`shrink-cmi` サブコマンドでは、1 つ以上の CMI リソースを含むドメインから特定の CMI デバイスに関連付けられた仮想 CPU またはコアを削除できます。指定された CMI デバイスはそのドメインに割り当てられている必要があります、またそのドメインはバインドされているか、またはアクティブである必要があります。

構文:

```
ldm shrink-cmi vcpus=num cmi_id=ID domain-name
```

```
ldm shrink-cmi cores=num cmi_id=ID domain-name
```

ここでは:

- `vcpus=num` は、ドメインから削除する仮想 CPU の数を指定します。
- `cores=num` は、ドメインから削除するコアの数を指定します。
- `cmi_id=ID` は、ドメインによって所有されている CMI デバイスを指定します。
- `domain-name` は、仮想 CPU またはコアが削除されるドメインを指定します。

CMI デバイスの CPU スレッドまたは CPU コアの削除

`evict-cmi` サブコマンドでは、CMI デバイスが割り当てられていないバインドされているか、またはアクティブなドメインから、特定の CMI デバイスに関連付けられた仮想 CPU またはコアを削除できます。

CMI デバイスとそれに関連付けられた仮想 CPU およびコアの割り当てを確認するには、`list-cmi` サブコマンドを実行します。

構文:

```
ldm evict-cmi vcpus=num cmi_id=ID domain-name
```

```
ldm evict-cmi cores=num cmi_id=ID domain-name
```

ここでは:

- `vcpus=num` は、ドメインから削除する仮想 CPU の数を指定します。

- `cores=num` は、ドメインから削除するコアの数を指定します。
- `cmi_id=ID` は、CMI デバイスを指定します。
- `domain-name` は、仮想 CPU またはコアが削除されるドメインを指定します。

CPU ソケット操作の実行

CPU ソケット関連のコマンドは、Fujitsu M10 プラットフォームにのみ適用されます。
CPU ソケットのスレッド、コア、またはメモリーの追加

`grow-socket` サブコマンドでは、特定の CPU ソケットに関連付けられた仮想 CPU、コア、またはメモリーをドメインに追加できます。そのドメインはバインドされているか、またはアクティブである必要があります。

ドメインに CMI デバイスが割り当てられている場合は、`grow-cmi` サブコマンドを使用すると、仮想 CPU またはコアをそのドメインに追加できます。

構文:

```
ldm grow-socket vcpus=num socket_id=ID domain-name
ldm grow-socket cores=num socket_id=ID domain-name
ldm grow-socket memory=size[unit] socket_id=ID domain-name
```

ここでは:

- `vcpus=num` は、ドメインに追加する仮想 CPU の数を指定します。
- `cores=num` は、ドメインに追加するコアの数を指定します。
- `memory=num` は、ドメインに追加するメモリーの量を指定します。デフォルトの量は `size` (バイト単位) です。異なるサイズの単位が必要な場合は、`unit` を、大文字または小文字を使用して次のいずれかの値として指定します。
 - G バイトの場合は G
 - K バイトの場合は K
 - M バイトの場合は M
- `socket_id=ID` は、CPU ソケットを指定します。
- `domain-name` は、仮想 CPU、コア、またはメモリーが追加されるドメインを指定します。

CPU ソケット制約の指定

`set-socket` サブコマンドは、ドメインが仮想 CPU、コア、およびメモリーの割り当てを行うことができる CPU ソケットを指定します。

CPU ソケット制約を使用してアクティブでないドメインをバインドすると、仮想 CPU、コア、およびメモリーは、指定された CPU ソケットからのみ選択されます。ドメインに仮想 CPU、コア、またはメモリー制約が指定されていない場合、CPU ソケット制約および使用可能なリソースからリソース制約が自動的に生成されます。既存の

仮想 CPU、コア、およびメモリー制約をすべてクリアするには、`-f` オプションを使用します。

バインドされたドメインに CPU ソケット制約を指定すると、既存の仮想 CPU、コア、およびメモリーのバインドは、指定された CPU ソケット制約と整合するように更新されます。

アクティブなドメインに CPU ソケット制約を指定すると、アクティブな仮想 CPU リソースおよび物理メモリー範囲は、基となる物理リソースが指定の CPU ソケット制約と整合するように再マッピングされます。

物理リソースがシステムから削除されると、CPU ソケット制約が機能低下する可能性があります。物理リソースの復元後は、元の CPU ソケット制約を復元できます。

構文:

```
ldm set-socket [-f] [--remap] socket_id=[ID[,ID[,...]]] domain-name
```

```
ldm set-socket [-f] [--remap] --restore-degraded domain-name
```

ここでは:

- `-f` は、アクティブでないドメインの既存の仮想 CPU、コア、およびメモリー制約をすべてクリアします。ドメインがバインドされているかアクティブな場合、指定された CPU ソケットのリソースが制約よりも少ないと、リソース制約、機能低下した CPU ソケット制約、またはその両方が変更される場合があります。
- `--remap` は、ある物理リソースにバインドされているアクティブな仮想リソースを、その仮想リソースを所有するドメインの実行中に別の物理リソースに移動します。
- `--restore-degraded` は、物理リソースの追加後に、ドメインをその元の CPU ソケットに復元します。
- `socket_id=ID` は、ドメインが制約される 1 つ以上の CPU ソケットを指定します。`socket_id=` を指定すると、すべての CPU ソケット制約がドメインから削除されます。
- `domain-name` は、CPU ソケット制約を追加するドメインを指定します。

CPU ソケットのスレッド、コア、またはメモリーの削除

`shrink-socket` サブコマンドでは、特定の CPU ソケットに関連付けられた仮想 CPU、コア、またはメモリーをドメインから削除できます。そのドメインはバインドされているか、またはアクティブである必要があります。

ドメインに CMI デバイスが割り当てられている場合は、`shrink-cmi` サブコマンドを使用すると、仮想 CPU またはコアをそのドメインから削除できます。

構文:

```
ldm shrink-socket vcpus=num socket_id=ID domain-name
```

```
ldm shrink-socket cores=num socket_id=ID domain-name
```

```
ldm shrink-socket memory=size[unit] socket_id=ID domain-name
```

ここでは:

- `vcpus=num` は、ドメインから削除する仮想 CPU の数を指定します。
- `cores=num` は、ドメインから削除するコアの数を指定します。
- `memory=num` は、ドメインから削除するメモリーの量を指定します。デフォルトの量は `size` (バイト単位) です。異なるサイズの単位が必要な場合は、`unit` を、大文字または小文字を使用して次のいずれかの値として指定します。
 - G バイトの場合は G
 - K バイトの場合は K
 - M バイトの場合は M
- `socket_id=ID` は、CPU ソケットを指定します。
- `domain-name` は、仮想 CPU、コア、またはメモリーが削除されるドメインを指定します。

EXAMPLE 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。

```
primary# ldm add-vds primary-vds0 primary
primary# ldm add-vsw net-dev=net0 primary-vsw0 primary
primary# ldm add-vcc port-range=5000-5100 primary-vcc0 primary
```

EXAMPLE 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。

```
primary# ldm list-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 net0     switch@0  1      1
VDS
  NAME          LDOM    VOLUME      OPTIONS  MPGROUP    DEVICE
  primary-vds0  primary
```

EXAMPLE 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リソースを利用でき、それらのリソース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リソースのみを設定し、残りのリソース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。リポートして変更を有効にする必要があります。

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。

```
primary# ldm start-reconf primary
primary# ldm set-vcpu 8 primary
primary# ldm set-memory 8G primary
primary# ldm add-spconfig initial
primary# shutdown -y -g0 -i6
primary# svcadm enable vntsd
```

EXAMPLE 4 バインドの一覧表示

バインドのリストを表示して、指定したリソース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リソースを確認できます。

```
primary# ldm list-bindings primary
NAME      STATE      FLAGS  CONS  VCPU  MEMORY  UTIL  NORM  UPTIME
primary   active     -n-cv-  UART   8     16G     0.2%  0.2%  1d 18h 5m

UUID
d8d2db22-21b9-e5e6-d635-92036c711e65

MAC
00:21:28:c1:3f:3c

HOSTID
0x84c13f3c

CONTROL
failure-policy=ignore
extended-mapin-space=on
cpu-arch=native
rc-add-policy=
shutdown-group=0
perf-counters=global,htstrand

DEPENDENCY
master=

CORE
CID  CPUSSET
0    (0, 1, 2, 3, 4, 5, 6, 7)

VCPU
VID  PID  CID  UTIL  NORM  STRAND
0    0    0    0.4% 0.4% 100%
1    1    0    0.2% 0.2% 100%
2    2    0    0.1% 0.1% 100%
3    3    0    0.1% 0.1% 100%
4    4    0    0.2% 0.2% 100%
5    5    0    0.5% 0.5% 100%
6    6    0    0.2% 0.2% 100%
7    7    0    1.2% 1.2% 100%

MEMORY
RA          PA          SIZE
0x20000000 0x20000000 8G
0x40000000 0x40000000 8G
```

```

VARIABLES
pm_boot_policy=disabled=1;ttfc=0;ttmr=0;

IO
DEVICE                                PSEUDONYM      OPTIONS
pci@400                               pci_0
niu@480                               niu_0
pci@400/pci@1/pci@0/pci@8            /SYS/MB/RISER0/PCIE0
pci@400/pci@2/pci@0/pci@8            /SYS/MB/RISER1/PCIE1
pci@400/pci@1/pci@0/pci@6            /SYS/MB/RISER2/PCIE2
pci@400/pci@2/pci@0/pci@c            /SYS/MB/RISER0/PCIE3
pci@400/pci@1/pci@0/pci@0            /SYS/MB/RISER1/PCIE4
pci@400/pci@2/pci@0/pci@a            /SYS/MB/RISER2/PCIE5
pci@400/pci@1/pci@0/pci@4            /SYS/MB/SASHBA0
pci@400/pci@2/pci@0/pci@4            /SYS/MB/SASHBA1
pci@400/pci@2/pci@0/pci@6            /SYS/MB/NET0
pci@400/pci@2/pci@0/pci@7            /SYS/MB/NET2

VCC
NAME          PORT-RANGE
primary-vcc0  5000-5100

VSW
NAME          MAC          NET-DEV  ID  DEVICE  LINKPROP
primary-vsw0  00:14:4f:fa:0b:57 net0     0   switch@0

DEFAULT-VLAN-ID PVID VID          MTU  MODE INTER-VNET-LINK
1             1             1500  on

VDS
NAME          VOLUME          OPTIONS          MPGROUP          DEVICE
primary-vds0

VCONS
NAME          SERVICE          PORT  LOGGING
UART

```

EXAMPLE 5 ネットワーク関連バインドの一覧表示

ドメインの仮想ネットワーク構成に関する情報を取得するためにバインドを一覧表示できます。

```

primary# ldm list-bindings -o network ldg3
NAME
ldg3

MAC
00:14:4f:fb:7d:03

VSW
NAME          MAC          NET-DEV  DVID|PVID|VIDs
----          ---          -         -
vsw-ldg3     00:14:4f:fa:0b:57 -         1|1|--

NETWORK
NAME          SERVICE          MACADDRESS          PVID|PVLAN|VIDs
----          -
vnet3       primary-vsw0@primary 00:14:4f:fa:e2:a1  1|--|--

PEER          MACADDRESS          PVID|PVLAN|VIDs

```

```

-----
primary-vsw0@primary      00:14:4f:fb:e8:d8      1|--|--
NAME      SERVICE                MACADDRESS              PVID|PVLAN|VIDs
-----
vnet1     primary-vsw1@primary   00:14:4f:f8:48:e5      1|--|--

PEER
-----
primary-vsw1@primary     00:14:4f:fa:10:db      1|--|--
vnet2@ldg3               00:14:4f:fb:fe:ec      1|--|--
vnet4@ldg3               00:14:4f:f9:91:d0      1|--|--
vnet50@ldg3              00:14:4f:f8:71:10      1|--|--

```

EXAMPLE 6 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リソースの存在の確認、ゲストドメインの追加、ドメインに必要なリソース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、ドメインへのリソースのバインド、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。

```

primary# ldm list-devices
primary# ldm add-domain ldg1
primary# ldm add-vcpu 8 ldg1
primary# ldm add-memory 8g ldg1
primary# ldm add-vnet vnet1 primary-vsw0 ldg1
primary# ldm add-vdsdev /dev/dsk/c0t1d0s2 vol1@primary-vds0
primary# ldm add-vdisk vdisk1 vol1@primary-vds0 ldg1
primary# ldm set-variable auto-boot\?=false ldg1
primary# ldm set-variable boot-device=vdisk1 ldg1
primary# ldm bind-domain ldg1
primary# ldm list-constraints -x ldg1 > ldg1.xml
primary# ldm add-spconfig ldg1_8cpu_1G
primary# ldm start-domain ldg1
primary# ldm list -l ldg1
primary# telnet localhost 5000

```

EXAMPLE 7 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。

```

primary# ldm set-vcons group=ldg1 service=primary-vc0 ldg2

```

最初のゲストドメインを除く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 list -l コマンドを使用すると、すべてのドメインが同じポートに接続されていることを確認できます。コンソールの使用方法については、[vntsd\(1M\)](#) マニュアルページを参照してください。

EXAMPLE 8 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。

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

EXAMPLE 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。

```
primary# ldm cancel-operation reconf primary
```

EXAMPLE 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。

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

EXAMPLE 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。

```
primary# ldm list-spconfig
factory-default
3guests [current]
data1
reconfig_primary
split1
primary# ldm list-spconfig -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。この状態にある間にシステムが電源再投入を実行した場合、Logical Domains Manager は、指定されたポリシーに基づいて 3guests の自動保存を実行しま

す。3guests が current としてマークされているため、自動保存のアクション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。remove-spconfig コマンドを使用して構成を明示的に削除すると、制御ドメイン上の自動保存のバージョンも削除されることに注意してください。その結果、制御ドメインにも SP にも構成は残りません。

EXAMPLE 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。

```
primary# ldm list-io
NAME                                     TYPE   BUS      DOMAIN  STATUS
----
pci_0                                    BUS    pci_0    primary IOV
niu_0                                    NIU    niu_0    primary
/SYS/MB/RISER0/PCIE0                    PCIE   pci_0    primary EMP
/SYS/MB/RISER1/PCIE1                    PCIE   pci_0    primary EMP
/SYS/MB/RISER2/PCIE2                    PCIE   pci_0    primary EMP
/SYS/MB/RISER0/PCIE3                    PCIE   pci_0    primary OCC
/SYS/MB/RISER1/PCIE4                    PCIE   pci_0    primary OCC
/SYS/MB/RISER2/PCIE5                    PCIE   pci_0    primary EMP
/SYS/MB/SASHBA0                          PCIE   pci_0    primary OCC
/SYS/MB/SASHBA1                          PCIE   pci_0    primary OCC
/SYS/MB/NET0                             PCIE   pci_0    primary OCC
/SYS/MB/NET2                             PCIE   pci_0    primary OCC
/SYS/MB/RISER0/PCIE3/IOVIB.PF0          PF      pci_0    primary
/SYS/MB/RISER1/PCIE4/IOVIB.PF0          PF      pci_0    primary
/SYS/MB/NET0/IOVNET.PF0                 PF      pci_0    primary
/SYS/MB/NET0/IOVNET.PF1                 PF      pci_0    primary
/SYS/MB/NET2/IOVNET.PF0                 PF      pci_0    primary
/SYS/MB/NET2/IOVNET.PF1                 PF      pci_0    primary
/SYS/MB/RISER0/PCIE3/IOVIB.PF0.VF0      VF      pci_0    primary
/SYS/MB/RISER0/PCIE3/IOVIB.PF0.VF1      VF      pci_0    primary
/SYS/MB/RISER0/PCIE3/IOVIB.PF0.VF2      VF      pci_0    iodom1
/SYS/MB/RISER0/PCIE3/IOVIB.PF0.VF3      VF      pci_0    iodom1
/SYS/MB/RISER1/PCIE4/IOVIB.PF0.VF0      VF      pci_0    primary
/SYS/MB/RISER1/PCIE4/IOVIB.PF0.VF1      VF      pci_0    primary
/SYS/MB/RISER1/PCIE4/IOVIB.PF0.VF2      VF      pci_0    iodom1
/SYS/MB/RISER1/PCIE4/IOVIB.PF0.VF3      VF      pci_0    iodom1
```

EXAMPLE 13 CPU コアアクティベーション情報の一覧表示

次の例は、Fujitsu M10 サーバー上の CPU コアアクティベーションに関する情報を示しています。PERMITS 列は、10 個の CPU コアアクティベーションが発行されたことを示しています。この合計には、すべての常時および従量課金ベースの CPU コアアクティベーションが含まれます。PERMANENT 列は、10 個の常時 CPU コアアクティベ

ションが存在することを示しており、それは従量課金ベースの CPU コアアクティベーションが 1 つも発行されていないことを意味します。IN USE 列は、そのうち 2 個の CPU コアアクティベーションだけが現在使用中であることを示しています。REST 列は、8 個の CPU コアアクティベーションが使用可能であることを示しています。

```
primary# ldm list-permits
CPU CORE
PERMITS (PERMANENT)  IN USE      REST
10      (10)          2          8
```

EXAMPLE 14 仮想 SCSI HBA および仮想 SAN の追加

次の例は、特定の SCSI HBA イニシエータポートに仮想 SAN を作成する方法、およびその仮想 SAN に仮想 SCSI HBA を関連付ける方法を示しています。

ldg1 ドメイン内の物理 SCSI HBA イニシエータポートを識別します。

```
primary# ldm list-hba -l ldg1
NAME                                     VSAN
----                                     -
/SYS/MB/SASHBA0/HBA0/PORT1
[/pci@300/pci@1/pci@0/pci@2/scsi@0/iport@1]
/SYS/MB/SASHBA0/HBA0/PORT2
[/pci@300/pci@1/pci@0/pci@2/scsi@0/iport@2]
/SYS/MB/SASHBA0/HBA0/PORT4
[/pci@300/pci@1/pci@0/pci@2/scsi@0/iport@4]
/SYS/MB/SASHBA0/HBA0/PORT8
[/pci@300/pci@1/pci@0/pci@2/scsi@0/iport@8]
/SYS/MB/PCIE1/HBA0/PORT0,0
[/pci@300/pci@1/pci@0/pci@4/SUNW,emlxs@0/fp@0,0]
/SYS/MB/PCIE1/HBA0,1/PORT0,0
[/pci@300/pci@1/pci@0/pci@4/SUNW,emlxs@0,1/fp@0,0]
```

一覧で最後のイニシエータポートに関連付けられたすべての SCSI デバイスを管理するために、ldg1 論理ドメインに仮想 SAN を作成します。

```
primary# ldm add-vsan /SYS/MB/PCIE1/HBA0,1/PORT0,0 port0 ldg1
/SYS/MB/PCIE1/HBA0,1/PORT0,0 resolved to device: /pci@300/pci@1/pci@0/pci@4/SUNW,emlxs@0,1/fp@0,0
```

ldg2 論理ドメインに仮想 SCSI HBA を作成します。これが仮想 SAN と連携して I/O 要求を物理 SCSI デバイスに送信します。

```
primary# ldm add-vhba port0_vhba port0 ldg2
```

新しく作成した仮想 SCSI HBA および仮想 SAN デバイスが存在することを確認します。

```
primary# ldm list -o san,hba ldg1 ldg2
NAME
ldg1

VSAN
NAME      TYPE  DEVICE IPORT
port0     VSAN  [/pci@300/pci@1/pci@0/pci@4/SUNW,emlxs@0,1/fp@0,0]

-----
NAME
ldg2
```

```

VHBA
NAME          VSAN          DEVICE TOUT SERVER
port0_vhba    port0          0      ldg1

```

EXAMPLE 15 ネットワークデバイスの一覧表示

次の例は、ldg1 ドメインのネットワークデバイス情報を示しています。

```

primary# ldm list-netdev ldg1
DOMAIN
ldg1

NAME          CLASS    MEDIA    STATE    SPEED    OVER    LOC
----          -
net0          VNET    ETHER    up        0        --      primary-vsw0/vne t0_ldg1
net3          PHYS    ETHER    up       10000    --      /SYS/MB/RISER1/PCIE4
net4          VSW     ETHER    up       10000    --      ldg1-vsw1
net1          PHYS    ETHER    up       10000    --      /SYS/MB/RISER1/PCIE4
net5          VNET    ETHER    up        0        --      ldg1-vsw1/vnet1_ldg1
net6          VNET    ETHER    up        0        --      ldg1-vsw1/vnet2_ldg1
aggr2        AGGR    ETHER    unknown  0        net1,net3 --
ldoms-vsw0.vport3  VNIC    ETHER    unknown  0        --      ldg1-vsw1/vnet2_ldg1
ldoms-vsw0.vport2  VNIC    ETHER    unknown  0        --      ldg1-vsw1/vnet1_ldg1
ldoms-vsw0.vport1  VNIC    ETHER    unknown  0        --      ldg1-vsw1/vnet2_ldg3
ldoms-vsw0.vport0  VNIC    ETHER    unknown  0        --      ldg1-vsw1/vnet2_ldg2

```

次の例は、-l オプションを指定することで表示される、ldg1 ドメイン上のネットワークデバイスの詳細なリストを示しています。

```

primary# ldm list-netdev -l ldg1
DOMAIN
ldg1

NAME          CLASS    MEDIA    STATE    SPEED    OVER    LOC
----          -
net0          VNET    ETHER    up        0        --      primary-vsw0/vnet0_ldg1
[/virtual-devices@100/channel-devices@200/network@0]
MTU           : 1500 [1500-1500]
IPADDR        : 10.129.241.200/255.255.255.0
MAC_ADDRS     : 00:14:4f:fb:9c:df

net3          PHYS    ETHER    up       10000    --      /SYS/MB/RISER1/PCIE4
[/pci@400/pci@1/pci@0/pci@0/network@0]
MTU           : 1500 [576-15500]
MAC_ADDRS     : a0:36:9f:0a:c5:d2

net4          VSW     ETHER    up       10000    --      ldg1-vsw1
[/virtual-devices@100/channel-devices@200/virtual-network-switch@0]
MTU           : 1500 [1500-1500]
IPADDR        : 192.168.1.2/255.255.255.0
MAC_ADDRS     : 00:14:4f:fb:61:6e

net1          PHYS    ETHER    up       10000    --      /SYS/MB/RISER1/PCIE4
[/pci@400/pci@1/pci@0/pci@0/network@0,1]
MTU           : 1500 [576-15500]
MAC_ADDRS     : a0:36:9f:0a:c5:d2

net5          VNET    ETHER    up        0        --      ldg1-vsw1/vnet1_ldg1
[/virtual-devices@100/channel-devices@200/network@1]
MTU           : 1500 [1500-1500]
IPADDR        : 0.0.0.0 /255.0.0.0

```

```

      : fe80::214:4fff:fef8:5062/ffc0::
MAC_ADDRS : 00:14:4f:f8:50:62

net6          VNET   ETHER   up      0      --      ldg1-vsw1/vnet2_ldg1
[/virtual-devices@100/channel-devices@200/network@2]
MTU          : 1500 [1500-1500]
IPADDR       : 0.0.0.0 /255.0.0.0
      : fe80::214:4fff:fef8:af92/ffc0::
MAC_ADDRS   : 00:14:4f:f8:af:92

aggr2         AGGR    ETHER   unknown 0      net1,net3 --
MODE         : TRUNK
POLICY       : L2,L3
LACP_MODE    : ACTIVE
MEMBER       : net1 [PORTSTATE = attached]
MEMBER       : net3 [PORTSTATE = attached]
MAC_ADDRS    : a0:36:9f:0a:c5:d2

ldoms-vsw0.vport3 VNIC   ETHER   unknown 0      --      ldg1-vsw1/vnet2_ldg1
MTU          : 1500 [576-1500]
MAC_ADDRS    : 00:14:4f:f8:af:92

ldoms-vsw0.vport2 VNIC   ETHER   unknown 0      --      ldg1-vsw1/vnet1_ldg1
MTU          : 1500 [576-1500]
MAC_ADDRS    : 00:14:4f:f8:50:62

ldoms-vsw0.vport1 VNIC   ETHER   unknown 0      --      ldg1-vsw1/vnet2_ldg3
MTU          : 1500 [576-1500]
MAC_ADDRS    : 00:14:4f:f9:d3:88

ldoms-vsw0.vport0 VNIC   ETHER   unknown 0      --      ldg1-vsw1/vnet2_ldg2
MTU          : 1500 [576-1500]
MAC_ADDRS    : 00:14:4f:fa:47:f4
              : 00:14:4f:f9:65:b5
              : 00:14:4f:f9:60:3f

```

EXAMPLE 16 ネットワークデバイス統計情報の一覧表示

次の例は、システム内のすべてのドメインのデフォルトのネットワーク統計情報を示しています。

```

primary# ldm list-netstat
DOMAIN
primary

NAME          IPACKETS      RBYTES      OPACKETS      OBYTES
-----
net3           0              0            0              0
net0          2.72M          778.27M     76.32K         6.01M
net4          2.72M          778.27M     76.32K         6.01M
net6           2              140         1.30K          18.17K
net7           0              0            0              0
net2           0              0            0              0
net1           0              0            0              0
aggr1          0              0            0              0
ldoms-vsw0.vport0 935.40K       74.59M     13.15K         984.43K
ldoms-vsw0.vport1 933.26K       74.37M     11.42K         745.15K
ldoms-vsw0.vport2 933.24K       74.37M     11.46K         747.66K
ldoms-vsw1.vport1 202.26K       17.99M     179.75K        15.69M
ldoms-vsw1.vport0 202.37K       18.00M     189.00K        16.24M
-----
DOMAIN

```

```

ldg1
NAME          IPACKETS    RBYTES      OPACKETS    OBYTES
-----
net0          5.19K      421.57K     68           4.70K
net3          0           0            2.07K        256.93K
net4          0           0            4.37K        560.17K
net1          0           0            2.29K        303.24K
net5          149        31.19K      78           17.00K
net6          147        30.51K      78           17.29K
aggr2         0           0            0            0
ldoms-vsw0.vport3 162        31.69K     52           14.11K
ldoms-vsw0.vport2 163        31.74K     51           13.76K
ldoms-vsw0.vport1 176        42.99K     25           1.50K
ldoms-vsw0.vport0 158        40.19K     45           4.42K
-----
DOMAIN
ldg2
NAME          IPACKETS    RBYTES      OPACKETS    OBYTES
-----
net0          5.17K      418.90K     71           4.88K
net1          2.70K      201.67K     2.63K        187.01K
net2          132        36.40K      1.51K        95.07K
-----
DOMAIN
ldg3
NAME          IPACKETS    RBYTES      OPACKETS    OBYTES
-----
net0          5.16K      417.43K     72           4.90K
net1          2.80K      206.12K     2.67K        190.36K
net2          118        35.00K      1.46K        87.78K

```

EXAMPLE 17 依存関係の一覧表示

次の例は、-l オプションを指定することで表示される、ドメインの詳細な依存関係情報を示しています。

```

primary# ldm list-dependencies -l
DOMAIN      DEPENDENCY  TYPE      DEVICE
primary
svcdom
ldg0        primary    VDISK     primary-vds0/vdisk0
           VNET      primary-vsw0/vnet0
svcdom     VDISK     svcdom-vds0/vdisk1
           VNET      svcdom-vsw0/vnet1
ldg1        primary    VDISK     primary-vds0/vdisk0
           VNET      primary-vsw0/vnet0
           IOV      /SYS/MB/NET0/IOVNET.PF0.VF0
svcdom     VDISK     svcdom-vds0/vdisk1
           VNET      svcdom-vsw0/vnet1
           IOV      /SYS/MB/NET2/IOVNET.PF0.VF0

```

次の例は、-l オプションと -r オプションの両方を指定することで表示される、依存関係ごとにグループ化された依存要素に関する詳細情報を示しています。

```

primary# ldm list-dependencies -r -l
DOMAIN      DEPENDENT  TYPE      DEVICE
primary     ldg0        VDISK     primary-vds0/vdisk0
           VNET      primary-vsw0/vnet0

```

```

ldg1          VDISK      primary-vds0/vdisk0
VNET         primary-vsw0/vnet0
IOV          /SYS/MB/NET0/IOVNET.PF0.VF0
svcdom       ldg0       VDISK      svcdom-vds0/vdisk1
VNET         svcdom-vsw0/vnet1
ldg1          VDISK      svcdom-vds0/vdisk1
VNET         svcdom-vsw0/vnet1
IOV          /SYS/MB/NET2/IOVNET.PF0.VF0

```

EXAMPLE 18 リソースグループの一覧表示

次の例は、各リソースグループの内容に関する情報を示しています。

```

primary# ldm list-rsrc-group
NAME                                CORE  MEMORY  IO
/SYS/CMU1                           12   512G    4
/SYS/CMU2                           12   512G    4
/SYS/CMU3                           12   512G    4

```

次の例は、/SYS/CMU1 リソースグループの内容に関する詳細情報を示しています。

```

primary# ldm list-rsrc-group -l /SYS/CMU1
NAME                                CORE  MEMORY  IO
/SYS/CMU1                           12   512G    4
CORE
BOUND                                CID
primary                             (64, 66, 68, 70, 72, 74, 80, 82, 84, 86, 88, 90)
MEMORY
PA          SIZE          BOUND
0x201ff0000000 256M          _sys_
0x20000e400000 412M          _sys_
0x200000000000 102M          _sys_
0x200006600000 32M           _sys_
0x200030000000 129792M       primary
0x280000000000 128G          primary
IO
DEVICE      PSEUDONYM      BOUND
pci@500     pci_8           primary
pci@540     pci_9           primary
pci@580     pci_10          primary
pci@5c0     pci_11          primary

```

EXAMPLE 19 inter-vnet リンクのステータスの取得

次の例は、inter-vnet-link=auto の場合に、inter-vnet リンクが有効になっているか、無効になっているかに関する情報を示します。

- この例の ldm list -o network の出力は、primary-vsw1 の inter-vnet-link が auto であり、仮想ネットワークの数が ldmd/auto_inter_vnet_link_limit SMF プロパティによって指定された最大数以下であることを示しています。結果として、inter-vnet リンクが有効になっています。そのため、INTER-VNET-LINK フィールドの値は、on/auto です。

```

# ldm list -o network primary
NAME
primary

```

```

MAC
00:21:28:c1:40:5e

VSW
NAME          MACADDRESS          NET-DEV  DVID|PVID|VIDs
-----
primary-vsw0  00:14:4f:fb:e8:d8  net0     1|1|--
  DEVICE          :switch@0          ID       :0
  LINKPROP        :--                 MTU      :1500
  INTER-VNET-LINK :on                 MODE     :--

primary-vsw1  00:14:4f:f9:b6:21  --       1|1|--
  DEVICE          :switch@1          ID       :1
  LINKPROP        :--                 MTU      :1500
  INTER-VNET-LINK :on/auto           MODE     :--

```

- この例の `ldm list -o network` の出力は、`primary-vsw1` の `inter-vnet-link=auto` および仮想ネットワークの数が `ldmd/auto_inter_vnet_link_limit` SMF プロパティによって指定された最大値を超えていることを示しています。結果として、`inter-vnet` リンクが無効になっています。そのため、`INTER-VNET-LINK` フィールドの値は、`off/auto` です。

```

# ldm list -o network primary
NAME
primary

MAC
00:21:28:c1:40:5e

VSW
NAME          MACADDRESS          NET-DEV  DVID|PVID|VIDs
-----
primary-vsw0  00:14:4f:fb:e8:d8  net0     1|1|--
  DEVICE          :switch@0          ID       :0
  LINKPROP        :--                 MTU      :1500
  INTER-VNET-LINK :on                 MODE     :--

primary-vsw1  00:14:4f:f9:b6:21  --       1|1|--
  DEVICE          :switch@1          ID       :1
  LINKPROP        :--                 MTU      :1500
  INTER-VNET-LINK :off/auto           MODE     :--

```

EXAMPLE 20 リンク状態を報告する仮想スイッチおよび仮想ネットワークの作成

- 次の例は、ldg1 ゲストドメインに ldg1-vsw0 仮想スイッチを追加します。仮想スイッチの net-dev 値は net1 になります。デフォルトで、仮想スイッチは linkprop=phys-state になります。

```
primary# ldm add-vsw net-dev=net1 ldg1-vsw0 ldg1
primary# ldm list -o network ldg1
...
VSW
  NAME           MACADDRESS           NET-DEV   DVID|PVID|VIDs
  ----           -
  ldg1-vsw0      00:14:4f:f8:3e:af    net1     1|1|--
             DEVICE           :switch@0         ID      :0
             LINKPROP        :phys-state       MTU     :1500
             INTER-VNET-LINK :on/auto          MODE    :--
...

```

- 次の例は、ldg1-vnet1 仮想ネットワークを ldg1 ゲストドメインの ldg1-vsw0 仮想スイッチに追加します。デフォルトで、仮想ネットワークは linkprop=phys-state になります。

```
primary# ldm add-vnet ldg1-vnet1 ldg1-vsw0 ldg1
primary# ldm list -o network ldg1
...
NETWORK
  NAME           SERVICE           MACADDRESS           PVID|PVLAN|VIDs
  ----           -
  ldg1-vnet1     ldg1-vsw0@ldg1    00:14:4f:fb:86:00    1|--|--
             DEVICE           :network@1         ID      :1
             LINKPROP        :phys-state       MTU     :1500
             MAXBW           :--               MODE    :--
             CUSTOM          :disable
             PRIORITY        :--               COS     :--
             PROTECTION      :--
...

```

- 次の例は、ldg1-vnet2 仮想ネットワークを ldg1 ゲストドメインの primary-vsw0 仮想スイッチに追加します。primary-vsw0 が linkprop=phys-state でない場合でも、ldg1-vnet2 仮想ネットワークはデフォルトで linkprop=phys-state になります。

```
primary# ldm add-vnet ldg1-vnet2 primary-vsw0 ldg1
primary# ldm list -o network ldg1
...
VSW
  NAME           MACADDRESS           NET-DEV   DVID|PVID|VIDs
  ----           -

```

```

-----
primary-vsw0  00:14:4f:f8:3c:a0  net0  1|1|--
  DEVICE      :switch@0          ID    :0
  LINKPROP    :--                MTU   :1500
  INTER-VNET-LINK :on/auto        MODE  :--

NETWORK
NAME          SERVICE                MACADDRESS          PVID|PVLAN|VIDs
-----
ldg1-vnet2    primary-vsw0@primary  00:14:4f:fa:b0:bd  1|--|--
  DEVICE      :network@2        ID    :2
  LINKPROP    :phys-state      MTU   :1500
  MAXBW       :--                MODE  :--
  CUSTOM      :disable
  PRIORITY    :--                COS   :--
  PROTECTION  :--
...

```

EXAMPLE 21 カスタムプロパティ値の一覧表示

次の `ldm list` コマンドは、`custom` プロパティが `enable`、`custom-max-mac-addr=3`、および `custom-max-vlans=3` に設定されている構成を表示します。

```

primary# ldm list ldg1
...
NAME          SERVICE                MACADDRESS          PVID|PVLAN|VIDs
-----
temp          primary-vsw0@primary  00:14:4f:fb:03:fd  1|--|--
  DEVICE      :network@4        ID    :4
  LINKPROP    :phys-state      MTU   :1500
  MAXBW       :--                MODE  :--
  CUSTOM      :enable
  MAX-CUSTOM-MACS:3          MAX-CUSTOM-VLANS:3
  PRIORITY    :--                COS   :--
  PROTECTION  :--
...

```

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。

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	pkg:/system/ldoms/ldomsmanager

属性型	属性値
インタフェースの安定性	不確実

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

[Oracle VM Server for SPARC 3.4 管理ガイド](#)

Name

ldmd — Logical Domains Manager デーモン

/opt/SUNWldm/bin/ldmd

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。ldmd デーモンは、サービスプロセッサ (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。物理ドメインがあるプラットフォームの場合、Logical Domains Manager は、各物理ドメインの制御ドメインでのみ実行されます。制御ドメインの名前は primary です。

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、破棄、再構成、およびリブート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。

SMF プロパティ

svccfg コマンドを使用して、次のプロパティを変更できます。

ldmd/audit

Logical Domains Manager 監査レコードを生成するかどうかを指定します。値 true は Logical Domains Manager 監査レコードを生成します。デフォルトで、値は false です。

ldmd/auto_inter_vnet_link_limit

inter-vnet-link=auto の場合、このプロパティは inter-vnet リンクを有効にすることが許可されている仮想ネットワークの最大数です。同じ仮想スイッチに接続された仮想ネットワークの数がこの制限を超えると、inter-vnet リンクは自動的に無効になります。デフォルト値は 8 です。

ldmd/autorecovery_policy

自動回復ポリシーを指定します。このプロパティには、次の値のいずれかを指定できます。

- 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。これは、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ブート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。

`ldmd/autoreplacement_policy_cpu`

Fujitsu M10 プラットフォームの CPU 自動置換ポリシーを指定します。このプロパティには、次の値のいずれかを指定できます。

- `autoreplacement_policy_cpu=1` – 障害が発生した CPU リソースの自動的な交換を試みる CPU 自動置換プロセスを有効にします。これはデフォルトのポリシーです。
- `autoreplacement_policy_cpu=0` – CPU 自動置換プロセスを無効にします。

`ldmd/autoreplacement_retry_counter`

Fujitsu M10 プラットフォーム上の CPU 自動置換プロセスの最大再試行回数を指定します。値 `0` を指定すると、試行の回数が無制限になります。デフォルト値は `5` 回です。

`ldmd/autoreplacement_retry_interval`

Fujitsu M10 プラットフォーム上の CPU 自動置換プロセスの再試行間の間隔を秒単位で指定します。最小の間隔は `1` 秒です。デフォルト値は `300` 秒です。

`ldmd/default_quick_stop`

使用する停止方法を指定します。Logical Domains Manager の起動時に `default_quick_stop` を `true` に設定した場合、コマンド行でオーバーライドされないかぎり、次の `ldm stop-domain` コマンドで `-q` の方法が使用されます。`default_quick_stop` を `false` に設定した場合、`ldm stop-domain` コマンドでは、指定されたドメインで使用可能な場合は `shutdown` コマンドが使用され、それ以外の場合は `-q` オプションに自動的にフォールバックします。デフォルト値は `false` です。

`ldmd/fj_ppar_dr_policy`

SP が `deleteboard` コマンドを使用して、動的メモリー削除要求を行う場合に使用するメモリー削除ポリシーを指定します。このプロパティは、Fujitsu M10 プラットフォームにのみ適用されます。

有効なポリシー値は次のとおりです。

- **auto** – 最近サポートされた `fj_ppar_dr_policy` ポリシー値 `ratio` を使用します。デフォルトのポリシーは `auto` です。
- **ratio** – 指定された量のメモリーがドメインから削除されるまで、各ドメインのメモリーサイズを同じ相対比率で維持します。
このポリシーでは、システム内のすべてのドメインが Oracle Solaris 11.3 以上の OS でインストールされている必要があります。`targeted` ポリシーは、いずれかのドメインが古いバージョンの Oracle Solaris OS でインストールされている場合に使用されます。
- **targeted** – 削除するボードのメモリーが割り当てられているドメインからのみメモリーを削除します。

`ldmd/hops`

MAC 衝突検出メッセージが破棄される前に、たどることができるメッセージのホップ数(またはサブネット数)を指定します。有効な値は、0(デフォルト値を使用)、1(同じサブネット)、32(同じサイト)、64(同じ領域)、128(同じ大陸)、および 255(無制限)です。デフォルトでは、`hops` は 1 に設定され、これは同じサブネット上のその他のマネージャーにのみマルチキャストメッセージが送信されることを示します。`hops` を 0 に設定すると、`ldmd` ではデフォルト値の 1 が使用されます。

`ldmd/incoming_migration_enabled`

`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。

`ldmd/migration_authn_required`

システムへのドメイン移行の認証を強制的に行います。デフォルトの動作では、ユーザーを認証する必要があります。このプロパティによって、システムからの移行は影響を受けません。パスワードを指定しない移行を許可するには、ターゲットマシン上で `migration_authn_required` を `false` に設定します。

このプロパティは XML/XMPP インタフェースでのみ使用され、移行の際に常にパスワードを入力する必要がある `ldm migrate-domain` コマンドでは使用されません。

`ldmd/pm_observability_enabled`

`ldmd` の起動時に電源管理 (PM) 可観測性モジュールを有効または無効にします。このモジュールを有効にすると、`ldmpower` コマンドを実行して消費電力データを表示できます。[ldmpower\(1M\) on page 111](#) のマニュアルページを参照してください。デフォルト値は `true` です。

`ldmd/outgoing_migration_enabled`

`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。

ldmd/recovery_mode

SP で復旧モードがリクエストされたときに行われるアクションを決定します。このプロパティは、回復モードをサポートするシステム (SPARC T5、SPARC M5、SPARC M6、Fujitsu M10 プラットフォーム など) 에만適用されます。有効な値は次のとおりです。

- **auto** は、管理者による介入なしで自動的に復旧を実行します。これはデフォルトの動作です。
- **never** は、復旧は実行されず、システムが工場出荷時のデフォルト構成のままになることを指定します。

ldmd/recovery_mode_boot_timeout

回復中のルートドメインのブートタイムアウトを分で指定します。有効な値は5分以上です。デフォルト値は30分間です。

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 デモンが別のシステム上の ldmd デモンと通信できるようにします。デフォルト値は true です。

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	pkg:/system/ldoms/ldomsmanager
インタフェースの安定性	不確実

[svcs\(1\)](#)、[drd\(1M\)](#)、[ldm\(1M\)](#) on page 7、[ldmad\(1M\)](#)、[ldmpower\(1M\)](#) on page 111、[svcadm\(1M\)](#)、[vntsd\(1M\)](#)、[attributes\(5\)](#)、[smf\(5\)](#)

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。

これらの SMF プロパティの変更を有効にするには、サービスをリフレッシュして再起動する必要があります。

`svcadm` コマンドを使用して、再起動の有効化、無効化、リフレッシュ、またはリクエスト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ステータスをクエリーします。

ldmd SMF プロパティの詳細については、[Oracle VM Server for SPARC 3.4 管理ガイド](#)を参照してください。

Name

ldmp2v — Oracle VM Server for SPARC Physical-to-Virtual (P2V) 変換ツール用のコマンド行インタフェース

```
ldmp2v collect [-a flash|none] [-O "flarcree-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-name

ldmp2v prepare -R guest-root [-c cpu] [-M memsize] [-o keep-hostid] [-o keep-mac]
[-v] -d data-dir domain-name

ldmp2v prepare -C domain-name

ldmp2v convert -i install-image -d data-dir [-v] [-x skip-ping-test] domain-name

ldmp2v convert [-j] -n interface -d data-dir [-v] [-x skip-ping-test] domain-name
```

Oracle VM Server for SPARC Physical-to-Virtual (P2V) 変換ツールは、既存の物理システムを、チップマルチスレッディング (CMT) システム上の論理ドメインで Oracle Solaris 10 OS を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、Solaris 9、または Oracle Solaris 10 OS 以降を実行する sun4u SPARC システム、または Oracle Solaris 10 OS を実行する非 Logical Domains sun4u システムを指定できます。ソースシステムのイメージは Oracle VM Server for SPARC ドメインに変換され、必要に応じて、プロセス中に最新の Oracle Solaris 10 バージョンにアップグレードされます。

Note - ldmp2v コマンドは、ZFS ルートがある Oracle Solaris 10 OS または Oracle Solaris 11 OS を実行する SPARC システムではサポートされていません。

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。

- **収集フェーズ**。物理ソースシステム上で実行されます。collect では、ソースシステムについて収集され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が作成されます。
- **準備フェーズ**。ターゲットシステムの制御ドメイン上で実行されます。prepare では、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が作成されます。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上で実行されます。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。

次のセクションからは、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。

収集フェーズ

```
ldmp2v collect [-a flash|none] [-O "flarcreate-options"] [-v] [-x mount-point [-x ...]]
-d data-dir
```

`ldmp2v collect` コマンドは、次のオプションを使用します。

- | | |
|--------------------------------------|---|
| <code>-a flash none</code> | 使用するアーカイブ方法を指定します。有効な値は、 <code>flash</code> または <code>none</code> です。デフォルトは <code>flash</code> です。 |
| <code>-O "flarcreate-options"</code> | <code>flarcreate</code> 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。 <code>flarcreate</code> のオプションのうち許可されているものは、 <code>-c</code> と <code>-x</code> のみです。 <code>-c</code> オプションはアーカイブを圧縮し、 <code>-x</code> 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 <code>flarcreate</code> オプションを指定できます。 <code>-O</code> オプションは、 <code>-a flash</code> 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。 |
| <code>-v</code> | 詳細モードを使用します。このモードでは、 <code>ldmp2v</code> によって発行されるメッセージがより詳細になります。 |
| <code>-x mount-point</code> | <code>mount-point</code> 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。 |
| <code>-d data-dir</code> | 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、 <code>root</code> 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-name
ldmp2v prepare -R guest-root [-c cpu] [-M memsize] [-o keep-hostid] [-o keep-mac]
[-v] -d data-dir domain-name
ldmp2v prepare -C domain-name
```

`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。

- | | |
|--------------------------------|---|
| <code>domain-name</code> | 操作の対象となる論理ドメインを指定します。 |
| <code>-b zvol file disk</code> | 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 (<code>zvol</code>)、プレーンファイル (<code>file</code>)、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 (<code>disk</code>) で構成されま |

す。このオプションは、`/etc/ldmp2v.conf` 内の `BACKEND_TYPE` の設定をオーバーライドします。

- 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** オプションの有効な使用方法として、**-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ようにします。`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、デフォルト名をオーバーライドします。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`、G 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指定します。記憶域サイズは *numunit* として指定され、*num* はメモリー容量、*unit* は次のいずれかになります。
- `m` または `M` は M バイトを表します。

- `g` または `G` は G バイトを表します。

`unit` が指定されていない場合は、M バイトになります。

デフォルトでは、`ldmp2v` コマンドが、物理システム内にある同量のメモリーを論理ドメインに割り当てます。必要であれば、ゲストドメインの最小記憶域サイズを満たすために、`-M` オプションで指定された記憶域サイズは 1G バイトになるように調整されます。

<code>-o keep-hostid</code>	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。
<code>-o keep-mac</code>	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。
<code>-p prefix</code>	バックエンドデバイスを作成する場所を指定します。 <code>zvol</code> バックエンドの場合は ZFS データセット、 <code>file</code> 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、 <code>/etc/ldmp2v.conf</code> の <code>BACKEND_PREFIX</code> パラメータより優先されます。
<code>-R guest-root</code>	非自動モードを選択します。OS イメージの変更手順は、 <code>guest-root</code> 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 <code>/etc/vfstab</code> を更新して、 <code>guest-root</code> 配下のファイルシステムのレイアウトに一致させます。
<code>-s</code>	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、 <code>/etc/ldmp2v.conf</code> の <code>BACKEND_SPARSE</code> パラメータより優先されます。
<code>-v</code>	詳細モードを使用します。このモードでは、 <code>ldmp2v</code> によって発行されるメッセージがより詳細になります。
<code>-x no-auto-adjust-fs</code>	<p>/、<code>/usr</code>、および <code>/var</code> 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。</p> <p><code>-m</code> 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。</p>
<code>-x remove-unused-slices</code>	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] [-x skip-ping-test] domain-name
ldmp2v convert [-j] -n interface -d data-dir [-v] [-x skip-ping-test] domain-name
```

ldmp2v convert コマンドは、次のオプションを使用します。

-d <i>data-dir</i>	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 <i>install-image</i>	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 <i>interface</i>	ネットワークインストールサーバーを使用する場合にブート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。
-x skip-ping-test	ソースシステムの IP アドレスが有効になっているかどうかを判断するための ping テストがスキップされます。このオプションは、元のシステムがアクティブではないときなど、重複する IP アドレスが存在しないことが明らかな場合のみ使用してください。



Caution - 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。

このセクションでは、3つのフェーズの例が含まれます。

EXAMPLE 22 収集フェーズの例

ldmp2v collect コマンドの使用法の例を次に示します。

- **NFS マウントされたファイルシステムを共有する。** 次の例では、collect フェーズを実行するためのもっとも簡単な方法を示します。ここでは、ソースシステムとターゲットシステムで NFS マウントされたファイルシステムを共有します。

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

- **NFS マウントされた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で NFS マウントされたファイルシステムを共有しない場合、ファイルシステムのイメージをローカルストレージに書き込んだあとに、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手順をスキップする。**すでに NetBackup などのサードパーティー製バックアップツールを使用してシステムのバックアップが可能である場合、none アーカイブ方法を使用すればファイルシステムのバックアップ手順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
```

Note --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と、flarcreate コマンドにオプションを渡すことによって、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。

- **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"
```

EXAMPLE 23 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
```

EXAMPLE 24 変換フェーズの例

`ldmp2v convert` コマンドの使用法の例を次に示します。

- **ネットワークインストールサーバーを使用する。** `ldmp2v convert` コマンドが、指定された仮想ネットワークインタフェースを使用することによって、ネットワーク上の論理ドメインをブートします。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。

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 イメージを使用する。** `ldmp2v convert` コマンドが、Solaris DVD ISO イメージを論理ドメインに接続し、そこからブートします。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。

Note - `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リポート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
```

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。

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	<code>pkg:/system/ldoms/ldomsmanager</code>
インタフェースの安定性	不確実

[ldm\(1M\)](#) on page 7、[attributes\(5\)](#)

[Oracle VM Server for SPARC 3.4 管理ガイド](#)

Name

ldmpower — ドメインごとの消費電力情報の表示

```
ldmpower [-ehiprstvx | -o hours | -m minutes] [-c resource] [-l ldom[,ldom[,...]]]
        [interval [count]]
```

ldmpower コマンドは、ドメインの消費電力データ (ワット) を表示します。デフォルトでは、実行中の各ドメインで消費されるプロセッサの電力が表示されます。オプションを使用すると、メモリーの消費電力や、システム全体の消費電力を実行中のドメイン間で割ったものを表示できます。ドメインごとのシステム消費電力は、ドメインごとのプロセッサとメモリーの消費電力から外挿されます。

オプションを付けずに指定した場合、ldmpower コマンドは、最近 15、30、60 秒間の平均消費電力を表示します。コマンドは最新の消費電力データを表示することもできます。より長い履歴が必要な場合、最近 1 時間を対象とした最大 60 個の 1 分間の平均値、および最近 14 日間を対象とした 336 個の 1 時間の平均値を表示できます。

このコマンドを非特権ユーザーとして実行するには、LDoms Power Mgmt Observability 権利プロファイルが割り当てられている必要があります。すでに LDoms Management または LDoms Review 権利プロファイルが割り当てられている場合、ldmpower コマンドを実行するためのアクセス権が自動的に付与されます。

このコマンドは、UltraSPARC T2、UltraSPARC T2 Plus、および Fujitsu M10 プラットフォームではサポートされていません。

ldmpower コマンドは、次のオプションおよびオペランドを使用します。

- | | |
|-------------|--|
| -c resource | 指定されたリソースタイプ <i>resource</i> の消費電力をドメインごとに表示します。有効な値は <i>processors</i> および <i>memory</i> です。このオプションの長いバージョンは <code>--component</code> です。 |
| -e | データ記録の開始以降の消費電力の最小値と最大値を表示します。このオプションの長いバージョンは <code>--extremes</code> です。 |
| -h | ldmpower コマンドオプションの説明を表示します。このオプションの代替の短いバージョンは <code>-?</code> 、このオプションの長いバージョンは <code>--help</code> です。 |
| -i | 瞬間消費電力データを表示します。このオプションの長いバージョンは <code>--instant</code> と <code>--instantaneous</code> です。 |
| -l ldom | 指定された 1 つ以上のドメインの消費電力データを表示します。ドメイン名はコンマで区切られます。このオプションの長いバージョンは <code>--list</code> です。 |

リクエストされた期間よりも短い時間でドメインがブートされた場合は、表示されるデータ量が少なくなることにご注意ください。

- m minutes** 最大 1 時間の平均消費電力データを一定の 1 分間隔で表示します。このオプションの長いバージョンは `--minutes` です。
リクエストされた期間よりも短い時間で Logical Domains デーモンが実行された場合は、表示されるデータ量が少なくなることにご注意ください。
- o hours** 最大 14 日間の平均消費電力データを一定の 1 時間間隔で表示します。このオプションの長いバージョンは `--hours` です。
リクエストされた期間よりも短い時間で Logical Domains デーモンが実行された場合は、表示されるデータ量が少なくなることにご注意ください。
- p** プロセッサ、メモリー、およびファンを含むシステム全体の総消費電力を表示します。このオプションの長いバージョンは `--platform` です。
- r** 過去 15、30、60 秒間の移動平均消費電力の履歴を表示します。このオプションの長いバージョンは `--rolling` です。
- s** 出力ヘッダーが表示されなくなります。このオプションの長いバージョンは `--suppress` です。
- t** 出力にタイムスタンプを表示します。このオプションの長いバージョンは `--timestamps` です。
- v** バージョンデータを表示します。このオプションの長いバージョンは `--version` です。
- x** 外挿消費電力の平均値を表示します。これらの平均値には、ドメインごとに消費されているシステムの全体的なワット数の割合が反映されます。ドメインごとのシステム消費電力は、ドメインごとのプロセッサとメモリーの消費電力から外挿されます。このオプションは、`-o`、`-i`、`-m`、および `-r` のいずれかのオプションと同時に使用できます。このオプションの長いバージョンは `--extrapolate` です。
- interval** 出力を *interval* 秒ごとに 1 回レポートします。
- count** レポートを *count* 回表示します。*count* を指定する場合は、まず *interval* を指定する必要があります。

EXAMPLE 25 プロセッサの消費電力データの表示

次のコマンドは、すべてのドメインについて、15 秒、30 秒、60 秒のプロセッサの移動平均消費電力データを表示します。

```
# ldmpower
Processor Power Consumption in Watts
DOMAIN 15_SEC_AVG 30_SEC_AVG 60_SEC_AVG
primary 75          84          86
gdom1   47          24          19
gdom2   10          24          26
```

EXAMPLE 26 システム全体の消費電力データの表示

次のコマンドは、システム全体の瞬間消費電力(ワット)を表示します。

- 次のコマンドは、システム、プロセッサ、メモリー、およびファンについて、瞬間消費電力データを表示します。-t オプションは、出力にタイムスタンプを含めません。

```
# ldmpower -tp
Overall Instant System Power Consumption in Watts
System          2012.08.23 23:13:33    738
Processors      2012.08.23 23:13:33    295
Memory          2012.08.23 23:13:33    138
Fans            2012.08.23 23:13:33     28
```

- 次のコマンドは、システム、プロセッサ、メモリー、およびファンについて、瞬間消費電力データを表示します。

```
# ldmpower -p
Overall Instant System Power Consumption in Watts
System          738
Processors      295
Memory          138
Fans            25
```

EXAMPLE 27 メモリーとプロセッサの消費電力データの表示

次のコマンドは、メモリー、プロセッサ、またはその両方について、ドメインごとの消費電力(ワット)を表示します。

- 次のコマンドは、メモリーの消費電力データを表示します。

```
# ldmpower -c memory
Memory Power Consumption in Watts
DOMAIN 15_SEC_AVG 30_SEC_AVG 60_SEC_AVG
primary 138          138          138
```

```
ldg1 19 19 19
ldg2 19 19 19
```

- 次のコマンドは、メモリーとプロセッサの移動消費電力データを表示します。

```
# ldmpower -c memory -c processors
Processor Power Consumption in Watts
DOMAIN 15_SEC_AVG 30_SEC_AVG 60_SEC_AVG
primary 63 61 60
ldg1 9 10 10
ldg2 9 9 10
```

```
Memory Power Consumption in Watts
DOMAIN 15_SEC_AVG 30_SEC_AVG 60_SEC_AVG
primary 138 138 138
ldg1 19 19 19
ldg2 19 19 19
```

- 次のコマンドは、メモリーとプロセッサの瞬間消費電力データを表示します。

```
# ldmpower -c memory -c processors -i
Processor Power Consumption in Watts
DOMAIN INSTANT
primary 292
ldg1 10
ldg2 10
```

```
Memory Power Consumption in Watts
DOMAIN INSTANT
primary 138
ldg1 19
ldg2 19
```

EXAMPLE 28 指定されたドメインのプロセッサの消費電力データの表示

次のコマンドは、gdom2 および gdom5 ドメインについて、プロセッサの瞬間消費電力データを表示します。-i オプションは、瞬間消費電力データを表示し、-t オプションは、出力にタイムスタンプを表示します。オプションの -1 引数を指定すると、gdom2 および gdom5 ドメインに関する情報のみが出力に表示されます。1 番目のオペランド 10 は、消費電力データが 10 秒間隔でレポートされることを示します。2 番目のオペランド 5 は、データ出力の反復回数です。

```
# ldmpower -it1 gdom2, gdom5 10 5
Processor Power Consumption in Watts
DOMAIN      TIMESTAMP      INSTANT
gdom2       2013.05.17 11:14:45 13
gdom5       2013.05.17 11:14:45 24

gdom2       2013.05.17 11:14:55 18
gdom5       2013.05.17 11:14:55 26
```

gdom2	2013.05.17 11:15:05	9
gdom5	2013.05.17 11:15:05	16
gdom2	2013.05.17 11:15:15	15
gdom5	2013.05.17 11:15:15	19
gdom2	2013.05.17 11:15:25	12
gdom5	2013.05.17 11:15:25	18

EXAMPLE 29 指定された時間での消費電力データの表示

次のコマンドは、指定された時間の消費電力データを表示します。

- 次のコマンドは、すべてのドメインについて、最近 12 時間の平均消費電力データを表示します。`-e` オプションは、データ記録の開始以降の最小値と最大値を表示し、`-t` オプションは、出力にタイムスタンプを表示します。`-o` オプション引数には、1 時間ごとの平均消費電力データを表示する時間数を指定します。データは最後にリクエストされた 1 時間ごとの計算から 1 時間の間隔で表示されます。

```
# ldmpower -eto 12
Per domain MINIMUM and MAXIMUM power consumption ever recorded:
primary      2013.05.17 08:53:06    3           Min Processors
primary      2013.05.17 08:40:44   273         Max Processors
gdom1        2013.05.17 09:56:35    2           Min Processors
gdom1        2013.05.17 08:53:06   134         Max Processors
gdom2        2013.05.17 10:31:55    2           Min Processors
gdom2        2013.05.17 08:56:35   139         Max Processors

primary      2013.05.17 08:53:06    99          Min Memory
primary      2013.05.17 08:40:44   182         Max Memory
gdom1        2013.05.17 09:56:35    13          Min Memory
gdom1        2013.05.17 08:53:06   20          Max Memory
gdom2        2013.05.17 10:31:55   65          Min Memory
gdom2        2013.05.17 08:56:35   66          Max Memory

Processor Power Consumption in Watts
12 hour's worth of data starting from 2013.05.16 23:17:02
DOMAIN      TIMESTAMP              1 HOUR AVG
primary     2013.05.17 09:37:35   112
gdom1      2013.05.17 09:37:35   15
gdom2      2013.05.17 09:37:35   26

primary     2013.05.17 10:37:35   96
gdom1      2013.05.17 10:37:35   12
gdom2      2013.05.17 10:37:35   21

primary     2013.05.17 11:37:35   85
```

```

gdom1          2013.05.17 11:37:35    11
gdom2          2013.05.17 11:37:35    23
...

```

- 次のコマンドは、すべてのドメインについて、最近 30 分間の平均消費電力データを表示します。-e オプションは、データ記録の開始以降の最小値と最大値を表示し、-t オプションは、出力にタイムスタンプを表示します。-m オプション引数には、平均消費電力データを表示する分数を指定します。データは 1 分間隔で表示されます。

```
# ldmpower -etm 30
```

```
Per domain MINIMUM and MAXIMUM power consumption ever recorded:
```

```

primary        2013.05.17 06:59:55    151           Min Processors
primary        2013.05.17 06:56:20    682           Max Processors
gdom1          2013.05.17 06:56:25    36            Min Processors
gdom1          2013.05.17 07:02:10    318           Max Processors
gdom2          2013.05.17 06:56:25    42            Min Processors
gdom2          2013.05.17 07:05:45    417           Max Processors

```

```

primary        2013.05.17 06:59:55    137           Min Memory
primary        2013.05.17 06:56:20    138           Max Memory
gdom1          2013.05.17 06:56:25    59            Min Memory
gdom1          2013.05.17 07:02:10    60            Max Memory
gdom2          2013.05.17 06:56:25    59            Min Memory
gdom2          2013.05.17 07:05:45    60            Max Memory

```

```
30 minute's worth of data starting from 2012.08.29 12:05:33:
```

```

DOMAIN        TIMESTAMP                1 MIN AVG
primary       2013.05.17 12:05:33    574
gdom1        2013.05.17 12:05:33    268
gdom2        2013.05.17 12:05:33    386

primary       2013.05.17 12:06:33    555
gdom1        2013.05.17 12:06:33    250
gdom2        2013.05.17 12:06:33    364

primary       2013.05.17 12:07:33    498
gdom1        2013.05.17 12:07:33    276
gdom2        2013.05.17 12:07:33    364
...

```

EXAMPLE 30 すべてのドメインの外挿消費電力データの表示

次のコマンドは、すべてのドメイン (primary、gdom1、および gdom2) について、外挿消費電力データを表示します。

```
# ldmpower -x
System Power Consumption in Watts
DOMAIN      15_SEC_AVG      30_SEC_AVG      60_SEC_AVG
primary     585/57.47%      701/68.96%      712/70.22%
gdom1       132/12.97%      94/9.31%        94/9.30%
gdom2       298/29.27%      218/21.47%      205/20.22%
```

EXAMPLE 31 すべてのリソースの消費電力データの表示

次のコマンドは、すべてのリソース (システム、プロセッサ、メモリー、およびファン) について、システム全体の移動平均電力を表示します。

デフォルトでは、移動平均電力が表示されるため、この出力は `ldmpower -rp` コマンドの出力に一致します。

```
# ldmpower -p
Resource Power Consumption in Watts
TYPE      15_SEC_AVG      30_SEC_AVG      60_SEC_AVG
System    1016            1016            1016
Processors 270             270             269
Memory    179             179             179
Fans      202             202             203
```

EXAMPLE 32 すべてのリソースの瞬間消費電力データの表示

次のコマンドは、すべてのリソース (システム、プロセッサ、メモリー、およびファン) について、システム全体の瞬間平均電力を表示します。

```
# ldmpower -ip
Resource Power Consumption in Watts
TYPE      INSTANT
System    1012
Processors 270
Memory    179
Fans      203
```

次の属性の説明については、[attributes\(5\)](#) の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	pkg:/system/ldoms/ldomsmanager

[ldmd\(1M\)](#) on page 99、[attributes\(5\)](#)

名前

ovmtadm — Oracle VM for SPARC テンプレートの管理

```
/opt/ovmtutils/bin/ovmtadm config  
/opt/ovmtutils/bin/ovmtadm create  
/opt/ovmtutils/bin/ovmtadm deploy  
/opt/ovmtutils/bin/ovmtadm help  
/opt/ovmtutils/bin/ovmtadm remove
```

ovmtadm コマンドを使用すると、ovmcreate、ovmtconfig、および ovmtdeploy コマンドを起動して、テンプレートを作成、構成、配備、および削除できます。

このコマンドは、構成するドメインが含まれるシステムの制御ドメイン上で実行します。

1 回につき 1 つのドメインのみを構成できます。

ovmtadm コマンドには、次のサブコマンドがあります。

config

ovmtconfig コマンドを実行します。コマンドオプションについては、[123 ページの ovmtconfig\(1M\)](#) のマニュアルページを参照してください。

create

ovmcreate コマンドを実行します。コマンドオプションについては、[129 ページの ovmcreate\(1M\)](#) のマニュアルページを参照してください。

deploy

ovmtdeploy コマンドを実行します。コマンドオプションについては、[133 ページの ovmtdeploy\(1M\)](#) のマニュアルページを参照してください。

help

コマンドについての詳細なヘルプを表示します。

remove

ovmtdeploy -U コマンドを実行します。コマンドオプションについては、[133 ページの ovmtdeploy\(1M\)](#) のマニュアルページを参照してください。

次の終了値が返されます。

0 正常に完了しました。

1 エラーが発生しました。

2 使用法エラーが発生しました。

99 コマンドファイルの実行でエラー。

例 33 テンプレートの配備

次の例は、`ovmtadm deploy` コマンドを使用して、`ldg1` ドメインに `./ovmtutils_data/small-1.ova` テンプレートを配備する方法を示しています。

```
primary# /opt/ovmtutils/bin/ovmtadm deploy -d ldg1 ./ovmtutils_data/small-1.ova
executing wrapped command: /opt/ovmtutils/bin/ovmtdeploy -d ldg1 ./ovmtutils_data/
small-1.ova
```

```
Oracle VM for SPARC Deployment Utility
ovmtdeploy Version 3.4.0.0.13.1910
Copyright (c) 2014, 2015, Oracle and/or its affiliates. All rights reserved.
```

```
STAGE 1 - EXAMINING SYSTEM AND ENVIRONMENT
-----
```

```
Checking user privilege
Performing platform & prerequisite checks
Checking for required services
Named resources available
```

```
STAGE 2 - ANALYZING ARCHIVE & RESOURCE REQUIREMENTS
-----
```

```
Checking .ova format and contents
Validating archive configuration
Checking sufficient resources present
WARNING: Virtual switch primary-vsw0 already exists
```

```
STAGE 3 - EXTRACTING ARCHIVE
-----
```

```
Extracting archive
Validating checksums
Decompressing disk image(s)
```

```
STAGE 4 - DEPLOYING DOMAIN
-----
```

```
Creating domain and adding resources
Validating deployment
```

例 34 テンプレートの作成

次の例は、`ovmtadm create` コマンドを使用して、`ldg1` ドメインから `./test.ova` テンプレートを作成する方法を示しています。

```
primary# /opt/ovmtutils/bin/ovmtadm create -d ldg1 -o ./test.ova -f -n
executing wrapped command: /opt/ovmtutils/bin/ovmtcreate -d ldg1 -o ./test.ova -f -n
```

```
Oracle VM for SPARC Template Creation Utility
```

ovmcreate Version: 3.4.0.0.13.1910
Copyright (c) 2014, 2015, Oracle and/or its affiliates. All rights reserved.

STAGE 1 - EXAMINING SYSTEM AND ENVIRONMENT

Performing platform & prerequisite checks
Checking user permissions
Checking for required packages
Checking for required services
Checking directory permissions

STAGE 2 - ANALYZING DOMAIN

Retrieving and processing attributes
Checking domain state
Getting domain resource settings
Discovering network topology
Discovering disk topology

STAGE 3 - ARCHIVE CREATION

Checking destination and current directory capacity
Compressing disk image
Creating XML configuration
Calculating manifest checksums
Creating archive file
Checking archive

PROCESS COMPLETED

Started: Fri Jan 15 10:40:53 PST 2016
Completed: Fri Jan 15 10:40:56 PST 2016
Elapsed time: 0:00:03

例 35 ドメインの構成

次の例は、ovmtadm config コマンドを使用して、com.oracle.solaris.system.computer-name プロパティの値を system1 に設定することによって、ldg1 ドメインを構成する方法を示しています。

```
primary# /opt/ovmtutils/bin/ovmtadm config -d ldg1 -v -p com.oracle.solaris.system.computer-  
name=system1  
executing wrapped command: /opt/ovmtutils/bin/ovmtconfig -d ldg1 -v -p  
com.oracle.solaris.system.computer-name=system1
```

Oracle VM for SPARC Configuration Utility
ovmtconfig Version: 3.4.0.0.13.1910
Copyright (c) 2014, 2015, Oracle and/or its affiliates. All rights reserved.

STAGE 1/7 - EXAMINING SYSTEM AND ENVIRONMENT

Checking operating system
Checking platform type
Checking user permissions
Checking packages
Checking host domain name
Checking host domain type
Checking services

STAGE 2/7 - PROCESSING COMMAND LINE OPTIONS

Parsing individual properties
Creating consolidated properties list

STAGE 3/7 - ANALYZING TARGET DOMAIN

STAGE 4/4 - SETTING TARGET DOMAIN ENVIRONMENT

Checking 1 properties to set as domain variables
Process completed

例 36 ドメインの削除

次の例は、`ovmtadm remove` コマンドを使用して、`ldg1` ドメインをアンインストールおよび削除する方法を示しています。

```
primary# /opt/ovmtutils/bin/ovmtadm remove ldg1
executing wrapped command: /opt/ovmtutils/bin/ovmtdeploy -U ldg1

Oracle VM for SPARC Deployment Utility
ovmtdeploy Version 3.4.0.0.13.1910
Copyright (c) 2014, 2015, Oracle and/or its affiliates. All rights reserved.
```

STAGE 1 - EXAMINING SYSTEM AND ENVIRONMENT

Checking user privilege
Performing platform & prerequisite checks

STAGE 2 - UNINSTALL DOMAIN

Stopping and unbinding domain
Removing virtual disks
Removing backend devices
Removing domain files
Uninstall completed

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	<code>pkg:/system/ldoms/ovmtutils</code>
インタフェースの安定性	不確実

[123 ページの ovmtconfig\(1M\)](#)、[129 ページの ovmtcreate\(1M\)](#)、
[133 ページの ovmtdeploy\(1M\)](#)、[139 ページの ovmtlibrary\(1M\)](#)、
[143 ページの ovmtprop\(1M\)](#)、[attributes\(5\)](#)

『Oracle VM Server for SPARC 3.4 開発者ガイド』

名前

ovmtconfig — Oracle VM for SPARC テンプレートの構成

```
/opt/ovmtutils/bin/ovmtconfig -d domain-name [-b] [-c command,command] [-f] [-h|H] [-n]
[-p name=value,name=value | -P prop-filename,prop-filename] [-q] [-s] [-v] [-Z]
```

ovmtconfig コマンドは、事前に定義された 1 つ以上のアクションを論理ドメインのファイルシステム上で実行できます。

このコマンドは、構成するドメインが含まれるシステムの制御ドメイン上で実行します。現在、ovmtconfig コマンドは内部フォーマット zpool、ZFS ファイルシステム、および単一のブート環境で、仮想ディスクが存在するドメインでのみ動作します。

1 回につき 1 つのドメインのみを構成できます。

このコマンドはスーパーユーザーとして実行する必要があります。

ovmtconfig コマンドには次のオプションがあります。

-b

OpenBoot auto-boot? プロパティ値を true に設定します。

-c *command,command*

実行する 1 つ以上のコマンドを指定します。*command* は、制御ドメインから認識できるファイルシステム上で見つかった有効な実行可能ファイルへのフルパスにできます。ただし、*command* をターゲットドメインのファイルシステムにすることはできません。引用符内の任意のコマンドオプションをカプセル化します。

指定するコマンドはコマンド行で指定された順序で実行されます。

指定したコマンドが実行されると、*command* にコマンド行引数が含まれるかどうかに応じて、次のプロパティファイルが更新されます。

- **コマンド行オプション。** ターゲットドメイン内のファイルシステムのマウントポイントで一時プロパティファイルを更新します。
- **コマンド行オプションなし。** マウントポイントでマスタープロパティファイルを更新します。

command が 0 以外の値を返す場合、ovmtconfig コマンドはエラーレベル 99 で終了します。プロパティファイルに対する更新は削除されます。

scripts ディレクトリ内にある次のスクリプトは、基本的な Oracle Solaris 11 および Oracle Solaris 10 ドメインを構成できます。これらのスクリプトは、サンプルの solaris.properties ファイル、およびコマンド行で指定された追加のプロパティとともに使用されます。

ovmt_s10_sysidcfg.sh

sysidconfig プロパティ値に基づいて Oracle Solaris 10 ドメインを構成します。

ovmt_s10_interactive.sh

ブート時にデフォルトの Oracle Solaris 10 対話型インストーラを使用して、OS をインストールします。

ovmt_s11_scprofile.sh

sc_profile.xml プロパティ値に基づいて Oracle Solaris 11 ドメインを構成します。

ovmt_s11_interactive.sh

ブート時にデフォルトの Oracle Solaris 11 対話型インストーラを使用して、OS をインストールします。

-d domain-name

現在のシステムに存在するドメインの名前を指定します。ドメインの状態は、非アクティブである必要があります。ただし、ディスクイメージなどすべてのリソースが存在し、ドメイン制約の一部である必要があります。実行中状態のドメインに **-f** オプションを使用する場合は注意してください。**-f** オプションの方法は、ファイルシステムの破損につながる可能性があります。



注意 - ovmtconfig は操作中に停止または中断しないでください。バックマウントされたドメインは自動的にロールバックされません。別の端末ウィンドウまたはセッションを使用して、すべての手動アクションを実行します。

-f

実行中のドメインを強制的に停止します。

-h または **-H**

要約されたヘルプファイルを表示します。

-n

予行演習を行います。これにより、ドメインの停止とバインド解除およびファイルシステムのバックマウントを含むすべての操作ステップが実行されます。この場合、実行されるコマンドと引数、および一時プロパティファイルの一覧がメッセージに表示されます。

-p name=value, name=value

名前と値のペアでプロパティのコンマ区切りリストを指定します。指定したプロパティは、一時マスタープロパティファイルでその他のプロパティと結合されます。名前と値のペアは順番に解析され、提供されるプロパティファイ

ル内の同一のプロパティ名よりも優先順位が高くなります。プロパティの名前と値に空白またはエスケープ文字を含めることはできません。

複雑なプロパティの例は次のようになります。

```
domain.group.property.instance=value
```

-P *prop-filename, prop-filename*

名前と値のペア形式のプロパティが含まれる標準テキストプロパティファイルのコンマ区切りのリストを指定します。ファイル内のプロパティは1行に1つずつ表示されます。プロパティファイルは順番に解析されます。プロパティがあとのプロパティファイルで重複している場合、前のプロパティファイルで指定された値は上書きされます。プロパティのファイル形式が正しくない場合、`ovmtconfig` はエラーレベル 2 を返します。

コマンド行で指定された名前と値のペアは、構成ファイル内の名前と値のペアよりも優先されます。

`props/solaris.properties` ファイルには、Oracle Solaris 11 および Oracle Solaris 10 ドメインを構成するために使用できるプロパティが記述されます。

-q

エラーおよびエラーレベルのみを報告します。

-s

バックマウント、コマンド実行、およびロールバックが正常に完了したあと、ドメインが自動的に起動します。`ovmtconfig` コマンドは、コマンドの実行時にエラーが発生した場合、ドメインの起動を試みません。`-s` オプションが指定されない場合、`ovmtconfig` はドメインを元の状態に戻します。

-v

次の形式で連結されたプロパティから論理ドメインのプロパティを作成します。

```
com.oracle.solaris.fmri.count=0
com.oracle.solaris.system.computer-name=solaris
```

-z

`ovmtconfig` コマンドを一時停止して、コマンドが指定されない場合であってもバックマウント後に手動アクションを実行できるようにします。このオプションは、一時ルートマウントポイント、および一時プロパティファイルのパス (該当する場合) を表示します。

例 37 ドメインでの手動アクションの実行

次のコマンドは、手動アクションを実行できるように `ldg0` ドメインをバックマウントします。`-z` オプションは、バックマウント時に手動アクションを実行できるようにコマンドを一時停止します。

```
primary# /opt/ovmtutils/bin/ovmtconfig -d ldg0 -z
```

例 38 単一コマンドの実行とプロパティの設定

次のコマンドは `ldg0` ドメインをバックマウントして単一コマンド `command.sh` を実行し、`com.oracle.solaris.system.computer-name` プロパティを `solaris` に設定します。

```
primary# /opt/ovmtutils/bin/ovmtconfig -d ldg0 -p com.oracle.solaris.system.computer-name=solaris \  
-c command.sh
```

例 39 構成ファイルを使用したドメインの構成

次のコマンドは、サンプルのスクリプト、プロパティ、およびプロパティファイルを使用して、`ldg0` ドメインを構成します。ドメインを強制的に停止およびバインド解除してから、完了時に再起動します。

```
primary# /opt/ovmtutils/bin/ovmtconfig -d ldg0 \  
-p com.oracle.solaris.system.computer-name=solaris, \  
com.oracle.solaris.network.ipaddr.0=10.0.0.100 \  
-P solaris.props,apps.props -c path-to-utils/scripts/ovmt_s11_scprofile.sh -s -f
```

例 40 変数の設定、およびドメインの停止と再起動

次のコマンドは、`ldg0` ドメインの論理ドメイン変数を設定します。この `ovmtconfig` コマンドは任意のコマンドを実行しませんが、操作の完了後にドメインを強制的に停止および再起動します。ドメインはバックマウントされません。

```
primary# /opt/ovmtutils/bin/ovmtconfig -d ldg0 -v -s -f
```

次の終了値が返されます。

- 0 正常に完了しました。
- 1 エラーが発生しました。
- 2 使用法エラーが発生しました。
- 99 コマンドファイルの実行でエラー。

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	pkg:/system/ldoms/ovmtutils
インタフェースの安定性	不確実

[119 ページのovmtadm\(1M\)](#)、[123 ページのovmtconfig\(1M\)](#)、
[129 ページのovmtcreate\(1M\)](#)、[133 ページのovmtdeploy\(1M\)](#)、
[139 ページのovmtlibrary\(1M\)](#)、[143 ページのovmtprop\(1M\)](#)、[attributes\(5\)](#)

『Oracle VM Server for SPARC 3.4 開発者ガイド』

名前

ovmtcreate — create Oracle VM for SPARC テンプレート

```
/opt/ovmtutils/bin/ovmtcreate -d domain-name -o template-name [-c class-name] [-C] [-f]
[-h | -H] [-m packaging-mode] [-n] [-P filename[, filename]] [-q] [-s description]
[-S] [-v version] [-V major-version] [-w directory]
```

ovmtcreate コマンドは、論理ドメインのトポロジを検査し、OVF アーカイブ (.ova) の形式でテンプレートを作成します。

このコマンドは、テンプレートを作成するドメインがあるシステムの制御ドメイン上でスーパーユーザーとして実行します。

このコマンドは、存在するデバイスに基づいて、仮想ディスクファイルがあるドメインからテンプレートを作成します。仮想ディスクファイルには任意の名前を付けることができます。ファイル名拡張子は自由に付けることができ、使用しなくても構いません。

ovmtcreate コマンドでは、ドメインの名前および作成するテンプレートの名前を次のように指定する必要があります。

-d domain-name 現在のシステムに存在するドメインの名前。ドメインの状態は、バインドされていて非アクティブである必要があります。ディスクイメージなどすべてのリソースが存在し、ドメイン制約の一部である必要があります。

-o template-name テンプレートのフルパス名を指定します。
テンプレートを作成するディレクトリには、ドメインに必要とされる圧縮されたディスクイメージおよびサポートするファイルをすべて格納できる十分なディスク容量が必要です。template-name は、OVF 仕様ファイル内の情報に使用されるとともに、テンプレート名の一部となります。

ovmtcreate コマンドには次のオプションがあります。

-c class-name 生成されるテンプレートのソフトウェア製品のクラス名を指定します。class-name は、OVF ファイルプロパティを設定するために使用されます。

-C 生成されたテンプレートの読み取りができることを検証します。このオプションは、テンプレートの内容またはメタデータは検証しません。

-
- f 十分なディスク領域があるかどうかなどのチェックは実行せずに、操作を強制的に完了します。
- h 要約されたヘルプファイルを表示します。ヘルプファイル全体を表示するときは、-H オプションを指定します。
- m *packaging-mode* パッケージングモードを指定します。有効な値は `openstack` ディスクイメージ形式またはデフォルトの `ovf` テンプレート形式です。
- ソースドメイン内の最初の仮想ディスクから直接、単一の圧縮されていない SPARC OpenStack と互換性のあるディスクイメージを作成するには、`ovmtcreate -m openstack` コマンドを使用します。このコマンドは、追加のディスクイメージ、OVF メタデータファイル、およびマニフェストファイルなどの追加のペイロード項目を含む完全なテンプレートを作成しません。また、このコマンドは、`.ova tar` ファイルにこれらのコンポーネントをカプセル化しません。説明を提供する、ボイラープレートファイルを指定する、またはマイナーおよびメジャーバージョンを指定するなどのほかのメタデータオプションは無視されます。
- `ovmtcreate -m ovf` コマンドは、完全な OVF テンプレートを作成しますが、これは `-m ovf` オプションを使用せずに `ovmtcreate` コマンドを実行する場合と同じです。
- n 予行演習を行います。これにより、テンプレートの作成を含むすべての操作ステップが実行されます。予行演習では、実際のディスクファイルではなく小さいブレースホルダイメージが使用されます。このオプションは操作を迅速化しますが、OVF プロパティなど生成されるテンプレートは引き続き点検できます。予行演習により有効なテンプレートが生成されるため、このオプションを使用してテンプレートを Oracle VM Manager などほかのアプリケーションに (クローンではなく) インポートできます。
- P *filename*[,
filename] テンプレートによって使用される構成可能なプロパティの一覧を提供する、1 つ以上の XML フラグメントファイルを指定します。指定したファイルは、テンプレートの作成プロセス中にテンプレート OVF ファイルへ結合されます。
- テンプレートを適切に構成するために必要なプロパティを指定するために、これらのファイルを使用できます。
- テンプレートに含まれるプロパティを一覧表示する方法については、`ovmtdeploy(1M)` のマニュアルページを参照してください。
- 通常、テンプレートで使用されるオペレーティングシステムおよびアプリケーションごとに別々のファイルを管理します。アプリケーション構成パラメータが変更されたときは、更新されたアプ

リケーションがテンプレートに反映されるように、ファイルを更新します。

Oracle Solaris 11 OS の XML 形式によるポイラプレートフラグメントについては、`ovmtutils/props` ディレクトリを参照してください。

- `-q` エラーおよびエラーレベルのみを報告します。
- `-s description` OVF ファイルプロパティを設定するために使用される、テンプレートの説明を指定します。
- `-S` テンプレートが作成されたあとで、ドメインのサマリーを提供します。
- `-v version` 生成されるテンプレートに含まれるソフトウェア製品のバージョンを指定します。`version` は、OVF ファイルプロパティを設定するために使用されます。
- `-V major-version` 生成されるテンプレートに含まれるソフトウェア製品のメジャーバージョンを指定します。`major-version` は、OVF ファイルプロパティを設定するために使用されます。
- `-w directory` ファイルを圧縮したり作業データを作成したりするディレクトリとして `directory` を指定します。デフォルトで、`ovmtcreate` コマンドは現在の作業ディレクトリを使用します。

このオプションを使用すると、ディスクファイルの元の場所からコマンドを実行した場合は作成後に手動で圧縮解除する必要があるため、問題が発生することがあります。この問題を回避するには、現在の作業ディレクトリ以外のディレクトリを指定します。`directory` には、圧縮解除されたディスクイメージおよびサポートファイルのすべてを格納できる十分なディスク領域が必要です。

例 41 ドメインの作成

次のコマンドは、`ovmtcreate` コマンドと `ovmtlibrary` コマンドの両方で使用できるディスクファイルを使用してドメイン `ldg0` を作成する方法を示しています。

この例では、基本的な Oracle VM Server for SPARC 仮想スイッチおよびディスクサーバーサービスが使用可能であることを前提としています。

```
primary# ldm create ldg0
primary# ldm add-vcpu 4 ldg0
primary# ldm add-mem 2048m ldg0
primary# mkdir -p /domains/ldg0
```

```

primary# mkfile 10G /domains/ldg0/System.img
primary# ldm add-vdsdev /domains/ldg0/System.img ldg0-vol0@primary-vds0
primary# ldm add-vdisk vdisk0 ldg0-vol0@primary-vds0 ldg0
primary# ldm add-vnet vnet0 primary-vsw0 ldg0
primary# ldm bind ldg0
primary# ldm start ldg0

```

例 42 最小限のプロパティを使用したテンプレートの作成

この例は、ldg1 ドメインの ~/templates ディレクトリにアーカイブを作成する方法を示しています。

```
primary# /opt/ovmtutils/bin/ovmtcreate -d ldg1 -o ~/templates
```

例 43 製品プロパティを指定するテンプレートの作成

この例は、ldg0 ドメインの ~/templates ディレクトリにアーカイブを作成する方法を示しています。-c、-n、-P、-s、-v、および -V オプション引数は、プロパティ値を指定します。

```
primary# /opt/ovmtutils/bin/ovmtcreate -d ldg0 -o ~/templates -n OVM_SOL11_SPARC \
-s "Oracle Solaris 11 for SPARC" -v 11 -V 5.11 -c com.oracle.solaris \
-P path-to-ovmtutils/props/solaris_11.boilerplate
```

次の終了値が返されます。

- 0 正常に完了しました。
- 1 エラーが発生しました。プロパティ値が設定されていないか、プロパティが見つかりませんでした。
- 2 使用法エラーが発生しました。

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	pkg:/system/ldoms/ovmtutils
インタフェースの安定性	不確実

[119 ページのovmtadm\(1M\)](#)、[123 ページのovmtconfig\(1M\)](#)、[133 ページのovmtdeploy\(1M\)](#)、[139 ページのovmtlibrary\(1M\)](#)、[143 ページのovmtprop\(1M\)](#)、[attributes\(5\)](#)

『Oracle VM Server for SPARC 3.4 開発者ガイド』

名前

ovmtdeploy — deploy Oracle VM for SPARC テンプレート

```
/opt/ovmtutils/bin/ovmtdeploy -d domain-name[,domain-name]... [-b] [-C core[,core]...]
[-c vcpu[,vcpu]...] [-D] [-E vnet[:option=value][,vnet[:option=value]]...]
[-e nic[,nic]...] [-I bus[,bus]...] [-k] [-l]
[-M start-addr:size[,start-addr:size]...] [-m size[,size]...] [-N number-of-domains] [-n]
[-o directory] [-p] [-P] [-q] [-S] [-s] [-t vsw[:option=value][,vsw[:option=value]]...]
[-v dev:option=value[,dev]...] -x template-name

/opt/ovmtutils/bin/ovmtdeploy [-q] -U domain-name
```

ovmtdeploy コマンドは、テンプレートの内容を読み取って完全に機能する 1 つ以上の論理ドメインを作成します。

このコマンドは十分な領域があって root により書き込み可能であるディレクトリからスーパーユーザーとして実行する必要があります。

-d オプションを使用して作成する論理ドメインの名前を指定する必要があります。デフォルトでは、名前はテンプレートの .ovf ファイル名から取得されます。デフォルトをオーバーライドする追加の名前を指定できます。テンプレートの複数のコピーを指定するには、-N オプションを使用します。テンプレートの複数のコピーが存在する場合は、それぞれの名前のベース名に整数の接尾辞が追加されます。たとえば、OVM_SOL11_SPARC は OVM_SOL11_SPARC_1 になります。

また、テンプレート (.ova) のパス *template-name* を指定する必要もあります。このファイルは、SPARC システムで使用するようフォーマットされたディスクイメージが含まれる OVF 1.0 アーカイブにする必要があります。

ovmtdeploy コマンドは、テンプレートの配備の前に、配備先デバイスに十分なスペースがあることを確認します。このコマンドは、使用可能な領域が十分にない場合は、エラーで終了します。

ovmtdeploy コマンドには次のオプションがあります。

-b

バインドステップをスキップします。

-c vcpu[,vcpu]...

テンプレートファイル (.ovf) で指定された仮想 CPU の数の設定をオーバーライドします。オーバーライドする仮想 CPU の数は、論理ドメインの数に一致する必要があります。

-C *core-ID*[,*core-ID*]...

指定された名前付きコアをドメインに割り当てます。*core-ID*は、数値または範囲にできます。たとえば、**-C 4,5,8-16** です。

-C オプションは、**-c** および **-N** オプションと相互排他です。

-e *nic*[,*nic*]...

このテンプレートに必要な仮想ネットワークスイッチで使用するネットワークアダプタを指定します。このオプションは、ゲストドメインネットワークインタフェースに接続するために仮想スイッチで使用されるプライマリドメインインタフェースを決定します。

-t オプションを使用すると、ドメインが持つインタフェースより多くのインタフェースをオーバーライドおよび割り当てできます。

システムに存在しない仮想スイッチを指定するために **-t** オプションを使用する場合、または制御ドメインに存在しない仮想スイッチをテンプレートで記述する場合、**-e** オプションを使用する必要があります。

[-E *vnet*[:*option=value*][,*vnet*[:*option=value*]]. . .]

.ovf ファイルに指定された仮想ネットワークインタフェースエントリをオーバーライドします。

option には次のデバイスオプションのいずれかを指定できます。

- **pvid=port-VLAN-ID** 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。
- **vid=VLAN-ID** は、仮想ネットワークデバイス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。

-h

使用方法に関するメッセージを表示します。**-H** オプションと同じです。

-I *bus*[,*bus*]...

指定されたバス、PCIe デバイス、または SR-IOV 仮想機能をドメインに追加します。

-I オプションは **-v**、**-C**、**-M**、および **-N** オプションと相互に排他的です。

-k

チェックサム検証をスキップします。テンプレートをはじめてデプロイする場合、またはテンプレートが不明なソースに由来する場合は、このオプションを使用しないでください。テンプレートの整合性をすでにチェックしていて、まだ管理下にある場合は、あとでそのテンプレートをデプロイするときにこのオプションを使用できます。

-l

デプロイせずにテンプレートの内容を一覧表示します。このオプションは、テンプレートに含まれるドメインの内容とトポロジの両方の説明を提供します。

-m *size*[,*size*]...

テンプレートファイル (.ovf) で指定されたメモリーサイズをオーバーライドします。オーバーライドするサイズの数、論理ドメインの数に一致する必要があります。*size* は、メモリーの量に単位を足したものです。単位は、M バイトの場合は m、G バイトの場合は g です。

-M *start-addr:size*[,*start-addr:size*]...

指定された名前付きメモリーブロックをドメインに割り当てます。*start-addr* は 16 進数または 10 進数のメモリーブロックの開始アドレスです。*start-addr* が 16 進数の場合、アドレスには 0x 接頭辞を使用してください。*start-addr* が 10 進数の場合、アドレスのあとに G を付けてください。*size* はメモリーの量で、あとに G を付けます。-M 128G:8G や -M 0x800000000000:20G などです。

-M オプションは、-m および -N オプションと相互排他です。

-n

デプロイ、ロールバック、またはアンインストールコマンドを一覧表示して、予行演習を行います。

-N *num-of-ldoms*

複数の論理ドメイン (*num-of-ldoms*) を単一テンプレートから作成します。この場合、ドメイン名には 0 から増分される整数の接尾辞が含まれます。この名前は、デプロイディレクトリでも使用されます (たとえば *dest-dir/OVM_SOL11_SPARC_1*)。

-N オプションは -v、-C、-I、および -M オプションと相互に排他的です。

-o *dest-dir*

テンプレートコンポーネントをコピーする UFS または ZFS ファイルシステムディレクトリのフルパスを指定します。このディレクトリは、存在していて、書き込み可能であり、十分な空き領域があることが必要です。このディレクトリが指定されない場合、現在の作業ディレクトリが使用されます。テンプレート名のデプロイディレクトリは、次のディレクトリの下に作成されます。たとえば、*dest-dir/OVM_SOL11_SPARC* です。

-p

-l オプションと同じで、完全修飾テンプレートプロパティ名を出力します。

-P

-p オプションと同じで、テンプレートプロパティの説明、タイプ、値の情報を出力します。

-q

出力を現在の作業ディレクトリ内のファイルにリダイレクトします。

-s

開始ステップをスキップします。

-S

正常にデプロイしたあとで、ドメイン構成ファイルを SP に保存します。

`[-t vsw[:option=value][,vsw[:option=value]]...]`

.ovf ファイルで指定された仮想スイッチエントリをオーバーライドします。

option には次のデバイスオプションのいずれかを指定できます。

- `pvid=port-VLAN-ID` 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。
- `vid=VLAN-ID` は、仮想ネットワークデバイス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。

-U *domain-name*

指定されたドメインのアンインストールコマンドを実行します。

-v *dev*

ターゲットデバイス、ディレクトリ、またはその両方のコンマ区切りリストを指定します。次のターゲットデバイスを指定できます。

- Raw ディスクデバイス (`/dev/rdisk/c3t3d0s2` など)
- iSCSI デバイス (`/dev/rdisk/c0t600144F00021283C1D7A53609BE10001d0s2` など)
- ZFS ボリューム (`/dev/zvol/rdisk/ovmt/domain` など)
- ディレクトリ

スライス上のターゲットデバイスおよびブロック型デバイス上のターゲットデバイスはサポートされません。

デバイス名 *dev* のあとには、次のプロパティをコンマ区切りリストで指定できます。

- マルチパスグループ: `mpgroup=group-name`
- 仮想ディスクタイムアウト: `timeout=seconds`
- 関連ディスクオプションの指定: `option=option` (`opt=option` でも可)
- *dev* の代替デバイスパスおよび仮想ディスクサービス (VDS) の指定:
`alt=path@VDS`。 `ldm add-vdsdev -q` コマンドを使用すると、デバイスが検証なしでドメインの `mpgroup` に追加されます。VDS が存在する必要もあります。`mpgroup` プロパティを指定した場合は、`alt` プロパティのみを指定します。

たとえば、`-v /dev/rdisk/c3t3d0s2:mpgroup=mp1:timeout=30:option=excl` です。

デバイスの数は、テンプレート内の仮想ディスクの数に一致する必要があります。

ディスクイメージを追加できます。この場合、`-v` を使用して指定するターゲットデバイスの数は、テンプレート内で指定されたディスクイメージの数よりも多い必要があります。また、スパーファイルとして作成される、ディスク上のファイルのサイズを指定する必要があります。

たとえば、`-v /dev/rdisk/c3t2d0s2,/ovmt/NewDisk.img:size=16G` です。

`-v` オプションは、`-N` オプションと相互排他です。

`-x`

テンプレートのデプロイ時に基盤となるディスクをデバイスエクステンツに展開しません。デフォルトで、デプロイプロセスは、指定されたデバイスまたはバックエンドデバイスのパーティションをエクステンツに展開します。

例 44 デプロイせずにテンプレートの内容を一覧表示

この例は、`OVM_SOL11_SPARC.ova` テンプレートの内容を一覧表示する方法を示しています。このコマンドはデプロイを実行しません。

```
primary# /opt/ovmtutils/bin/ovmtdeploy -l OVM_SOL11_SPARC.ova
```

例 45 単一論理ドメインへのデプロイ

この例は、`$HOME/ovmt` ディレクトリ内の `OVM_SOL11_SPARC.ova` テンプレートを `ldg1` ドメインにデプロイする方法を示しています。

```
primary# /opt/ovmtutils/bin/ovmtdeploy -d ldg1 -o ~/ovmt OVM_SOL11_SPARC.ova
```

例 46 複数のドメインへのデプロイと設定のオーバーライド

この例は、`$HOME/ovmt/OVMSOL11_SPARC.ova` テンプレートを使用して 2 つのドメインにデプロイする方法を示しています。指定されたコマンド行オプションは、テンプレート内のメモリおよびコア設定をオーバーライドして、8 および 16 コアを使用し、それぞれ 8G バイトのメモリと 16G バイトのメモリを使用します。

```
primary# /opt/ovmtutils/bin/ovmtdeploy -N 2 -c 8,16 -m 8g,16g -o ~/ovmt OVM_SOL11_SPARC.ova
```

例 47 デプロイ済みドメインのアンインストール

この例は、`OVM_SOL11_SPARC.ova` テンプレートに基づいて `OVM_SOL11_SPARC_1` をアンインストールする方法を示します。

```
primary# /opt/ovmtutils/bin/ovmtdeploy -U OVM_SOL11_SPARC_1 OVM_SOL11_SPARC.ova
```

次の終了値が返されます。

-
- 0 正常にデプロイされました。
 - 1 コマンドの前提条件が満たされていません。
 - 2 ランタイムエラーが発生しました。

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	pkg:/system/ldoms/ovmtutils
インタフェースの安定性	不確実

[119 ページのovmtadm\(1M\)](#)、[123 ページのovmtconfig\(1M\)](#)、
[129 ページのovmtcreate\(1M\)](#)、[139 ページのovmtlibrary\(1M\)](#)、
[143 ページのovmtprop\(1M\)](#)、[attributes\(5\)](#)

『Oracle VM Server for SPARC 3.4 開発者ガイド』

名前

ovmtlibrary — Oracle VM for SPARC テンプレート ライブラリの管理

```
/opt/ovmtutils/bin/ovmtlibrary [-V] [-h | -H]
/opt/ovmtutils/bin/ovmtlibrary -c init [-f] [-q | -v] -l library
/opt/ovmtutils/bin/ovmtlibrary -c store [-d description] [-f] -l library [-n template-name]
-o URI [-q | -v]
/opt/ovmtutils/bin/ovmtlibrary -c list [-a | -p | -o | -t type] -l library [-m]
[-n template-name | -i template-id] [-q | -v] [-s [-e event-id]]
/opt/ovmtutils/bin/ovmtlibrary -c delete [-a] -l library [-n template-name | -i template-id]
[-q | -v]
```

ovmtlibrary コマンドは、次の方法で Oracle VM for SPARC テンプレート 情報を管理できます。

- `-c init` は、ライブラリを初期化します
- `-c store` は、テンプレートを格納します
- `-c list` は、テンプレートを一覧表示します
- `-c delete` はテンプレートを削除します

テンプレートを格納するために使用されるテンプレートアーカイブファイルには、次のものがが必要です。

- ファイル拡張子 `.ova`
- ファイル拡張子 `.ovf` を持つ XML 構成ファイル
- `.ovf` ファイル内の情報に一致する 1 つ以上の圧縮されたディスクイメージ
- `.mf` ファイルのファイル拡張子を持つチェックサム情報ファイル

テンプレートを格納するときにイベント ID が出力されます。この ID を使用して格納操作のステータスをチェックできます。

ovmtlibrary コマンドを実行するためにスーパーユーザー権限は必要ありません。

ovmtlibrary コマンドには次のオプションが含まれます。

- | | |
|----------------------|---|
| <code>-a</code> | 指定されたテンプレートのすべてのバージョンを一覧表示します。デフォルトでは、最新バージョンのみが表示されます。 |
| <code>-c init</code> | テンプレートライブラリを初期化します。 |

<code>-c store</code>	ライブラリ内にテンプレートを格納します。
<code>-c list</code>	ライブラリ内のテンプレートを一覧表示します。
<code>-c delete</code>	ライブラリからテンプレートを削除します。
<code>-d description</code>	テンプレートの説明を指定します。
<code>-e event-ID</code>	格納操作のステータスを表示するイベント ID を指定します。
<code>-f</code>	既存のライブラリの上書きを強制的に実行します。
<code>-h</code>	ヘルプメッセージを表示します。
<code>-i template-ID</code>	一覧表示するテンプレートの ID を指定します。
<code>-l library</code>	テンプレートライブラリのパスを指定します。
<code>-m</code>	機械解析可能な形式で出力を生成します。
<code>-n template-name</code>	テンプレートの名前を指定します。 格納操作では、このオプションは格納するテンプレート名を指定します。デフォルトでは、OVF 名はこのオプションが指定されていない場合に使用されます。 一覧表示操作では、このオプションは一覧表示するテンプレート名を指定します。デフォルトでは、すべてのテンプレートの最新バージョンが表示されます。
<code>-o URI</code>	格納するテンプレートオブジェクトの URI を指定します。有効な URI の値は、file://、http://、および ftp:// を含みます。
<code>-o</code>	指定されたテンプレートのオブジェクトを一覧表示します。テンプレート名またはテンプレート ID を指定する必要があります。テンプレート名を指定した場合、その最新バージョン内のオブジェクトのみが一覧表示されます。
<code>-p</code>	-i オプションを使用して指定したテンプレート ID に対するプロパティ情報を表示します。
<code>-q</code>	ヘッダーとメッセージを抑制します。
<code>-s</code>	イベントステータスを表示します。

-
- t *type* 指定されたテンプレートのオブジェクトタイプを指定します。関連するテンプレート名またはテンプレートIDを指定する必要もあります。
 - v 詳細情報を表示します。
 - V このコマンドに関するバージョン情報を表示します。

次の終了値が返されます。

- 0 正常に完了しました。
- 1 エラーが発生しました。プロパティ値が設定されていないか、プロパティが見つかりませんでした。
- 2 使用法エラーが発生しました。

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	pkg:/system/ldoms/ovmtutils
インタフェースの安定性	不確実

[119 ページのovmtadm\(1M\)](#)、[123 ページのovmtconfig\(1M\)](#)、[129 ページのovmtcreate\(1M\)](#)、[133 ページのovmtdeploy\(1M\)](#)、[attributes\(5\)](#)

『Oracle VM Server for SPARC 3.4 開発者ガイド』

名前

ovmtprop — Oracle VM for SPARC テンプレート プロパティの表示

```
/opt/ovmtutils/bin/ovmtprop get-prop -a [-h] [-q] [-V]
/opt/ovmtutils/bin/ovmtprop get-prop -k name[,name,...] [-h] [-q] [-V]
/opt/ovmtutils/bin/ovmtprop get-prop -K name[,name,...] [-h] [-q] [-V]
/opt/ovmtutils/bin/ovmtprop set-prop prop-name=prop-value,... [-h] [-q] [-V] domain-name
```

ovmtprop コマンドでは、Oracle Solaris OS プロパティを表示および設定できます。プロパティと値は、名前と値のペアとして指定されます。ovmtprop get-prop コマンドを使用してプロパティ値を表示し、ovmtprop set-prop および ovmtconfig -v コマンドを使用してプロパティを作成します。

ovmtprop get-prop コマンドと ovmtprop set-prop コマンドは同時に実行しないでください。

ドメインのプロパティ値を設定するには、制御ドメインで ovmtprop set-prop コマンドをスーパーユーザーとして実行します。アプリケーションのプロパティ間の衝突を避けるため、名前と値のペアを指定するときには完全修飾名前空間を使用することが最善です。

プロパティとその値を表示するには、プロパティ情報を取得するドメインで ovmtprop get-prop コマンドを実行します。プロパティ情報を表示するのにスーパーユーザーになる必要はありません。

すべてのプロパティを表示するには、-k または -K オプションを使用してプロパティ検索基準を指定するか、または引数なしで -a オプションを使用します。検索基準をドメイン別に指定すると、単一アプリケーションのすべてのプロパティを一覧表示するなど、階層プロパティを返すことができます。

ovmtprop コマンドには次のオプションが含まれます。

- a
すべてのプロパティの名前と値をそれぞれ別々の行に表示します。このオプションは -k オプションとともに使用できません。
- h
使用法に関する文を出力します。

-k name[,name,...] [all]

クエリーするプロパティ名のコマ区切りリストを指定します。プロパティ名とプロパティ値は、別々の行に表示されます。**-k all** オプションを指定することは、**-a** オプションを使用することと同じです。

-K name

クエリーにプロパティ名を指定し、プロパティ値のみを出力します。

-q

指定されたプロパティとプロパティ値のみをそれぞれ別々の行に表示します。プロパティが表示される順序は、コマンド行に指定したプロパティの順序に一致します。コピーライトバナーとエラー文字列は抑制されます。

-V

バージョンを出力します。

例 48 ゲストドメインの `com.oracle.solaris.system.computer-name` プロパティの設定

この例は、`ldg0` ドメインで `com.oracle.solaris.system.computer-name` プロパティを `solaris11` に設定する方法を示しています。

```
primary# /opt/ovmtutils/bin/ovmtprop set-prop com.oracle.solaris.system.computer-name=solaris11 ldg0
```

例 49 `com.oracle.solaris.system.computer-name` プロパティとプロパティ値の表示

この例は、`com.oracle.solaris.system.computer-name` プロパティの値を表示する方法を示しています。出力にはプロパティ名とプロパティ値の両方が表示されます。

```
$ /opt/ovmtutils/bin/ovmtprop get-prop -k com.oracle.solaris.system.computer-name
com.oracle.solaris.system.computer-name=s11
```

例 50 複数の階層プロパティの表示

この例は、`com.oracle.solaris` プロパティの下にあるプロパティとその値を表示する方法を示しています。

```
$ /opt/ovmtutils/bin/ovmtprop get-prop -k com.oracle.solaris
solaris.system.computer-name=s11
solaris.network.ipaddr.0=192.168.0.1
```

例 51 すべてのプロパティ値の取得

この例は、すべてのプロパティのプロパティ値を取得する方法を示しています。出力にはプロパティ名とプロパティ値の両方が表示されます。

```
$ /opt/ovmtutils/bin/ovmtprop get-prop -a
com.oracle.solaris.system.computer-name=s11
com.oracle.solaris.network.ipaddr.0=192.168.0.1
...
```

次の終了値が返されます。

- 0 正常に完了しました。
- 1 エラーが発生しました。プロパティ値が設定されていないか、プロパティが見つかりませんでした。
- 2 使用法エラーが発生しました。

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	pkg:/system/ldoms/ovmtutils
インタフェースの安定性	不確実

[119 ページのovmtadm\(1M\)](#)、[123 ページのovmtconfig\(1M\)](#)、
[129 ページのovmtcreate\(1M\)](#)、[133 ページのovmtdeploy\(1M\)](#)、
[139 ページのovmtlibrary\(1M\)](#)、[attributes\(5\)](#)

『Oracle VM Server for SPARC 3.4 開発者ガイド』

