



shaping tomorrow with you

# Oracle VM Server for SPARC Idm コマンド クイックリファレンス

2016年3月  
第1.2版  
富士通株式会社

## 本資料について

- 目的
  - 本リファレンスはOracle VM Server for SPARC 3.0 / 3.1で提供しているldmコマンドのうち、主に使われるサブコマンド、オプションの使い方について紹介しています。
  - 詳細については、Oracle社のドキュメントを参照してください。  
「Oracle VM Server for SPARC 3.1 リファレンスマニュアル」  
[http://docs.oracle.com/cd/E49214\\_01/index.html](http://docs.oracle.com/cd/E49214_01/index.html)
  - 「Oracle VM Server for SPARC 3.0 リファレンスマニュアル」  
[http://docs.oracle.com/cd/E38737\\_01/index.html](http://docs.oracle.com/cd/E38737_01/index.html)
- 留意事項
  - Oracle VM Server for SPARCで提供しているldmコマンドすべてのサブコマンド、オプションを紹介しておりません。すべての情報が必要な場合は、上記のOracle社のドキュメントをご参照ください。
  - Oracle VM Server for SPARCのバージョンにより、使用できるサブコマンドやオプションが異なります。ご注意ください。

## 使用条件

- 著作権・商標権・その他の知的財産権について  
コンテンツ(文書・画像・音声等)は、著作権・商標権・その他の知的財産権で保護されています。本コンテンツは、個人的に使用する範囲でプリントアウトまたはダウンロードできます。ただし、これ以外の利用(ご自分のページへの再利用や他のサーバへのアップロード等)については、当社または権利者の許諾が必要となります。
- 保証の制限  
本コンテンツについて、当社は、その正確性、商品性、ご利用目的への適合性等に関して保証するものではなく、そのご利用により生じた損害について、当社は法律上のいかなる責任も負いかねます。本コンテンツは、予告なく変更・廃止されることがあります。
- 輸出または提供  
本ドキュメントを輸出又は提供する場合は、外国為替及び外国貿易法及び米国輸出管理関連法規等の規制をご確認の上、必要な手続きをおとり下さい。

## 商標

- UNIXは、米国およびその他の国におけるオープン・グループの登録商標です。
- SPARC Enterprise, SPARC64, SPARC64ロゴ、およびすべてのSPARC商標は、米国SPARC International, Inc.のライセンスを受けて使用している、同社の米国およびその他の国における商標または登録商標です。
- OracleとJavaは、Oracle Corporation およびその子会社、関連会社の米国およびその他の国における登録商標です。
- その他各種製品名は、各社の製品名称、商標または登録商標です。

## ■ ldmコマンド クイックリファレンス 目次

<a href="#">1. domain(dom)</a>	ドメインの作成及び操作
<a href="#">2. vcpu</a>	仮想CPUの設定
<a href="#">3. core</a>	コアの設定
<a href="#">4. memory(mem)</a>	メモリの設定
<a href="#">5. spconfig(config)</a>	構成情報の追加、削除
<a href="#">6. hvdump</a>	ハイパーバイザのダンプデータ収集の設定
<a href="#">7. vconscn(vcc)</a>	仮想コンソールサービスの設定
<a href="#">8. vswitch(vsw)</a>	仮想スイッチサービスの追加、削除
<a href="#">9. vdiskserver(vds)</a>	仮想ディスクサービスの作成
<a href="#">10. vdiskserverdevice(vdsdev)</a>	仮想ディスクサービスデバイスの追加、削除
<a href="#">11. vdisk</a>	仮想ディスクの追加、削除
<a href="#">12. vnet</a>	仮想ネットワークの追加、削除
<a href="#">13. io</a>	物理I/Oデバイスの設定
<a href="#">14. vconsole(vcons)</a>	仮想コンソールのオプションの設定
<a href="#">15. variable(var)</a>	環境変数の設定
<a href="#">16. migrate</a>	ドメインの移行
<a href="#">17. operation(op)</a>	操作の取消
<a href="#">18. policy</a>	CPU Dynamic Resource Management の設定
<a href="#">19. list(ls)</a>	各情報のリスト表示
<a href="#">20. system</a>	ドメイン構成の復元
<a href="#">21. reconf</a>	遅延再構成モードの設定

- \* () 内はショートフォーマット、以降はショートフォーマットを使用  
 [] 内は選択項目(省略可能)、<> 内は必須項目、<a | b> はaまたはbのどちらかを指定  
 ★はSPARC M10専用サブコマンド・オプション

## 1. domain(dom)

add-domain (create)		ドメインの作成
形式	ldm create -i <file名> ldm create [オプション] <ドメイン名1,ドメイン名2, ...>	
オプション	-i <file名> *ドメイン名は指定しません	構成ファイル(xml)を指定し、ドメインを作成する
	mac-addr=<number>	ドメインのMACアドレスを指定する
	hostid=<number>	ドメインのホストIDを指定する
	failure-policy=<ignore stop reset panic>	マスタードメインの障害ポリシーを指定する
	master=<masterドメイン名...>	マスタードメインを指定する
	max-cores=<number unlimited>	ドメインに割り当てるコアの最大数を指定する unlimited: 制限なくコア数を割り当て可能 (デフォルトはunlimited)
	cpu-arch=<native migration-class1 sparc64-class1 generic> *Oracle VM 3.1の追加オプション	ライブマイグレーション先のサーバの種類を指定する native: 同じCPUタイプのサーバ間でのみ移動 でき、CPU性能を最大限に活用可能 migration-class1: SPARC T4、SPARC T5、 およびSPARC M5間で移動でき、 SPARC T4のCPU性能を活用可能 sparc64-class1: SPARC M10間で移動でき SPARC64 XのCPU性能を活用可能 generic: 任意のCPUタイプのサーバ間で移動 できるが、全てのCPUがサポートする 命令以外使用できないので、性能に 影響がでる可能性あり (デフォルトは native)
rc-add-policy=[iov]	PCI Express Direct I/Oを有効にする	
★ shutdown-group=<priority>	ドメインの停止グループ番号を指定する	
使用例	ldm create ldom1	
	ldm create -i ldom1.xml	
	ldm create master=primary ldom1	

set-domain (modify)		ドメインの設定変更
形式	ldm modify -i <file名> ldm modify [オプション] <ドメイン名1, ドメイン名2, ...>	
オプション	-i <file名> *ドメイン名は指定しません	構成ファイル(xml)により、ドメインの設定を変更する
	mac-addr=<number>	ドメインのMACアドレスを指定する
	hostid=<number>	ドメインのホストIDを指定する
	failure-policy=<ignore stop reset panic>	マスタードメインの障害ポリシーを指定する
	master=<masterドメイン名...>	マスタードメインを設定する
	max-cores=<number unlimited>	ドメインに割り当てるコアの最大数を指定する unlimited: 制限なくコア数を割り当て可能 (デフォルトはunlimited)
	cpu-arch=<native migration-class1 sparc64-class1 generic> *Oracle VM 3.1の追加オプション	ライブマイグレーション先のサーバの種類を指定する native: 同じCPUタイプのサーバ間でのみ移動 でき、CPU性能を最大限に活用可能 migration-class1: SPARC T4, SPARC T5, およびSPARC M5間で移動でき、 SPARC T4のCPU性能を活用可能 sparc64-class1: SPARC M10間で移動でき SPARC64 XのCPU性能を活用可能 generic: 任意のCPUタイプのサーバ間で移動 できるが、全てのCPUがサポートする 命令以外使用できないので、性能に 影響がでる可能性あり (デフォルトは native)
rc-add-policy=[iov]	PCI Express Direct I/Oを有効にする	
★ shutdown-group=<priority>	ドメインの停止グループ番号を指定する	
使用例	ldm modify failure-policy=reset master=primary ldom1	

remove-domain (destroy)		ドメインの削除
形式	ldm destroy -a ldm destroy <ドメイン名1, ドメイン名2, ...>	
オプション	-a *ドメイン名は指定しません	全てのドメインを削除する
使用例	ldm destroy ldom1 ldom2 ldm destroy -a	

bind-domain (bind)		ドメインのリソースバインド
形式	ldm bind [オプション] <-i file名   ドメイン名>	
オプション	-i <file名>	構成ファイル(xml)により、ドメインを作成・バインドする
	-f	無効なデバイスが検出されても、強制的にバインドする
	-q	デバイスの検証をしないで、バインドする
使用例	ldm bind ldom1 ldm bind -i ldom1.xml	

unbind-domain (unbind)		ドメインのリソース解除
形式	ldm unbind <ドメイン名>	
使用例	ldm unbind ldom1	

start-domain (start)		ドメインの起動
形式	ldm start <-a   -i file名   ドメイン名1, ドメイン名2, ...>	
オプション	-a *ドメイン名は指定しません	全てのドメインを起動する
	-i <file名> *ドメイン名は指定しません	構成ファイル(xml)により、ドメインを作成・起動する
使用例	ldm start ldom1 ldm start -a ldm start -i ldom1.xml	

stop-domain (stop)		ドメインの停止
形式	ldm stop [-f   -q] <-a   ドメイン名1, ドメイン名2, ...> ldm stop [-h   -r   -t sec] [-m msg] <-a   ドメイン名1, ドメイン名2, ...>	
オプション	-f	強制的にドメインを停止する
	-q	ドメインをすばやく停止する (uadminコマンド)
	-h	OSを停止させ、ドメインを停止 (shutdownコマンド)
	-r	ドメインのOSをリブートする (shutdownコマンド)
	-t sec	sec秒後にドメインをすばやく停止する
	-m msg	停止/リブートするドメインに送信するメッセージを指定
	-a	*ドメイン名は指定しません 全てのドメインを停止する
使用例	ldm stop ldom1 ldom2 ldm stop -a ldm stop -f ldom1 ldm stop -r -m "We will reboot your domain." ldom1	

panic-domain (panic)		ドメインの強制パニック
形式	ldm panic <ドメイン名>	
使用例	ldm panic ldom1	

## 2. vcpu

add-vcpu		CPUスレッドの追加
形式	<code>ldm add-vcpu &lt;number&gt; &lt;ドメイン名&gt;</code>	
使用例	<code>ldm add-vcpu 8 primary</code>	
	<code>ldm add-vcpu 16 ldom1</code>	

set-vcpu		CPUスレッド数の設定
形式	<code>ldm set-vcpu &lt;number&gt; &lt;ドメイン名&gt;</code>	
使用例	<code>ldm set-vcpu 4 primary</code>	
	<code>ldm set-vcpu 16 ldom1</code>	

remove-vcpu (rm-vcpu)		CPUスレッドの削除
形式	<code>ldm rm-vcpu [オプション] &lt;number&gt; &lt;ドメイン名&gt;</code>	
オプション	<code>-f</code>	CPUスレッドの削除を強制的に試行する
使用例	<code>ldm rm-vcpu 8 primary</code>	
	<code>ldm rm-vcpu -f 4 ldom1</code>	

## 3. core

add-core		CPUコアの追加
形式	<code>ldm add-core &lt;number&gt;</code>	
	<code>ldm add-core cid=&lt;core-ID&gt;[,core-ID][,...] &lt;ドメイン名&gt;</code>	
オプション	<code>cid=&lt;core-ID&gt;[, ...]</code>	追加するコアのID(物理位置)を指定する
使用例	<code>ldm add-core 8 primary</code>	
	<code>ldm add-core cid=60 ldom1</code>	

set-core		CPUコア数の設定
形式	<code>ldm set-core &lt;number&gt;</code>	
	<code>ldm set-core cid=[core-ID][,core-ID][,...] &lt;ドメイン名&gt;</code>	
オプション	<code>cid=[core-ID][, ...]</code>	割り当てるコアのID(物理位置)を指定する
使用例	<code>ldm set-core 4 primary</code>	
	<code>ldm set-core cid=60 ldom1</code>	

remove-core (rm-core)		CPUスレッドの削除
形式	<code>ldm rm-core &lt;number&gt;</code>	
	<code>ldm rm-core cid=[core-ID][,core-ID][,...] &lt;ドメイン名&gt;</code>	
オプション	<code>-f</code>	CPUコアの削除を強制的に試行する
	<code>cid=[core-ID][, ...]</code>	削除するコアのID(物理位置)を指定する
使用例	<code>ldm rm-core 4 primary</code>	
	<code>ldm rm-core cid=60 ldom1</code>	

## 4. memory (mem)

add-memory (add-mem)		メモリ容量の追加
形式	ldm add-mem [--auto-adj] <mem-size[unit]> <ドメイン名> ldm add-mem mblock=<PA-start:mem-size[unit]>[,<PA-start:mem-size[unit]>][,...] <ドメイン名>	
オプション	--auto-adj	追加するメモリ容量を256MB単位に切り上げる
	unit	メモリの追加単位を指定する、Gバイトは"G" or "g", Mバイトは"M" or "m", Kバイトは"K" or "k"
	mblock	追加するメモリの物理アドレスを指定する PA-start:メモリの開始アドレス(16進数)を指定 mem-size:メモリの容量(256MB単位)を指定
使用例	ldm add-mem 1G primary	
	ldm add-mem 1024M ldom1	
	ldm add-mem mblock=0x7e00000000:32G ldom2	

set-memory (set-mem)		メモリ容量の設定
形式	ldm set-mem [--auto-adj] <number>[unit] <ドメイン名> ldm set-mem mblock=<PA-start:mem-size[unit]>[,<PA-start:mem-size[unit]>][,...] <ドメイン名>	
オプション	--auto-adj	割り当てるメモリ容量を256MB単位に切り上げる
	unit	メモリの設定単位を指定する、Gバイトは"G" or "g", Mバイトは"M" or "m", Kバイトは"K" or "k"
	mblock	割り当てるメモリの物理アドレスを指定する PA-start:メモリの開始アドレス(16進数)を指定 mem-size:メモリの容量(256MB単位)を指定
使用例	ldm set-mem 1G primary	
	ldm set-mem 1024M ldom1	
	ldm set-mem mblock=0x7e00000000:16G ldom2	

remove-memory (rm-mem)		メモリ容量の削除
形式	ldm rm-mem [--auto-adj] <number>[unit] <ドメイン名> ldm set-mem mblock=<PA-start:mem-size[unit]>[,<PA-start:mem-size[unit]>][,...] <ドメイン名>	
オプション	--auto-adj	割り当てるメモリ容量を256MB単位に切り上げる
	unit	メモリの設定単位を指定する、Gバイトは"G" or "g", Mバイトは"M" or "m", Kバイトは"K" or "k"
	mblock	削除するメモリの物理アドレスを指定する PA-start:メモリの開始アドレス(16進数)を指定 mem-size:メモリの容量(256MB単位)を指定
使用例	ldm rm-mem 1G primary	
	ldm set-mem 1024M ldom1	
	ldm rm-mem mblock=0x7e00000000:8G ldom2	



## 5. spconfig (config)

add-spconfig (add-config)		ドメインの構成情報を保存
形式	ldm add-config [オプション] <config名>	
使用例	ldm add-config config_initial	

set-spconfig (set-config)		使用する構成情報を変更
形式	ldm set-config <config名>	
使用例	ldm set-config config_initial	

remove-spconfig (rm-config)		ドメインの構成情報を削除
形式	ldm rm-config [オプション] <config名>	
使用例	ldm rm-config config_initial	

list-spconfig (ls-config)		ドメインの構成情報を一覧表示
形式	ldm ls-config [オプション]	
使用例	ldm ls-config	

## 6. hvdump

★ set-hvdump		ハイパーバイザのダンプデータ取得の設定を変更	
形式	ldm set-hvdump [オプション]		
オプション	hvdump=<on off>	*初期値はon	ダンプ収集を有効または無効にする
	hvdump-reboot=<on off>	*初期値はoff	ダンプ収集後の自動リブートを有効または無効にする
使用例	ldm set-hvdump hvdump=off		
	ldm set-hvdump hvdump-reboot=on		

★ list-hvdump (ls-hvdump)		ハイパーバイザのダンプデータ取得の設定を確認	
形式	ldm ls-hvdump		
使用例	ldm ls-hvdump		

★ start-hvdump		ハイパーバイザのダンプデータを手動で取得	
形式	ldm start-hvdump		
使用例	ldm start-hvdump		

## 7. vconscon (vcc)

add-vconscon (add-vcc)		仮想コンソールサービスの作成
形式	ldm add-vcc port-range=<x>-<y> <vcc名> <ドメイン名>	
オプション	x	コンソールポートの開始番号
	y	コンソールポートの終了番号
使用例	ldm add-vcc port-range=5000-5100 primary-vcc0 primary	

set-vconscon (set-vcc)		仮想コンソールサービスの設定変更
形式	ldm set-vcc port-range=<x>-<y> <vcc名>	
オプション	x	コンソールポートの開始番号
	y	コンソールポートの終了番号
使用例	ldm set-vcc port-range=5000-5100 primary-vcc0	

remove-vconscon (rm-vcc)		仮想コンソールサービスの削除
形式	ldm rm-vcc [オプション] <vcc名>	
オプション	-f	強制的に削除する
使用例	ldm rm-vcc primary-vcc0	
	ldm rm-vcc -f primary-vcc0	

## 8. vswitch (vsw)

add-vswitch (add-vsw)		仮想スイッチの作成
形式	ldm add-vsw [オプション] <vsw名> <ドメイン名>	
オプション	-q	ネットワークデバイスの検証をしない
	pvid=<pvid_id>	VLAN IDをタグなしモードで指定する
	vid=<vid_id1><,...>	VLAN IDをタグ付きモードで指定する
	linkprop=phys-state	リンクベースのIPMPを使用する場合に設定する
	mac-addr=<number>	仮想スイッチのMACアドレスを指定する
	net-dev=<ネットワークインタフェース名>	仮想スイッチの元となる物理デバイスを指定する
	mtu=<size>	仮想スイッチデバイスの最大転送単位を指定する
	id=<switch-id>	仮想スイッチデバイスの ID を指定する
	inter-vnet-link=<on   off>	同じ仮想スイッチのvnet間の直接通信をOn/Offする
使用例	ldm add-vsw net-dev=net1 pvid=10 vid=20,30 primary-vsw0 primary	
	ldm add-vsw net-dev=net1 linkprop=phys-state primary-vsw0 primary	

set-vswitch (set-vsw)		仮想スイッチの設定変更
形式	ldm set-vsw [オプション] <vsw名>	
オプション	-q	ネットワークデバイスの検証をしない
	pvid=[pvid_id]	タグなしモードのVLAN IDを設定変更する
	vid=[vid_id1,...]	タグ付きモードのVLAN IDを設定変更する
	mac-addr=[number]	仮想スイッチのMACアドレスを変更する
	net-dev=[ネットワークインタフェース名]	仮想スイッチの元となる物理デバイスを変更する
	linkprop=[phys-state]	リンクベースのIPMPを設定する
	mtu=[size]	最大転送単位 (MTU) を変更する
	inter-vnet-link=<on   off>	同じ仮想スイッチのvnet間の直接通信をOn/Offする
使用例	ldm set-vsw net-dev=net1 pvid=10 vid=20,30 primary-vsw0	
	ldm set-vsw linkprop=phys-state primary-vsw0	

remove-vswitch (rm-vsw)		仮想スイッチの削除
形式	ldm rm-vsw [オプション] <vsw名>	
オプション	-f	強制的に削除する
使用例	ldm rm-vsw primary-vsw0	
	ldm rm-vsw -f primary-vsw0	

## 9. vdiskserver (vds)

add-vdiskserver (add-vds)		仮想ディスクサービスの作成
形式	ldm add-vds <仮想ディスクサービス名> <ドメイン名>	
使用例	ldm add-vds primary-vds0 primary	

remove-vdiskserver (rm-vds)		仮想ディスクサービスの削除
形式	ldm rm-vds <仮想ディスクサービス名>	
オプション	-f	強制的に仮想ディスクサービスを削除する
使用例	ldm rm-vds primary-vds0 ldm rm-vds -f primary-vds0	

## 10. vdiskserverdevice (vdsdev)

add-vdiskserverdevice (add-vdsdev)		仮想ディスクサービスに仮想ディスクデバイスを登録
形式	ldm add-vdsdev [オプション] <バックエンドデバイス名> <ボリューム名>@<仮想ディスクサービス名>	
オプション	-f	強制的に仮想ディスクサービスにデバイスを追加する
	-q	バックエンドの検証をしない
	options=<ro   slice   excl>	仮想ディスクのオプションを指定(以下のいずれか) ro: 読み取り専用の仮想ディスク slice: 単一スライスの仮想ディスク excl: 排他的なアクセス(1つのドメインのみで使用)
使用例	ldm add-vdsdev /dev/zvol/dsk/ovm/vdisk0 vol0@primary-vds0 ldm add-vdsdev -f options=ro /dev/dsk/c3t1d0s2 vol1@primary-vds0	

set-vdiskserverdevice (set-vdsdev)		仮想ディスクサービスに登録されたデバイスの設定変更
形式	ldm set-vdsdev [オプション] <ボリューム名>@<仮想ディスクサービス名>	
オプション	-f	強制的に仮想ディスクサービスのデバイス設定を変更
	options=<ro   slice   excl>	仮想ディスクのオプションを変更(以下のいずれか) ro: 読み取り専用の仮想ディスク slice: 単一スライスの仮想ディスク excl: 排他的なアクセス(1つのドメインのみで使用)
	mpgroup=<mpgroup>	ディスクのマルチパスグループ名を指定する (仮想ディスクのフェイルオーバーの構築時に使用)
使用例	ldm set-vdsdev -f options=ro vol0@primary-vds0	

remove-vdiskserverdevice (rm-vdsdev)		仮想ディスクサービスに登録されたデバイスの削除
形式	ldm rm-vdsdev [オプション] <ボリューム名>@<仮想ディスクサービス名>	
オプション	-f	強制的に仮想ディスクサービスからデバイスを削除する
使用例	ldm rm-vdsdev vol0@primary-vds0 ldm rm-vdsdev -f vol0@primary-vds0	

## 11. vdisk

add-vdisk		仮想ディスクの作成
形式	ldm add-vdisk [オプション] <仮想ディスク名> <ボリューム名>@<仮想ディスクサーバ名> <ドメイン名>	
オプション	timeout=[seconds]	仮想ディスクへの接続をリトライする時間を指定
	id=<disk-id> *初期値はIDを自動生成	仮想ディスクデバイスの IDを設定
使用例	ldm add-vdisk vdisk0 vol0@primary-vds0 ldom1	

set-vdisk		仮想ディスクの設定変更
形式	ldm set-vdisk [オプション] <仮想ディスク名> <ドメイン名>	
オプション	timeout=[seconds]	仮想ディスクへの接続をリトライする時間を指定
	volume=<ボリューム名>@<仮想ディスクサーバ名>	仮想ディスクに割り当てるボリュームを変更する
使用例	ldm set-vdisk volume=vol1@primary-vds0 vdisk0 ldom1	

remove-vdisk (rm-vdisk)		仮想ディスクの削除
形式	ldm rm-vdisk [オプション] <仮想ディスク名> <ドメイン名>	
オプション	-f	強制的に仮想ディスクを削除する
使用例	ldm rm-vdisk vdisk0 ldom1	
	ldm rm-vdisk -f vdisk0 ldom1	

## 12. vnet

add-vnet		仮想ネットワークデバイスの作成
形式	ldm add-vnet [オプション] <仮想ネットワークデバイス名> <仮想スイッチ名> <ドメイン名>	
オプション	mac-addr=<number>	仮想ネットワークデバイスの MAC アドレスを指定する
	alt-mac-addr=<auto   number1> [,auto   number2]	代替MACアドレスを指定する 指定した代替MACアドレスの数だけ、 仮想NIC (vNIC)を作成可能 autoを指定すると、自動でMACアドレスが生成される
	*auto とnumberは混在可能	
	*Oracle VM 3.1の追加オプション	
	pvid=<pvid_id>	VLAN IDをタグなしモードで設定する
	vid=<vid_id>[,...]	VLAN IDをタグ付きモードで設定する
	linkprop=phys-state	リンクベースのIPMPを使用する場合に設定する
id=<network-id> *初期値はIDを自動生成	仮想ネットワークデバイスのIDを指定する	
mtu=<size>	最大転送単位 (MTU) を設定する	
使用例	ldm add-vnet vnet0 primary-vsw0 ldom1	
	ldm add-vnet pvid=10 vid=20,30 vnet0 primary-vsw0 ldom1	
	ldm add-vnet linkprop=phys-state vnet0 primary-vsw0 ldom1	

set-vnet		仮想ネットワークデバイスの設定変更
形式	ldm set-vnet [オプション] <仮想ネットワークデバイス名> <ドメイン名>	
オプション	mac-addr=<number>	仮想ネットワークデバイスの MAC アドレスを変更する
	alt-mac-addr=<auto   number1> [,auto   number2]	代替MACアドレスを指定する 指定した代替MACアドレスの数だけ、 仮想NIC (vNIC)を作成可能 autoを指定すると、自動でMACアドレスが生成される
	*auto とnumberは混在可能	
	*Oracle VM 3.1の追加オプション	
	vswitch=<仮想スイッチ名>	vnetに割り当てる仮想スイッチを変更する
	pvid=<pvid_id>	VLAN IDをタグなしモードで設定する
	linkprop=phys-state	リンクベースのIPMPを設定できるようにする
vid=<vid_id>[,...]	VLAN IDをタグ付きモードで設定する	
mtu=<size>	最大転送単位 (MTU) を設定する	
使用例	ldm set-vnet vswitch=primary-vsw1 vnet0 ldom1	
	ldm set-vnet pvid=10 vid=20,30 vswitch=primary-vsw1 vnet0 ldom1	
	ldm set-vnet linkprop=phys-state vswitch=primary-vsw1 vnet0 ldom1	
	ldm set-vnet alt-mac-addr=auto,auto vnet0 ldom1	

remove-vnet (rm-vnet)		仮想ネットワークデバイスの削除
形式	ldm rm-vnet [オプション] <仮想ネットワークデバイス名> <ドメイン名>	
オプション	-f	強制的に仮想ネットワークデバイスを削除する
使用例	ldm rm-vnet vnet0 ldom1 ldm rm-vnet -f vnet0 ldom1	

## 13. io

add-io		物理I/O デバイスの追加
形式	ldm add-io [オプション] <PCIeバス(ルートコンプレックス)名> <ドメイン名> ldm add-io <PCIeバス   PCI Express Direct I/Oデバイス名   SR-IOV 仮想デバイス名> <ドメイン名>	
オプション	iov=on   off *初期値はoff	ルートコンプレックスのPCI Express Direct I/O, SR-IOV機能を有効(on)または無効(off)にする
使用例	ldm add-io pci@8600 ldom1 ldm add-io iov=on pci@8700 ldom2	

set-io		PCIeバス(ルートコンプレックス)の設定変更
形式	ldm set-io iov=<on   off> <PCIeバス(ルートコンプレックス)名>	
オプション	iov=on   off *初期値はoff	ルートコンプレックスのPCI Express Direct I/O, SR-IOV機能を有効(on)または無効(off)にする
使用例	ldm add-io iov=off pci@8700	

remove-io (rm-io)		物理I/O デバイスの削除
形式	ldm rm-io [オプション] <PCIeバス   PCIe Direct I/Oデバイス名   SR-IOV 仮想デバイス名> <ドメイン名>	
オプション	-n	コマンドが成功するかどうか事前確認する
使用例	ldm rm-io -n pci@8600 ldom1	

## 14. vconsole (vcons)

set-vconsole (set-vcons)		仮想コンソールの設定
形式	ldm set-vcons [オプション] <ドメイン名>	
オプション	port=[port-number]	ドメインのコンソールに使用するポート番号を指定する
	group=group	コンソールグループ名を指定する (同一のポートで、複数のドメインのコンソールに接続)
	service=vcc-server	仮想コンソールのサービスを指定
	log=<on   off>	*初期値はon 仮想コンソールのロギングを有効/無効に設定する
使用例	ldm set-vcons port=5010 ldom1 ldm set-vcons service=primary-vcc1 ldom2	

## 15. variable (var)

add-variable (add-var)		環境変数の設定追加
形式	ldm add-var <オプション> <ドメイン名>	
オプション	variable名1=<value1> [variable名2=value2] [...]	環境変数を指定する
使用例	ldm add-var auto-boot? =true ldom1 ldm add-var boot-device=vdisk0 ldom1	

set-variable (set-var)		環境変数の設定変更
形式	ldm set-var <オプション> <ドメイン名>	
オプション	variable名1=<value1> [...]	環境変数を変更する
使用例	ldm set-var auto-boot? =false boot-device=vdisk0 ldom1	

remove-variable (rm-var)		環境変数の設定を削除
形式	ldm rm-var <オプション> <ドメイン名>	
オプション	<variable名1> [variable名2] [...]	環境変数の設定を削除する
使用例	ldm rm-var auto-boot? ldom1	

## 16. migrate

migrate-domain		ドメインの移行
形式	ldm migrate [オプション] <移行対象ドメイン名> [<移行先ユーザ名>@]<移行先ホスト名>[:移行先ドメイン名]	
オプション	-f	ドメインの移行を強制的に行なう
	-n	移行の予行演習を実行する
	-p <passwd_file>	パスワードの自動読取を指定する
	*移行先サーバのroot/パスワードが記述されたファイル <ターゲットホスト名> *IPアドレスの指定も可能	移行先のホスト名を指定する
	[:ターゲットドメイン名] *ドメイン名の変更	移行元のドメイン名を移行先で変更する
使用例	ldm migrate -n -p passfile ldom1 root@t5120-ldm ldm migrate ldom1 192.168.10.20:ldom2	

## 17. operation (op)

cancel-operation (cancel-op)		操作の取消
形式	ldm cancel-op [オプション] <ドメイン名>	
オプション	reconf	遅延再構成の取消
	migration	マイグレーション操作の取消
	memdr	メモリー動的再構成削除の取消
使用例	ldm cancel-op reconf primary ldm cancel-op migration ldom1	

## 18. policy

add-policy		CPU Dynamic Resource Management ポリシーの指定
形式	<code>ldm add-policy [オプション] name=&lt;ポリシー名&gt; &lt;ドメイン名&gt;...</code>	
オプション	<code>enable=yes   no</code>	ポリシーの有効、無効を設定する
	<code>priority=&lt;value&gt;</code> ※数値が小さい程、優先度が高くなる	複数ポリシーの実行優先順位を設定する
	<code>attack=&lt;value&gt;</code>	仮想CPUを追加する際の単位を指定する
	<code>decay=&lt;value&gt;</code>	仮想CPUを削除する際の単位を指定する
	<code>sample-rate=&lt;value&gt;</code>	負荷をサンプルリングする周期を秒単位で指定する。
	<code>tod-begin=&lt;hh&gt;:&lt;mm&gt;[:&lt;ss&gt;]</code> ※h:時, m:分, s:秒	ポリシーが有効となる時間を指定する
	<code>tod-end=&lt;hh&gt;:&lt;mm&gt;[:&lt;ss&gt;]</code>	ポリシーが無効となる時間を指定する
	<code>util-lower=&lt;percent&gt;</code>	CPU使用率の下限を%で指定する
	<code>util-upper=&lt;percent&gt;</code>	CPU使用率の上限を%で指定する
	<code>vcpu-min=&lt;value&gt;</code>	仮想CPUの最小値を指定する
<code>vcpu-max=&lt;value&gt;</code>	仮想CPUの最大値を指定する	
使用例	<code>ldm add-policy enable=yes priority=1 attack=1 sample-rate=8 name=policy_1 primary</code>	
	<code>ldm add-policy enable=no priority=2 vpu-min=8 vcpu=max=24 name=policy_2 ldom1</code>	

set-policy		CPU Dynamic Resource Management ポリシーの設定
形式	<code>ldm set-policy [オプション] name=&lt;ポリシー名&gt; &lt;ドメイン名&gt;...</code>	
オプション	<code>enable=&lt;yes   no&gt;</code>	ポリシーの有効、無効を設定する
	<code>priority=&lt;value&gt;</code> ※数値が小さい程、優先度が高くなる	複数ポリシーの実行優先順位を設定する
	<code>attack=&lt;value&gt;</code>	仮想CPUを追加する際の単位を設定する
	<code>decay=&lt;value&gt;</code>	仮想CPUを削除する際の単位を設定する
	<code>sample-rate=&lt;value&gt;</code>	負荷をサンプルリングする周期を秒単位で設定する。
	<code>tod-begin=&lt;hh&gt;:&lt;mm&gt;[:&lt;ss&gt;]</code> ※h:時, m:分, s:秒	ポリシーが有効となる時間を設定する
	<code>tod-end=&lt;hh&gt;:&lt;mm&gt;[:&lt;ss&gt;]</code>	ポリシーが無効となる時間を設定する
	<code>util-lower=&lt;percent&gt;</code>	CPU使用率の下限を%で設定する
	<code>util-upper=&lt;percent&gt;</code>	CPU使用率の上限を%で設定する
	<code>vcpu-min=&lt;value&gt;</code>	仮想CPUの最小値を設定する
<code>vcpu-max=&lt;value&gt;</code>	仮想CPUの最大値を設定する	
使用例	<code>ldm set-policy enable=no name=policy_1 primary</code>	
	<code>ldm set-policy enable=yes priority=1 tod-begin=09:00 tod-end=15:30 name=policy_2 ldom1</code>	

remove-policy (rm-policy)		CPU Dynamic Resource Management ポリシーの削除
形式	<code>ldm rm-policy [name=]&lt;ポリシー名&gt; ... &lt;ドメイン名&gt;</code>	
使用例	<code>ldm rm-policy name=policy_1 primary</code>	
	<code>ldm rm-policy policy_2 ldom1</code>	

## 19. list (ls)

list-domain (ls-dom)		ドメインの構成情報の表示
形式	ldm ls-dom [オプション] [ドメイン名]	
オプション	-e	仮想サービスや仮想デバイスを含む情報を表示する
	-l	ドメイン構成の詳細を表示する
	★ -S	CPUやメモリなどのリソースのステータスを表示する -e, -lと共に使用 (正常時はok、障害時はfailを表示)
	-o <format: console,core,cpu,crypto, disk,domain,memory,network,physio, resmgmt,serial,status>	特定の情報(CPUやメモリなど)のみを表示する
	-p	スクリプトなどで解析可能な形式で表示する
使用例	ldm ls-dom	
	ldm ls-dom -S -o cpu	

list-services (ls-services)		構成済みの仮想サービスの表示
形式	ldm ls-services [オプション] [ドメイン名]	
オプション	-e	自動設定されるサービスやデバイスも含めて表示する
	-p	スクリプトなどで解析可能な形式で表示する
使用例	ldm ls-services	
	ldm ls-services -p primary	

list-bindings (ls-bindings)		ドメインに割当てられているリソースの表示
形式	ldm ls-bindings [オプション] [ドメイン名]	
オプション	-e	自動設定のサービスやデバイスも含めて表示する
	-p	スクリプトなどで解析可能な形式で表示する
使用例	ldm ls-bindings	
	ldm ls-bindings -p ldom1	

list-devices (ls-devices)		リソースの情報の表示
形式	ldm ls-devices [オプション]	
オプション	-a	全てのリソースを表示する
	※-aがない場合、使用していないリソースのみを表示	
	-p	スクリプトなどで解析可能な形式で表示する
	★ -S	リソースのステータスを表示する (正常時はok、障害時はfailを表示)
	core	CPUコアに関する情報を表示する
	cpu	CPU スレッドの情報を表示する
	memory	メモリの情報を表示する
	io	物理I/Oデバイス情報を表示する
使用例	ldm ls-devices	
	ldm ls-devices -a	
	ldm ls-devices memory	

list-constraints (ls-constraints)		環境設定情報の表示
形式	ldm ls-constraints [-e] [-p] [ドメイン名] ldm ls-constraints [-x] [ドメイン名]	
オプション	-e	自動設定のサービスやデバイスも含めて表示する
	-p	スクリプトなどで解析可能な形式で表示する
	-x	環境設定情報をxml形式で出力する
使用例	ldm ls-constraints	
	ldm ls-constraints -x ldom1	



list-variable (list-var)		環境変数の一覧を表示
形式	ldm list-var [オプション] <ドメイン名>	
オプション	variable名1 variable名2 ...	環境変数の一覧を表示する
使用例	ldm list-var auto-boot? boot-device ldom1	

list-io (ls-io)		物理I/Oデバイスの表示
形式	ldm list-io [オプション] <PCIeバス   PCIe Direct I/Oデバイス名   SR-IOV 物理デバイス名>	
オプション	-l	ルートコンプレックスとPCIe Direct I/Oデバイスの関係などの詳細を表示する
	-p	スクリプトなどで解析可能な形式で表示する
使用例	ldm list-io	
	ldm list-io -l	

★ list-permits (ls-permits)		Capacity on Demand 許可情報の一覧表示
形式	ldm ls-permits	
使用例	ldm ls-permits	

## 20. system

init-system		ドメインの構成の復元
形式	ldm init-system [オプション] -i <xmlファイル名>	
オプション	-i	ドメインの復元に使用するxmlファイルを指定する
	-f	ドメインの構成チェックを省略する
	-r	ドメイン復元後、自動でシステムを再起動する
	-s	仮想サービス(vds, vcc, vsw)のみを復元する
使用例	ldm init-system -r -i /OVM/alldomains.xml	

## 21. reconf

start-reconf		遅延再構成状態の開始
形式	ldm start-reconf <ドメイン名>	
使用例	ldm start-reconf primary	

cancel-reconf		遅延再構成状態の取り消し
形式	ldm cancel-reconf <ドメイン名>	
使用例	ldm cancel-reconf primary	