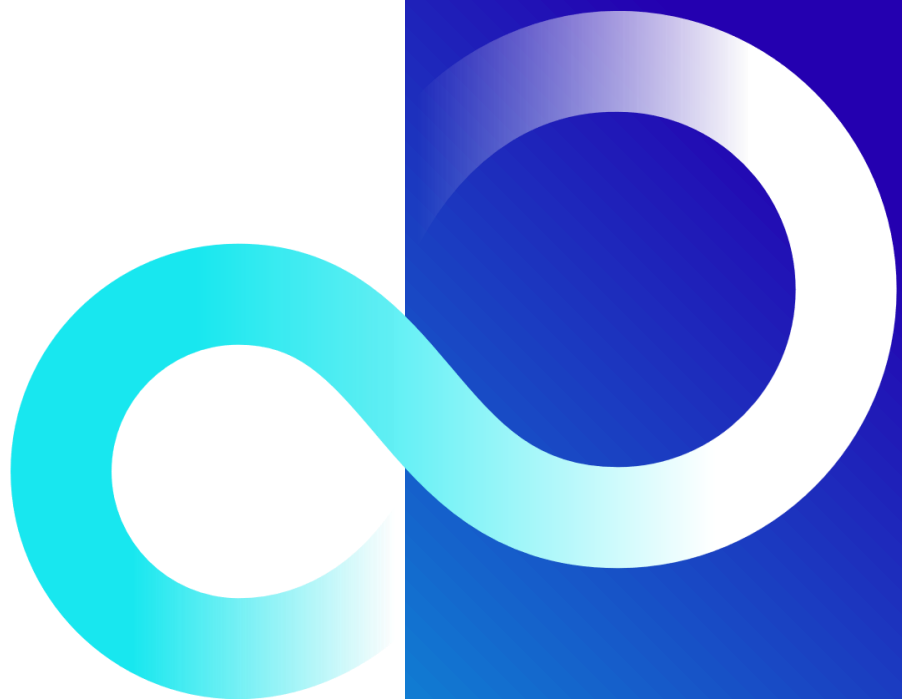


Oracle VM Server for
SPARCを使ってみよう
(構築・運用手順書)



2019年6月

第4.0版

富士通株式会社

■ 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
本製品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。

■ 商標について

- 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。

はじめに

本書の内容

- Oracle VM Server for SPARC の構築手順を紹介しています。
別冊の『Oracle VM Server for SPARC を使ってみよう（構築・運用ガイド）』を参照しながら、本書を活用してください。
- Oracle VM Server for SPARC を使ってみよう
<https://www.fujitsu.com/jp/sparc-technical/document/solaris/#ovm>

留意事項

- 本書は、SPARC M12-1, Oracle VM Server for SPARC 3.6, Oracle Solaris 11.4 を使用した手順を紹介しています。
- 本書は、以下の環境（バージョン）を元に作成されています。

ハードウェア／ファームウェア	
サーバ	SPARC M12-1
XCP ファームウェア	XCP3060
ソフトウェア	
OS	Oracle Solaris 11.4
SRU (Support Repository Update)	SRU19031 (SRU11.4.7.4.0)

- 本書に記載されているコマンドの実行結果は、環境によって異なる場合があります。

本書での表記

- 以下の用語は略称を用いて表記する場合があります。

略称	正式名称
Solaris	Oracle Solaris
Oracle VM	Oracle VM Server for SPARC

- Logical Domains (LDoms) は、Ver2.0 より、Oracle VM Server for SPARC に名称が変更となりました。
- コマンド実行例の表記は、実行環境に応じてプロンプトが異なります。また、ゲストドメインで実行する場合は囲み枠を区別しています。
 - ▶ 本書では、「ゲストドメイン 1」、「ゲストドメイン 2」の 2 つのゲストドメインを構築します。詳しくは、「[本書で構築する仮想環境](#)」—「[本書で構築する環境のイメージ図](#)」を参照してください。

例 1: XSCF で実行するコマンド

```
XSCF> version -c xcp
```

- ▶ プロンプトを「XSCF>」と表記します。

例 2: OBP で実行するコマンド

```
{0} ok boot
```

- ▶ プロンプトを「{0} ok」と表記します。

例 3: 各ドメインの OS で実行するコマンド

制御ドメイン

```
primary# cat /etc/release
```

- ▶ プロンプトを「primary#」と表記します。

ゲストドメイン

```
ldom1# cat /etc/release
```

```
ldom2# cat /etc/release
```

- ▶ ゲストドメイン 1 のプロンプトを「ldom1#」、ゲストドメイン 2 のプロンプトを「ldom2#」と表記します。

- コマンド実行例の上に記載している ldm コマンドの【書式】は、ロングフォーマットです。

初出時のみ、ショートフォーマットを以下の形で併記しています。

SF ロングフォーマット → ショートフォーマット

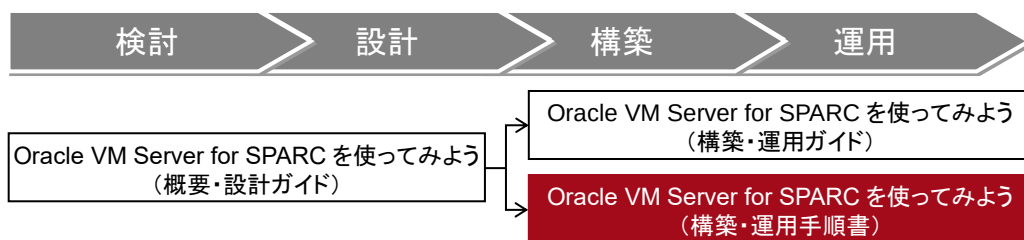
例を以下に示します。以下の例では、ロングフォーマット「ldm add-vdiskserver」のショートフォーマットが「ldm add-vds」であることを示しています。

【書式】ldm add-vdiskserver <仮想ディスクサービス名> <ドメイン名>

SF ldm add-vdiskserver → ldm add-vds

```
primary# ldm add-vdiskserver primary-vds0 primary
```

ドキュメントの位置づけ



目次

本書で構築する仮想環境	1
1. サーバ環境の確認	2
1.1. ファームウェア環境の確認	2
1.2. OS 環境の確認	2
2. 制御ドメインの構築	4
2.1. SRU(Support Repository Updates)の適用	4
2.2. 仮想サービスの作成	6
2.2.1. 仮想化環境構築前の確認	6
2.2.2. 仮想サービスの作成	7
2.3. 制御ドメインのリソース設定	12
2.4. Oracle VM の設定内容の保存	14
3. ゲストドメインの構築	16
3.1. ゲストドメイン 1 の構築	16
3.1.1. ゲストドメインの作成	16
3.1.2. リソースの割り当て	17
3.1.3. 環境パラメーターの設定	18
3.1.4. ゲストドメインの起動	20
3.1.5. ゲストドメインへの OS インストール	21
3.1.6. ESF、パッチの適用	22
3.1.7. 構築後のゲストドメインの環境確認	23
3.2. ゲストドメイン 2 の構築	24
4. ネットワークの冗長化	27
4.1. 制御ドメインでの冗長化	27
4.1.1. リンクアグリゲーションの設定(制御ドメイン上)	27
4.1.2. 仮想スイッチサービスの作成(制御ドメイン上)	29
4.1.3. ゲストドメイン上の設定	29

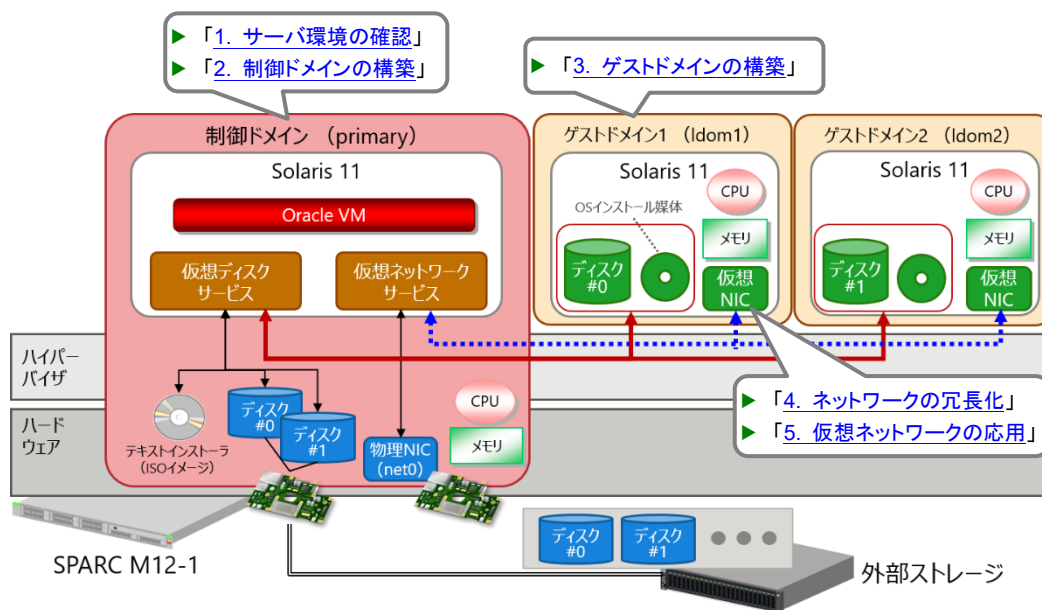
4.1.4.	冗長化設定の削除	29
4.2.	ゲストドメインでの冗長化	31
4.2.1.	仮想スイッチの作成(制御ドメイン上)	31
4.2.2.	仮想ネットワークデバイスの割り当て(制御ドメイン上)	32
4.2.3.	IPMP の設定(ゲストドメイン上)	32
4.2.4.	冗長化設定の削除	33
5.	仮想ネットワークの応用	35
5.1.	VLAN によるネットワークの統合	35
5.1.1.	異なるサブネットの業務を 1 つの NIC で統合	35
5.1.2.	VLAN 設定の解除	37
5.2.	仮想ネットワークの帯域制御	38
5.3.	ゲストドメイン間の内部ネットワークの構築	43
6.	I/O ルートドメインの構築	46
6.1.	物理 I/O デバイスの環境確認	46
6.2.	PCIe バスの割り当て	47
6.3.	PCIe バスの割り当ての解除	49
7.	ドメインの操作	50
7.1.	ドメインの操作	50
7.1.1.	ドメインの状態確認	50
7.1.2.	ドメインの停止	51
7.1.3.	ドメインの起動	52
7.1.4.	ドメインの仮想コンソール接続	52
7.1.5.	ドメインの仮想コンソール接続の切断	54
7.1.6.	制御ドメインの停止	54
7.1.7.	制御ドメインの再起動	54
7.2.	サーバ電源の操作	55
7.2.1.	サーバの電源 OFF(制御ドメインで実行する場合)	55
7.2.2.	サーバの電源 OFF(XSCF で実行する場合)	55
7.2.3.	サーバの電源 ON	56
8.	リソースの操作	60

8.1. 使用リソースの確認	60
8.2. ドメインへの CPU の追加／削減	60
8.3. ドメインの CPU 割り当て方式の変更	63
8.4. ドメインへのメモリの割り当て	64
8.5. ドメインへの仮想ネットワークの追加	66
8.6. ドメインの仮想ネットワークの削除	69
8.7. ドメインへの仮想ディスクの追加	71
8.8. ドメインの仮想ディスクの削除	76
9. ドメインのバックアップとリストア	80
9.1. ゲストドメインのバックアップとリストア	80
9.1.1. ゲストドメインのバックアップ	80
9.1.2. ゲストドメインのリストア	82
9.2. 制御ドメインのバックアップとリストア	83
9.2.1. 制御ドメインのバックアップ	83
9.2.2. 制御ドメインのリストア	84
付録 1. Oracle VM Server for SPARC の初期化	87
付録 1.1. ゲストドメインの削除	87
付録 1.2. 仮想環境の初期化	87
改版履歴	91

本書で構築する仮想環境

本書で構築する環境のイメージ図

本書では、制御ドメインと2つのゲストドメイン（ゲストドメイン1とゲストドメイン2）を構築し、ゲストドメインのネットワーク（仮想 NIC）を冗長化します。



1. サーバ環境の確認

1.1. ファームウェア環境の確認

1) XCP ファームウェア版数を確認します。

SPARC M12-1 / Oracle VM Server for SPARC 3.6 の場合、XCP のバージョンが「3030」以降であることを確認します。

```
XSCF> version -c xcp
BB#00-XSCF#0 (Master)
XCP0 (Reserve): 3060
XCP1 (Current): 3060
```

Point

「XSCF>」は XSCF のプロンプトです。

1.2. OS 環境の確認

ここでは、すでに OS がインストールされている状態での操作を説明します。

1) サーバが起動していることを確認します。

Status が「Running」(POST 処理が完了して稼働中)であることを確認してください。

```
XSCF> showpci -a
PPAR-ID  LSB  PSB  Status
00              Running
          00  00-0
```

2) OS のコンソールに接続します。

```
XSCF> console -p 0

Console contents may be logged.
Connect to PPAR-ID 0?[y|n] :y
```

3) 一般ユーザでログインします。

```
primary console login: user-id
Password:*****
```

Point

通常、Solaris 11 では root がユーザではなく、role(役割)として設定されるため、root ではログインできません。

4) root 権限へ切り替えます。

```
$ su -  
Password:*****
```

5) OS の版数を確認します。

```
primary# cat /etc/release  
Oracle Solaris 11.4 SPARC  
Copyright (c) 1983, 2019, Oracle and/or its affiliates. All rights reserved.  
Assembled 01 February 2019
```

Point

SPARC M12/M10 では、制御ドメインは Solaris 10 1/13 以降または 11.2 以降である必要があります。また、Oracle VM 3.6 を導入する場合は、Solaris 11.4 以降である必要があります。
詳細は、『[Oracle VM Server for SPARC を使ってみよう\(概要・設計ガイド\)](#)』の「5. Oracle VM Server for SPARC の適用要件」を参照してください。

6) サーバの構成情報を確認します。

i) CPU 数を確認します。

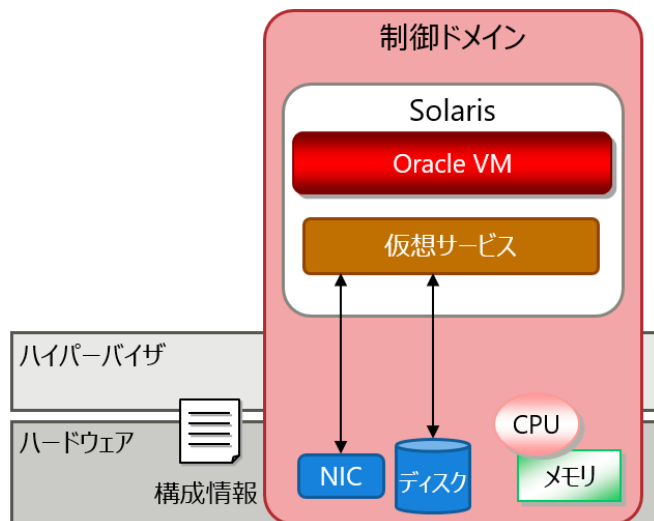
```
primary# psrinfo -vp  
The physical processor has 6 cores and 48 virtual processors (0-47)  
The core has 8 virtual processors (0-7)  
The core has 8 virtual processors (8-15)  
The core has 8 virtual processors (16-23)  
The core has 8 virtual processors (24-31)  
The core has 8 virtual processors (32-39)  
The core has 8 virtual processors (40-47)  
SPARC64-XII (chipid 0, clock 3200 MHz)
```

ii) カーネルアーキテクチャとメモリ容量を確認します。

```
primary# prtconf -v | head  
System Configuration: Oracle Corporation sun4v  
Memory size: 128512 Megabytes  
System Peripherals (Software Nodes):  
~ (省略) ~
```

2. 制御ドメインの構築

制御ドメインを作成し、各種設定を行います。



2.1. SRU (Support Repository Updates) の適用

SRU (OS の修正パッケージ) を適用し、Oracle VM Server for SPARC のバージョンをアップデートします。

1) OS に適用されている修正パッケージ (SRU) を確認します。

本書の環境では SRU が適用されていないため、「Summary」に SRU 版数が表示されません。

```
primary# pkg info entire
Name: entire
Summary: Incorporation to lock all system packages to the same build
Description: This package constrains system package versions to the same
build. WARNING: Proper system update and correct package
selection depend on the presence of this incorporation.
Removing this package will result in an unsupported system.
Category: Meta Packages/Incorporations
～（省略）～
```

2) Oracle VM Server for SPARC のバージョンを確認します。

```
primary# ldm -V

Logical Domains Manager (v 3.6.0.0.23)
Hypervisor control protocol v 1.13
Using Hypervisor MD v 1.3
～（省略）～
```

- ▶ 「ldm」は、Oracle VM Server for SPARC の管理コマンドです。
- ▶ SPARC M12/M10 のプレインストール OS には、Oracle VM Server for SPARC 3.6 以降がインストールされています。

3) BE (Boot Environment) の状態を確認します。

デフォルトでは、「solaris」という名前の BE のみが作成されています。

```
primary# beadm list
BE Name Flags Mountpoint Space Policy Created
-----
solaris NR / 3.04G static 2018-10-19 19:21
```

4) 適用可能な修正パッケージの一覧を表示します。

【書式】pkg update [オプション]

【オプション】-n: 実際にはアップデートしない

-v: 詳細を表示する

```
primary# pkg update -nv
```

5) 新規に BE を作成し、SRU を適用します。

【書式】pkg update [オプション]

【オプション】--be-name BE 名: 新しい BE を作成して、その BE に対して修正パッケージを適用

```
primary# pkg update --be-name be01
```

- ▶ 適用する SRU のバージョンによっては、ライセンスへの同意を必要とするメッセージが表示されます。
メッセージが表示された場合は、--accept オプションを指定して実行します。

6) BE が作成されたことを確認します。

SRU が適用された BE「be01」が作成されています。

```
primary# beadm list
BE Name Flags Mountpoint Space Policy Created
-----
be01 R - 3.71G static 2018-10-22 11:11
solaris N / 330.30M static 2018-10-19 19:21
```

- ▶ 「Flags」の意味は以下のとおりです。
N: 現在有効である BE
R: 再起動後に有効になる BE

7) OS を再起動します。

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

8) 現在有効になっている BE を確認します。

BE「be01」が有効になっていることを確認してください。

```
primary# beadm list
BE Name Flags Mountpoint Space Policy Created
-----
be01 NR / 3.71G static 2018-10-22 11:11
solaris - - 379.98M static 2018-10-19 19:21
```

9) SRU が適用されたことを確認します。

手順 1)と比較し、バージョンが上がっていることを確認してください。

```
primary# pkg info entire
      Name: entire
      Summary: entire incorporation including Support Repository Update
              (Oracle Solaris 11.4.7.4.0).
      Description: This package constrains system package versions to the same
                  build.  WARNING: Proper system update and correct package
                  selection depend on the presence of this incorporation.

~ (省略) ~
```

10) Oracle VM Server for SPARC のバージョンを確認します。

```
primary# ldm -V

Logical Domains Manager (v 3.6.1.0.5)
  Hypervisor control protocol v 1.13
  Using Hypervisor MD v 1.3

System PROM:
~ (省略) ~
```

▶ Oracle VM のバージョンが上がったことを確認します。

SRU(Support Repository Update)の適用後、ESF(Enhanced Support Facility)など、必要なパッケージをインストールしてください。インストール方法については、以下のドキュメントを参照してください。

- 『Oracle Solaris 11 を使ってみよう(構築・運用手順書)』

<https://www.fujitsu.com/jp/sparc/technical/document/solaris/#os>

2.2. 仮想サービスの作成

仮想サービスの作成方法を説明します。サービスドメインとして制御ドメインが仮想サービスを提供します。

2.2.1. 仮想化環境構築前の確認

1) Logical Domains Manager デーモン(ldmd)が起動していることを確認します。

STATE が「online」であることを確認してください。

```
primary# svcs svc:/ldoms/ldmd:default
STATE      STIME      FMRI
online     23:26:51   svc:/ldoms/ldmd:default
```

▶ サービス名は省略形「ldmd」でも確認できます。

2) 現在構築されているドメインを確認します。

デフォルトでは、制御ドメイン (primary) のみ構築されています。

【書式】`ldm list-domain [-e] [-l] [-o <フォーマット>] [-p] [-S] [<ドメイン名>...]`

[SF] `ldm list-domain` → `ldm list`

```
primary# ldm list-domain
NAME          STATE      FLAGS    CONS    VCPU    MEMORY    UTIL    NORM    UPTIME
primary       active     -n-c--   UART    48      128512M   0.0%    0.0%    19m
```

2.2.2. 仮想サービスの作成

Point

操作の途中で、Solaris のインストール DVD の媒体 (ISO イメージ) を使用します。あらかじめ用意しておいてください。

1) システム全体の物理リソースを確認します。

```
primary# ldm list-devices -a
CORE
  ID      %FREE    CPUSET
  0        0      (0, 1, 2, 3, 4, 5, 6, 7)
  1        0      (8, 9, 10, 11, 12, 13, 14, 15)
  2        0      (16, 17, 18, 19, 20, 21, 22, 23)
  4        0      (32, 33, 34, 35, 36, 37, 38, 39)
  5        0      (40, 41, 42, 43, 44, 45, 46, 47)
  6        0      (48, 49, 50, 51, 52, 53, 54, 55)

~ (省略) ~
IO
  DEVICE                                PSEUDONYM    BOUND    OPTIONS
  pci@8000                             PCIE0        yes
  pci@8100                             PCIE1        yes
  pci@8200                             PCIE2        yes
  pci@8300                             PCIE3        yes
PCIEDEV
  pci@8000/pci@4/pci@0/pci@1           /SYS/MBU/NET0  yes
  pci@8100/pci@4/pci@0/pci@9           /SYS/PCIO      yes
  pci@8100/pci@4/pci@0/pci@0           /SYS/MBU/SASHBA0 yes
  pci@8200/pci@4/pci@0/pci@0           /SYS/MBU/NET2  yes
  pci@8200/pci@4/pci@0/pci@8           /SYS/PCI1      yes
  pci@8300/pci@4/pci@0/pci@1           /SYS/PCI2      yes
IOVDEV
```

- ▶ `-a` オプションを指定すると、割り当て済みのリソースも含めて表示されます。`-a` オプションを指定しない場合、空きリソースが表示されます。
- ▶ CORE: CPU コア
VCPU: 仮想 CPU

MAU: 暗号化ユニット

MEMORY: 物理メモリ

IO: IO デバイス

2) 制御ドメインに割り当てられているリソースを確認します。

```
primary# ldm list-bindings primary
NAME          STATE      FLAGS    CONS   VCPU  MEMORY  UTIL  NORM  UPTIME
primary       active    -n-c--   UART   48    128512M  0.0%  0.0%  1h 40m

UUID
    db556de9-143b-43e0-ae3e-3a92dfd1e444

MAC
    b0:99:28:98:10:42

HOSTID
    0x90060032

CONTROL
    failure-policy=ignore
    extended-mapin-space=off
    cpu-arch=native
    rc-add-policy=
    shutdown-group=0
    perf-counters=global, htstrand
    boot-policy=n/a
    effective-max-pagesize=16GB
    hardware-max-pagesize=16GB

DEPENDENCY
    master=

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

VCPU
    VID  PID  CID  UTIL  NORM  STRAND
    0    0    0    0.3% 0.3% 100%
    1    1    0    0.0% 0.0% 100%
    ~ (省略) ~

IO
    DEVICE          PSEUDONYM      OPTIONS
    pci@8000        PCIE0
    pci@8100        PCIE1
    pci@8200        PCIE2
```

```
pci@8300                PCIE3
pci@8000/pci@4/pci@0/pci@1 /SYS/MBU/NET0
~（省略）~
pci@8300/pci@4/pci@0/pci@1 /SYS/PCI2
VCONS
NAME          SERVICE          PORT  LOGGING
              UART
```

- ▶ 制御ドメインに割り当てられているリソースを表示します。
- ▶ MAC:MAC アドレス
- ▶ VCONS: 仮想コンソール

Point

デフォルトではすべてのリソースが制御ドメインに割り当てられています。

3) 構築済みの仮想サービスを確認します。

何も構成されていない(コマンド実行結果が何も表示されない)ことを確認してください。

```
primary# ldmd list-services
primary#
```

←何も表示されません。

4) 制御ドメインに仮想ディスクサービスを作成します。

【書式】ldm add-vdiskserver <仮想ディスクサービス名> <ドメイン名>

[SF] ldm add-vdiskserver → ldm add-vds

```
primary# ldm add-vdiskserver primary-vds0 primary
```


5) 制御ドメインが認識している物理ディスクを確認します。

本環境では、ゲストドメインのシステムディスク用に仮想ディスクサービスに割り当てます。仮想ディスクサービスに割り当てる物理ディスクを確認してください。

```
primary# format
Searching for disks...done

AVAILABLE DISK SELECTIONS:
  0. c0t5000039698132D40d0 <TOSHIBA-AL13SEB600-3703-558. 91GB>
    /scsi_vhci/disk@g5000039698132d40
    /dev/chassis/SYS/HDD0/disk
  1. c0t500003970830F9E1d0 <TOSHIBA-AL13SEB600AL14SE-3703-558. 91GB>
    /scsi_vhci/disk@g500003970830f9e1
    /dev/chassis/SYS/HDD1/disk
  2. c0t600000E00D000000000103460000000d0 <FUJITSU-ETERNUS_DXL-0000-50. 00GB>
    /scsi_vhci/disk@g600000e00d0000000001034600000000
  3. c0t600000E00D0000000001034600010000d0 <FUJITSU-ETERNUS_DXL-0000-50. 00GB>
    /scsi_vhci/disk@g600000e00d0000000001034600010000
~ (省略) ~
```

- ▶ ここでは例として、2. c0t600000E00D000000000103460000000d0 の物理ディスクを仮想ディスクサービスに割り当てます。
- ▶ 0. c0t5000039698132D40d0 は、制御ドメインのシステム領域 (rpool) で使用しています。
- ▶ [Ctrl]+[C]キーで format メニューを終了します。

6) 仮想ディスクサービスにディスクを割り当てます。

【書式】ldm add-vdiskserverdevice [-f] [-q] [options={ro,slice,excl}] [mpgroup=<mpgroup>]

<backend> <ボリューム名>@<仮想ディスクサービス名>

[SF] ldm add-vdiskserverdevice → ldm add-vdsdev

```
primary# ldm add-vdiskserverdevice /dev/dsk/c0t600000E00D000000000103460000000d0s2
vol0@primary-vds0
```

- ▶ 仮想ディスクサービスに物理ディスク全体を割り当てる場合は、s2 を指定します。

7) Solaris のインストール DVD の媒体 (ISO イメージ)を確認します。

```
primary# ls -l /ISO
total 1991307
-rw-r--r--  1 root    root      1018736640 Oct 22 13:30 Sol11.4.iso
```

8) 仮想ディスクサービスに、ISO イメージ (Solaris のインストール DVD) を Read Only で割り当てます。

```
primary# ldm add-vdiskserverdevice options=ro /ISO/Sol11.4.iso vol_iso@primary-vds0
```

9) 仮想コンソールサービスを作成します。

【書式】ldm add-vconscon port-range=<ポート範囲> <仮想コンソールサービス名> <ドメイン名>

[SF] ldm add-vconscon → ldm add-vcc

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

10) ネットワークデバイスのインターフェース名を確認します。

「ixgbe0」のインターフェース名を確認してください。

```
primary# dladm show-phys -L
LINK          DEVICE          LOC
net0          ixgbe0           /SYS/MBU/CMP0/PEX0
net1          ixgbe1           /SYS/MBU/CMP0/PEX0
net2          ixgbe2           /SYS/MBU/CMP0/PEX2
net3          ixgbe3           /SYS/MBU/CMP0/PEX2
```

11) 仮想スイッチサービスを作成します。

制御ドメインの物理 LAN ポート「net0」から、仮想スイッチ「primary-vsw0」を構築します。

【書式】ldm add-vswitch [-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>] [vsw-relay-mode=<local|remote>] <仮想スイッチサービス名> <ドメイン名>

SF ldm add-vswitch → ldm add-vsw

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

12) 作成した仮想サービスを確認します。

```
primary# ldm list-services
VCC
  NAME          LDOM          PORT-RANGE
  primary-vcc0  primary       5000-5100

VSW
  NAME          LDOM          MACADDRESS          NET-DEV  DVID|PVID|VIDs
  ----          ----          -
  primary-vsw0  primary       00:14:4f:f8:93:ac   net0     1|1|--

VDS
  NAME          LDOM          VOLUME          OPTIONS          MPGROUP          DEVICE
  primary-vds0  primary       vol0
/dev/dsk/c0t600000E00D00000000103460000000d0s2
                                vol_iso          ro
/ISO/Sol11.4.iso
```

2.3. 制御ドメインのリソース設定

制御ドメインに割り当てる CPU とメモリリソースを設定します。

1) 制御ドメインに割り当てる CPU(コア)の数を指定します。

制御ドメインに割り当てる CPU を 2 コア(16 スレッド)に変更します。

```
primary# ldadm set-core 2 primary
```

2) 制御ドメインを遅延再構成状態にします。

```
primary# ldadm start-reconf primary
Initiating a delayed reconfiguration operation on the primary domain.
All configuration changes for other domains are disabled until the primary
domain reboots, at which time the new configuration for the primary domain
will also take effect.
```

- ▶ 制御ドメインのメモリ容量を自由に指定するために、遅延再構成にします。本操作を省略(動的にメモリ容量を変更)する場合、メモリの変動サイズが 256 MB の倍数になるように、メモリ容量を指定する必要があります。

3) 制御ドメインに割り当てるメモリ容量を指定します。

```
primary# ldadm set-memory 16g primary
-----
Notice: The primary domain is in the process of a delayed reconfiguration.
Any changes made to the primary domain will only take effect after it reboots.
-----
```

Point

「16g」で 16 GB(16 × 1024 MB)になります。

4) 制御ドメインに割り当てられる予定のリソースを確認します。

```
primary# ldm list-bindings primary
NAME          STATE      FLAGS    CONS   VCPU  MEMORY  UTIL  NORM  UPTIME
primary       active    -ndcv-   UART   16    16G     0.1%  0.1%  2h 34m

UUID
    db556de9-143b-43e0-ae3e-3a92dfd1e444
~ (省略) ~
CORE
    CID      CPUSSET
    0        (0, 1, 2, 3, 4, 5, 6, 7)
    1        (8, 9, 10, 11, 12, 13, 14, 15)

VCPU
    VID      PID      CID      UTIL  NORM  STRAND
    0        0        0        0.2%  0.2%  100%
    1        1        0        0.1%  0.1%  100%
~ (省略) ~
    15       15       1        0.1%  0.1%  100%

MEMORY
    RA              PA              SIZE
    0xa0000000      0x7e00a0000000  16G
~ (省略) ~
VCC
    NAME          PORT-RANGE
    primary-vcc0  5000-5100

VSW
    NAME          MACADDRESS          NET-DEV  DVID|PVID|VIDs
    -----
    primary-vsw0  00:14:4f:f8:93:ac   net0     1|1|--

VDS
    NAME          VOLUME          OPTIONS          MPGROUP          DEVICE
    primary-vds0  vol0
/dev/dsk/c0t600000E00D000000000103460000000d0s2
                    vol_iso          ro
                                      /ISO/Sol11.4.iso

VCONS
    NAME          SERVICE          PORT  LOGGING
                    UART
```

▶ FLAGS に「d」が表示されている状態は、再起動後に構成されることを表しています。

《参考》 FLAGS の意味

- : 可変部分
- c : 制御ドメイン
- d : 遅延再構成状態
- n : 通常(OS 起動)状態
- s : 起動中(停止中)
- t : 切り替え(OBP)状態
- v : サービスドメイン

2.4. Oracle VM の設定内容の保存

ドメインや仮想サービスの構成情報を、SP(サービスプロセッサ)に保存します。

1) 設定した構成情報を SP(サービスプロセッサ)に保存します。

【書式】ldm remove-spconfig <config 名>

ldm add-spconfig <config 名>

☐ SF ldm remove-spconfig → ldm rm-config

☐ SF ldm add-spconfig → ldm add-config

```
primary# ldm add-spconfig config_initial
```

《注意》

同じ名前の構成情報を作成している場合は、既存の構成情報を削除してから保存してください。

(ldm remove-spconfig <構成情報名>)

Point

Oracle VM の設定・構成を変更した場合は、以下のコマンドを実行して必ず構成情報を保存してください。

```
# ldm add-spconfig config_initial
```

2) 保存されている構成情報を確認します。

```
primary# ldm list-spconfig
factory-default
config_initial [current]
```

3) 制御ドメインを再起動し、設定内容を反映します。

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

4) 再起動後、正しく構成されていることを確認します。

FLAGS の「d」(遅延再構成状態)が表示されていないことを確認してください。

```
primary# ldm list-domain -l primary
NAME          STATE      FLAGS    CONS    VCPU    MEMORY    UTIL    NORM    UPTIME
primary       active    -n-cv-   UART    16      16G       0.4%    0.4%    3m
~ (省略) ~

VCONS
NAME          SERVICE          PORT    LOGGING
              UART
```

5) 仮想ネットワークターミナルサーバデーモン(vntsd)の状態を確認します。

```
primary# svcs svc:/ldoms/vntsd:default
STATE      STIME    FMRI
disabled    9:37:18  svc:/ldoms/vntsd:default
```

▶ サービス名は省略形「vntsd」でも確認できます。

6) vntsd を有効にします。

Point

ゲストメインの仮想コンソールにアクセスするには、vntsd を有効にする必要があります。

また、vntsd を有効にするには、事前に仮想コンソールサービス(vcc)を設定する必要があります。

本書では、「2.2.2. 仮想サービスの作成」の[手順 9](#))で vcc を設定済みです。

```
primary# svcadm enable svc:/ldoms/vntsd:default
```

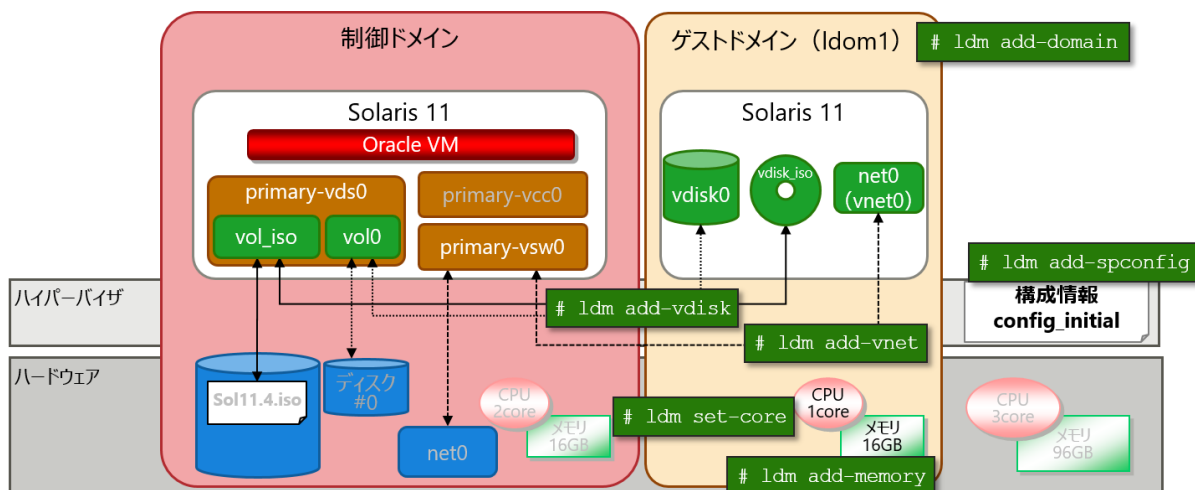
7) vntsd の状態を確認します。

```
primary# svcs svc:/ldoms/vntsd:default
STATE      STIME    FMRI
online      9:38:35  svc:/ldoms/vntsd:default
```

3. ゲストドメインの構築

ゲストドメイン 1 (ldom1) とゲストドメイン 2 (ldom2) を作成し、各種設定を行います。

3.1. ゲストドメイン 1 の構築



3.1.1. ゲストドメインの作成

ドメインの基盤を作成します。

1) ゲストドメインへ割り当て可能なリソースを確認します。

ドメインに利用されていないリソースが表示されます。

```
primary# ldm list-devices
CORE
  ID      %FREE  CPUSET
  2        100   (16, 17, 18, 19, 20, 21, 22, 23)
  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)

VCPU
  PID      %FREE
  16        100
  17        100
  ~ (省略) ~
  55        100

MEMORY
  PA              SIZE
  0x7e04a0000000  112128M
```

2) ゲストドメインを作成します。

【書式】ldm add-domain <新規に作成するドメイン名>

[SF] ldm add-domain → ldm create

```
primary# ldm add-domain ldom1
```

3.1.2. リソースの割り当て

ゲストドメインが利用する CPU、メモリ、および仮想 I/O デバイスなどを割り当てます。

1) ゲストドメインに割り当てる CPU のコア数を指定します。

【書式】ldm set-core <CPU コア数> <ドメイン名>

```
primary# ldm set-core 1 ldom1
```

《参考》「スレッド数指定」による CPU 割り当て

ゲストドメインに割り当てる CPU のスレッド数を指定します。

```
primary# ldm set-vcpu 4 ldom1
```

▶ SPARC M12 の場合、4 スレッド単位で割り当ててください (SPARC M10 の場合は、1 スレッド単位で問題ありません)。

2) 割り当てるメモリ容量を指定します。

【書式】ldm set-memory [--auto-adj] <メモリ容量> <ドメイン名>

[SF] ldm set-memory → ldm set-mem

```
primary# ldm set-memory 16g ldom1
```

3) 仮想ネットワークデバイス(仮想 NIC)を割り当てます。

【書式】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>]

<仮想ネットワークインターフェース名> <仮想スイッチ名> <ドメイン名>

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


4) 仮想コンソールポートを割り当てます。

【書式】ldm set-vconsole [port=<port-number>][group=group][service=vcc-server][log=<on|off>] <ドメイン名>

SF ldm set-vconsole → ldm set-vcons

```
primary# ldm set-vconsole port=5000 ldom1
```

5) 仮想ディスクを割り当てます。

【書式】ldm add-vdisk [timeout=<seconds>] [id=<diskid>]<仮想ディスク名> <仮想ボリューム名>@<仮想ディスクサービス名> <ドメイン名>

```
primary# ldm add-vdisk vdisk0 vol0@primary-vds0 ldom1
primary# ldm add-vdisk vdisk_iso vol_iso@primary-vds0 ldom1
```

3.1.3. 環境パラメーターの設定

OS の自動起動の有無や、ブートデバイスなどのパラメーターを設定します。

1) auto-boot の設定（ドメイン起動時に自動で OS を起動するかどうか）を実施します。

【書式】ldm set-variable auto-boot?=<false | true> <ドメイン名>

SF ldm set-variable → ldm set-var

```
primary# ldm set-variable auto-boot?=false ldom1
Variables or keys have been set and those settings will be lost over a powercycle. To
persist variables or keys over a powercycle, a configuration must be saved to the SP
after updates.
```

- ▶ 「設定変更後は構成情報を SP（サービスプロセッサ）に保存してください」という内容のメッセージが表示されます。
なお、メッセージが表示されない場合もありますが、問題ありません。
- ▶ [手順 6](#) で構成情報を SP（サービスプロセッサ）に保存します。

2) ブートデバイスを設定します。

【書式】ldm set-variable boot-device=<仮想ディスク> <ドメイン名>

```
primary# ldm set-variable boot-device=vdisk0 ldom1
Variables or keys have been set and those settings will be lost over a powercycle. To
persist variables or keys over a powercycle, a configuration must be saved to the SP
after updates.
```

- ▶ メッセージが表示されない場合もありますが、問題ありません。

3) リソースやパラメーターなどを確認します。

```
primary# ldm list-bindings ldom1
NAME          STATE    FLAGS    CONS    VCPU    MEMORY    UTIL  NORM  UPTIME
ldom1         inactive -----      8      16G
~ (省略) ~

VARIABLES
auto-boot?=false
boot-device=vdisk0

NETWORK
NAME          SERVICE          MACADDRESS          PVID|PVLAN|VIDs
-----
vnet0         primary-vsw0      00:14:4f:fa:e9:94    |--|--

DISK
NAME          VOLUME          TOUT ID    DEVICE  SERVER          MPGROUP
vdisk0        vol0@primary-vds0      0
vdisk_iso     vol_iso@primary-vds0    1

VCONS
NAME          SERVICE          PORT  LOGGING
5000
```

4) ゲストドメインをバインドします。

【書式】ldm bind-domain [-f] [-q] (-i <ファイル名> | <ドメイン名>)

[SF] ldm bind-domain → ldm bind

```
primary# ldm bind-domain ldom1
```

5) ゲストドメインの状態を確認します。

STATE が「bound」になっていることを確認してください。

```
primary# ldm list-domain
NAME          STATE    FLAGS    CONS    VCPU    MEMORY    UTIL  NORM  UPTIME
primary       active   -n-cv-   UART    16      16G      0.0%  0.0%  30m
ldom1         bound    -----  5000    8      16G
```

6) 構成情報を更新します。

既存の構成情報を削除したあと、最新の構成情報を SP（サービスプロセッサ）に保存します。

```
primary# ldm remove-spconfig config_initial
primary# ldm add-spconfig config_initial
```

3.1.4. ゲストドメインの起動

定義したリソースを結合し、ゲストドメインを起動します。

1) ゲストドメインを起動します。

【書式】ldm start-domain (-a | -i <file> | [-f] [-m] <ドメイン名>...)

SF ldm start-domain → ldm start

```
primary# ldm start-domain ldom1
LDom ldom1 started
```

2) 再度、ゲストドメインの状態を確認します。

```
primary# ldm list-domain
NAME          STATE      FLAGS    CONS    VCPU  MEMORY  UTIL  NORM  UPTIME
primary       active     -n-cv-   UART    16    16G     0.1%  0.1%  32m
ldom1         active     -t----- 5000    8     16G     12%   12%   17s
```

3) ゲストドメインに接続できることを確認します。

【書式】telnet localhost <ドメインのコンソールポート番号>

```
primary# telnet localhost 5000
```

以下のメッセージが出力されたあと、[Enter]キーを押すと、ok プロンプトが表示されます。

```
Connecting to console "ldom1" in group "ldom1" ....
Press ~? for control options ..
```

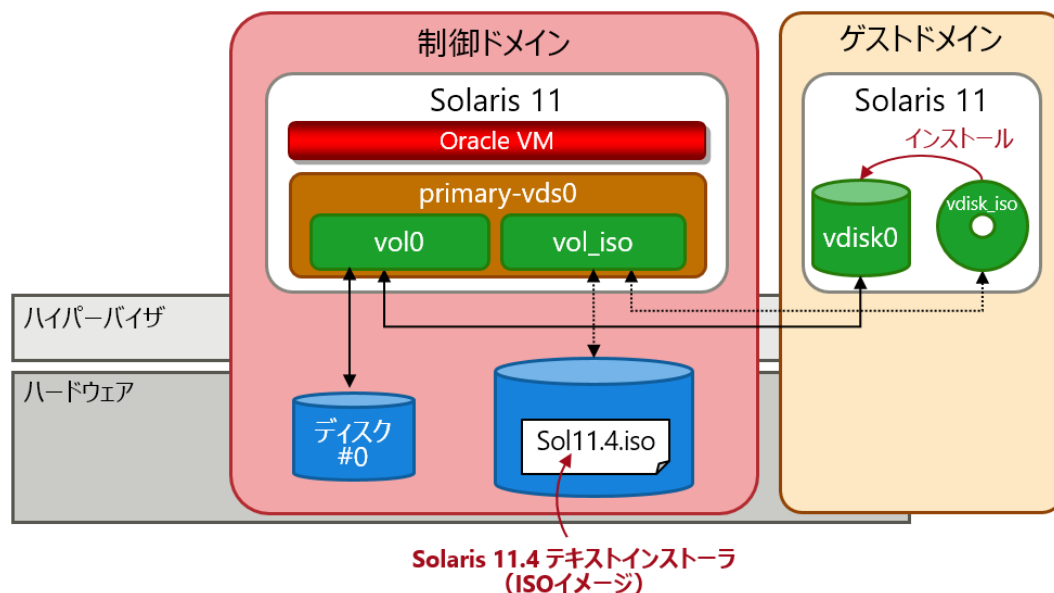
▶ ok プロンプトが表示されれば接続は成功です。

《参考》 ゲストドメインからのコンソールログアウト方法

ゲストドメインのコンソール環境から、制御ドメインへ戻るには、~(チルダ)+. (ドット)を入力します。「~.」は表示されません。

3.1.5. ゲストドメインへの OS インストール

ゲストドメインの仮想ディスクに OS をインストールします。



以下の 3 種類のインストール方法のうち、本書では ISO イメージによるインストールを実施します。

- ISO イメージによるインストール
- DVD メディアによるインストール
- インストールサーバによるネットワークインストール

1) ゲストドメインに接続します。

```
primary# telnet localhost 5000
```

2) デバイスを確認します。

ldm add-vdisk で割り当てた仮想ディスク (vdisk_iso、vdisk0) と、ldm add-vnet で割り当てた仮想 NIC (vnet0) が表示されることを確認します。

```
{0} ok devalias
vdisk_iso      /virtual-devices@100/channel-devices@200/disk@1
vdisk0         /virtual-devices@100/channel-devices@200/disk@0
vnet0          /virtual-devices@100/channel-devices@200/network@0
net            /virtual-devices@100/channel-devices@200/network@0
disk           /virtual-devices@100/channel-devices@200/disk@0
virtual-console /virtual-devices/console@1
name           aliases
```

3) ゲストドメインに OS をインストールします。

Point

OS インストールを開始する前に、文字コードが「UTF-8」に設定されていることを確認してください。
TeraTerm をお使いの方は、Setup[設定]→Terminal[端末]の Kanji(Receive)[漢字-受信]と
Kanji(Transmit)[漢字-送信]の項目で、設定できます。

なお、文字化けが生じた場合は、ターミナルソフトの文字コードを UTF-8 に変更したあと、[Esc]+
[3]キーを押してください。

```
{0} ok boot vdisk_iso
```

▶ 通常の OS インストールと同様に行います。

《参考》 警告メッセージが表示される場合

OS のバージョンによって、以下のメッセージが表示される場合があります。影響はありませんので
無視してください。

```
WARNING: Bootblk signature verification failed  
WARNING: Can not verify signature on unsupported binary format  
WARNING: Signature verification of boot-archive bootblk failed  
WARNING: Signature verification of UNIX kernel image failed
```

4) ゲストドメインのコンソールに一般ユーザでログインして、root 権限に切り替えます。

OS インストール時に設定したユーザとユーザパスワードで、ゲストドメインにログインします。

```
ldom1 console login: user-id  
Password:*****  
ldom1$ su -  
Password:*****
```

3.1.6. ESF、パッチの適用

通常の OS と同様に SRU (Support Repository Update) を適用し、ESF (Enhanced Support Facility)
などの必要なパッケージをインストールします。
詳細は、以下を参照してください。

- 『Oracle Solaris 11 を使ってみよう(構築・運用手順書)』
<https://www.fujitsu.com/jp/sparc/technical/document/solaris/#os>

3.1.7. 構築後のゲストドメインの環境確認

ゲストドメイン 1 が正しく構築できたことを確認します。

- 1) `/etc/path_to_inst` ファイルに `vnet` が存在することを確認します。

```
ldom1# grep vnet /etc/path_to_inst
"/virtual-devices@100/channel-devices@200/network@0" 0 "vnet"
```

- 2) ゲストドメインが認識しているネットワークインターフェースを確認します。

```
ldom1# dladm show-phys -L
LINK          DEVICE      LOC
net0          vnet0       --
```

- 3) ゲストドメインの IP アドレスを確認します。

```
ldom1# ipadm show-addr
ADDROBJ      TYPE      STATE      ADDR
lo0/v4       static    ok         127.0.0.1/8
net0/v4       static    ok         192.168.10.201/24
lo0/v6       static    ok         ::1/128
net0/v6       addrconf  ok         fe80::214:4fff:fe8:2388/10
```

- 4) ゲストドメインから同一サブネット内のサーバへアクセスできることを確認します。

```
ldom1# ping 192.168.10.xxx
```

- 5) 仮想 CPU の数を確認します。

```
ldom1# psrinfo -vp
The physical processor has 8 virtual processors (0-7)
SPARC64-XII (chipid 0, clock 3200 MHz)
```

- 6) メモリ容量を確認します。

```
ldom1# prtconf -v | head
System Configuration: Oracle Corporation sun4v
Memory size: 16384 Megabytes
System Peripherals (Software Nodes):
~ (省略) ~
```

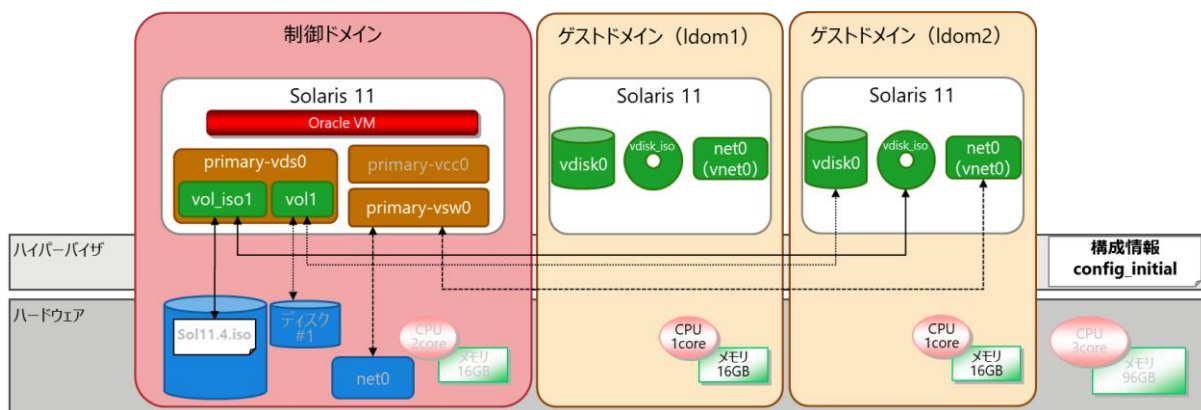
- 7) ディスクデバイスを確認します。

```
ldom1# format
```

- ▶ ゲストドメインに割り当てた仮想ディスクを確認できます。
- ▶ 仮想ディスクにはターゲット ID はありません。デバイス名は、「cXtXdX」ではなく、「cXdX」の形式になります。
- ▶ [Ctrl]+[C]キーで format メニューを終了します。

3.2. ゲストドメイン 2 の構築

ゲストドメイン 1 (ldom1) と同様に、ゲストドメイン 2 (ldom2) を構築します。



- 1) ゲストドメインを作成します。

```
primary# ldm add-domain ldom2
```

- 2) リソース(CPU 数とメモリ容量)を設定します。

```
primary# ldm set-core 1 ldom2
primary# ldm set-memory 16g ldom2
```

- 3) 仮想ネットワークデバイス(仮想 NIC)を割り当てます。

ここでは、ldom1 構築時と同じ仮想ネットワークサービス(primary-vsw0)を使用します。

```
primary# ldm add-vnet vnet0 primary-vsw0 ldom2
```

- 4) 仮想コンソールポートを割り当てます。

ldom1 と別のポート番号を指定します。

```
primary# ldm set-vconsole port=5001 ldom2
```

- 5) 仮想ディスクサービスにディスクを割り当てたあと、ゲストドメインに割り当てます。

ここでは、ldom1 構築時と同じ仮想ディスクサービス(primary-vds0)を使用します。ldom1 と別の物理ディスクを割り当てます。

```
primary# ldm add-vdiskserverdevice /dev/dsk/c0t600000E00D0000000001034600010000d0s2
vol1@primary-vds0
primary# ldm add-vdisk vdisk0 vol1@primary-vds0 ldom2
```

▶ 仮想ディスクサービスに物理ディスク全体を割り当ててる場合は、s2 を指定します。

- 6) 仮想ディスクサービスに ISO イメージ(Solaris のインストール DVD)を「Read Only」で割り当てたあと、ゲストドメインに割り当てます。

```
primary# ldm add-vdiskserverdevice options=ro /ISO/Sol11.4.iso vol_iso1@primary-vds0
primary# ldm add-vdisk vdisk_iso vol_iso1@primary-vds0 ldom2
```

7) **auto-boot** の設定 (ドメイン起動時に自動で OS を起動するか) を実施します。

```
primary# ldm set-variable auto-boot?=false ldom2
```

Variables or keys have been set and those settings will be lost over a powercycle. To persist variables or keys over a powercycle, a configuration must be saved to the SP after updates.

▶ メッセージが表示されない場合もありますが、問題ありません。

8) ゲストドメインのブートデバイスを設定します。

```
primary# ldm set-variable boot-device=vdisk0 ldom2
```

Variables or keys have been set and those settings will be lost over a powercycle. To persist variables or keys over a powercycle, a configuration must be saved to the SP after updates.

▶ メッセージが表示されない場合もありますが、問題ありません。

9) ゲストドメインのリソースやパラメーターなどを確認します。

```
primary# ldm list-bindings ldom2
```

NAME	STATE	FLAGS	CONS	VCPU	MEMORY	UTIL	NORM	UPTIME
ldom2	inactive	-----		8	16G			

～（省略）～

VARIABLES

```
auto-boot?=false
boot-device=vdisk0
```

NETWORK

NAME	SERVICE	MACADDRESS	PVID PVLAN VIDs
vnet0	primary-vsw0	00:14:4f:f8:61:8f	-- --

DISK

NAME	VOLUME	TOUT	ID	DEVICE	SERVER	MPGROUP
vdisk0	vol1@primary-vds0		0			
vdisk_iso	vol_iso1@primary-vds0		1			

VCONS

NAME	SERVICE	PORT	LOGGING
		5001	

10) ゲストドメインをバインドします。

```
primary# ldm bind-domain ldom2
```

11) 構成情報を更新します。

```
primary# ldm remove-spconfig config_initial
primary# ldm add-spconfig config_initial
```


12) ゲストドメインを起動します。

```
primary# ldmd start-domain ldom2
```

13) ゲストドメイン 1 (ldom1)と同様に OS をインストールします。また、SRU (Support Repository Update)を適用し、ESF (Enhanced Support Facility)などの必要なパッケージをインストールします。

操作方法については、以下を参照してください。

「[3.1.5. ゲストドメインへの OS インストール](#)」

「[3.1.6. ESF、パッチの適用](#)」

「[3.1.7. 構築後のゲストドメインの環境確認](#)」

4. ネットワークの冗長化

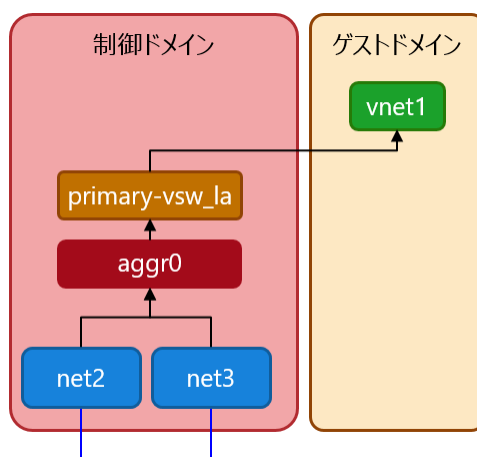
ゲストドメインのネットワーク(仮想 NIC)を冗長化するには、以下の 2 つの方法があります。

- 制御ドメインでの冗長化
- ゲストドメインでの冗長化

4.1. 制御ドメインでの冗長化

本書では、リンクアグリゲーションを使用し、制御ドメインでネットワークを冗長化します。

リンクアグリゲーションを使用して作成したネットワークインターフェース(aggrX)を、仮想スイッチ(vsw)として使用できます。



4.1.1. リンクアグリゲーションの設定(制御ドメイン上)

1) ネットワークの設定を確認します。

本手順では、「ixgbe2(net2)」と「ixgbe3(net3)」でリンクアグリゲーションを構成します。

```

primary# dladm show-phys -L
LINK          DEVICE      LOC
net0          ixgbe0      /SYS/MBU/CMPO/PEX0
net1          ixgbe1      /SYS/MBU/CMPO/PEX0
net2          ixgbe2      /SYS/MBU/CMPO/PEX2
net3          ixgbe3      /SYS/MBU/CMPO/PEX2
~ (省略) ~
  
```

2) リンクアグリゲーションを設定する NIC が使用されていないことを確認します。

NIC が表示されない、または STATE が「down」であることを確認してください。

```
primary# ipadm show-if
IFNAME    CLASS    STATE    ACTIVE OVER
lo0       loopback ok       yes    --
net0      ip       ok       yes    --
```

▶ STATE が「ok」の場合、NIC が使用されています。

Point

NIC が使用されている場合は、IP インターフェースを削除(delete-ip)するか、またはほかの NIC を使用する必要があります。

3) リンクアグリゲーションを設定してデバイスを作成します。

本書では、データリンクマルチパスモード(DLMP)に設定します。-m オプションで「dlmp」を指定します。

【書式】dladm create-aggr [-t] [-R root-dir] [-m mode] [-P policy] [-L lcapmode] [-T time] [-u address] -l ether-link1 [-l ether-link2...] <aggr-link>

```
primary# dladm create-aggr -m dlmp -l net2 -l net3 aggr0
```

4) 作成したデバイスを確認します。

i) リンクアグリゲーションが設定されていることを確認します。

CLASS に「aggr」（リンクアグリゲーション）、OVER に「リンクアグリゲーションに設定した NIC」が表示されていることを確認してください。

```
primary# dladm show-link
LINK          CLASS    MTU    STATE    OVER
aggr0         aggr     1500   up       net2 net3
~ (省略) ~
```

ii) リンクアグリゲーション設定元の NIC の状態を確認します。

net2 と net3 が「attached」であることを確認してください。

```
primary# dladm show-aggr -x
LINK    PORT    SPEED DUPLEX    STATE    ADDRESS          PORTSTATE
aggr0   --      1000Mb full   up       2:8:20:30:9a:37  --
        net2    1000Mb full   up       b0:99:28:99:e1:c2 attached
        net3    1000Mb full   up       b0:99:28:99:e1:c3 attached
```

4.1.2. 仮想スイッチサービスの作成(制御ドメイン上)

- 1) 作成したデバイスから、仮想スイッチサービスを作成します。

```
primary# ldem add-vswitch net-dev=aggr0 primary-vsw_la primary
```

- 2) 仮想ネットワークデバイス(仮想 NIC)を割り当てます。

```
primary# ldem add-vnet vnet1 primary-vsw_la ldom1
```

4.1.3. ゲストドメイン上の設定

- 1) ldom1 上で、仮想ネットワークデバイスが追加されたことを確認します。

vnet1(net1)が追加されていることを確認してください。

```
ldom1# dladm show-phys -L
LINK          DEVICE      LOC
net0          vnet0       --
net1          vnet1       --
```

- 2) IP アドレスを設定します。

```
ldom1# ipadm create-ip net1
ldom1# ipadm create-addr -T static -a 10.10.100.201/24 net1/v4
```

- 3) ネットワークの設定を確認します。

```
ldom1# ipadm show-addr
ADDROBJ      TYPE      STATE      ADDR
lo0/v4       static    ok         127.0.0.1/8
net1/v4       static    ok         10.10.100.201/24
~ (省略) ~
```

- 4) ゲストドメインから外部サーバへアクセスできることを確認します。

```
ldom1# ping 10.10.100.xxx
```

4.1.4. 冗長化設定の削除

- 1) ldom1 の net1 の IP アドレスを削除します。

net1 が削除されていることを確認してください。

```
ldom1# ipadm delete-ip net1
ldom1# ipadm show-addr
ADDROBJ      TYPE      STATE      ADDR
lo0/v4       static    ok         127.0.0.1/8
net0/v4       static    ok         192.168.10.201/24
lo0/v6       static    ok         ::1/128
net0/v6       addrconf  ok         fe80::214:4fff:fe8:2388/10
```

2) 仮想ネットワークデバイス(仮想 NIC)の割り当てを解除します。

```
primary# ldm remove-vnet vnet1 ldom1
```

3) 仮想スイッチサービスを削除します。

```
primary# ldm remove-vswitch primary-vsw_la
```

4) ldom1 上で、仮想ネットワークデバイスが削除されたことを確認します。

vnet1(net1)が削除されていることを確認してください。

```
ldom1# dladm show-phys -L
LINK          DEVICE      LOC
net0          vnet0       --
```

5) リンクアグリゲーションの設定を解除します。

Point

リンクアグリゲーションを使用して作成したデバイス(aggr0)に IP アドレスを設定している場合は、事前にインターフェースの削除(ipadm delete-ip aggr0)を実行してください。

```
primary# dladm delete-aggr aggr0
```

6) デバイスを確認します。

i) リンクアグリゲーションが解除されていることを確認します。

aggr0 が表示されないことを確認してください。

```
primary# dladm show-link
LINK          CLASS      MTU    STATE    OVER
net0          phys      1500   up       --
~ (省略) ~
```

ii) リンクアグリゲーション設定元の NIC の状態を確認します。

コマンド実行結果が何も表示されないことを確認してください。

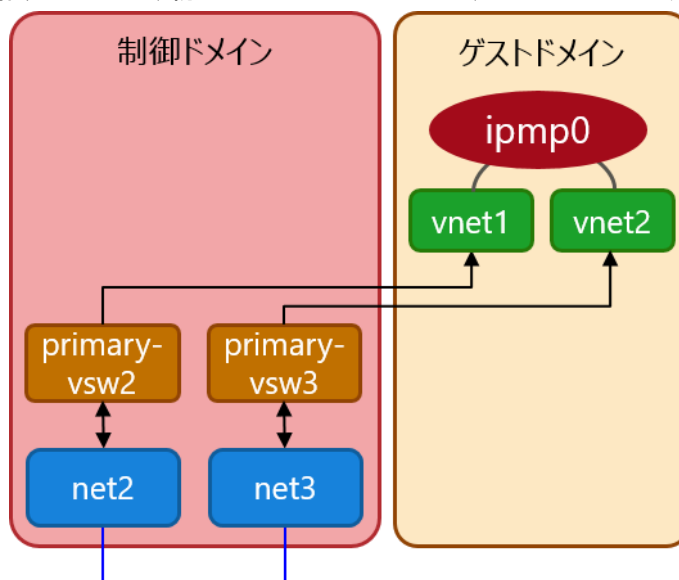
```
primary# dladm show-aggr -x
primary# ←何も表示されません。
```

4.2. ゲストドメインでの冗長化

本書では、IPMP（リンクベース）を使用してゲストドメインでネットワークを冗長化します。

ゲストドメイン上で IPMP（リンクベース）を構築する場合、仮想ネットワークデバイスを割り当てるときに、linkprop オプションを指定します。

linkprop オプションの指定によって、物理 NIC のステータスが、ゲストドメインに通知されます。



4.2.1. 仮想スイッチの作成(制御ドメイン上)

1) 制御ドメインで、物理ネットワークデバイスを確認します。

本手順では、ixgbe2 (net2) と ixgbe3 (net3) を使用し、ゲストドメイン上で IPMP を構成します。

```
primary# dladm show-phys -L
LINK          DEVICE          LOC
net0           ixgbe0           /SYS/MBU/CMPO/PEX0
net1           ixgbe1           /SYS/MBU/CMPO/PEX0
net2           ixgbe2           /SYS/MBU/CMPO/PEX2
net3           ixgbe3           /SYS/MBU/CMPO/PEX2
~ (省略) ~
```

2) IPMP を設定する NIC が使用されていないことを確認します。

NIC が表示されない、または STATE が「down」であることを確認してください。

```
primary# ipadm show-if
IFNAME  CLASS  STATE  ACTIVE  OVER
lo0     loopback ok      yes     ---
net0    ip      ok      yes     ---
net1    ip      down    no      ---
net2    ip      down    no      ---
net3    ip      down    no      ---
```

▶ STATE が「ok」の場合、NIC が使用されています。

3) IPMP 構築用仮想スイッチを作成します。

```
primary# ldm add-vswitch net-dev=net2 primary-vsw2 primary
primary# ldm add-vswitch net-dev=net3 primary-vsw3 primary
```

4.2.2. 仮想ネットワークデバイスの割り当て(制御ドメイン上)

1) 仮想ネットワークデバイス(vnet1 と vnet2)を作成します。

```
primary# ldm add-vnet linkprop=phys-state vnet1 primary-vsw2 ldom1
primary# ldm add-vnet linkprop=phys-state vnet2 primary-vsw3 ldom1
```

4.2.3. IPMP の設定(ゲストドメイン上)

1) ldom1 のネットワークインターフェースの構成を確認します。

net1 と net2 の 2 つのデバイスを認識していることを確認してください。

```
ldom1# dladm show-phys -L
LINK          DEVICE      LOC
net0          vnet0       --
net1          vnet1       --
net2          vnet2       --
```

2) ネットワークの設定を確認します。

```
ldom1# ipadm show-addr
ADDROBJ      TYPE      STATE      ADDR
lo0/v4       static    ok         127.0.0.1/8
net0/v4       static    ok         192.168.10.201/24
lo0/v6       static    ok         ::1/128
net0/v6       addrconf ok         fe80::214:4fff:fe8:2388/10
```

3) net1 と net2 を有効化します。

```
ldom1# ipadm create-ip net1
ldom1# ipadm create-ip net2
```

4) net1 と net2 で IPMP を構築します。

```
ldom1# ipadm create-ipmp -i net1 -i net2 ipmp0
ldom1# ipmpstat -i
INTERFACE    ACTIVE    GROUP      FLAGS      LINK      PROBE      STATE
net2         yes      ipmp0      -----    up        disabled    ok
net1         yes      ipmp0      --mbM--    up        disabled    ok
```

5) IPMP の IP アドレスを設定します。

```
ldom1# ipadm create-addr -T static -a local=10.10.100.201/24 ipmp0/v4
ldom1# ipadm show-addr
```

ADDROBJ	TYPE	STATE	ADDR
lo0/v4	static	ok	127.0.0.1/8
~ (省略) ~			
ipmp0/v4	static	ok	10.10.100.201/24
~ (省略) ~			

6) 外部サーバへアクセスできることを確認します。

```
ldom1# ping 10.10.100.xxx
10.10.100.xxx is alive
```

4.2.4. 冗長化設定の削除

1) 削除する IPMP を確認します。

```
ldom1# ipmpstat -g
```

GROUP	GROUPNAME	STATE	FDT	INTERFACES
ipmp0	ipmp0	ok	—	net2 net1

2) IPMP に設定しているネットワークインターフェースを削除します。

IPMP に設定しているすべての IP インターフェースが使用できなくなったという意味のメッセージが表示されます。

```
ldom1# ipadm remove-ipmp -i net1 -i net2 ipmp0
Oct 24 13:11:26 guest1 in.mpathd[95]: All IP interfaces in group ipmp0 are now
unusable
```

3) IPMP を削除し、IPMP が表示されないことを確認します。

```
ldom1# ipadm delete-ipmp ipmp0
ldom1# ipmpstat -g
```

4) net1 と net2 を無効化します。

```
ldom1# ipadm delete-ip net1
ldom1# ipadm delete-ip net2
```

5) ネットワークの設定を確認します。

ipmp0 が削除されていることを確認してください。

```
ldom1# ipadm show-addr
```

ADDROBJ	TYPE	STATE	ADDR
lo0/v4	static	ok	127.0.0.1/8
net0/v4	static	ok	192.168.10.201/24
lo0/v6	static	ok	:::1/128
net0/v6	addrconf	ok	fe80::214:4fff:fe8:c33d/10

6) 仮想ネットワークデバイス(仮想 NIC)の割り当てを解除します。

```
primary# ldm remove-vnet vnet1 ldom1
primary# ldm remove-vnet vnet2 ldom1
```

7) 仮想スイッチサービスを削除します。

```
primary# ldm remove-vswitch primary-vsw2
primary# ldm remove-vswitch primary-vsw3
```

8) ldom1 上で、仮想ネットワークデバイスが削除されたことを確認します。

vnet1(net1)と vnet2(net2)が削除されていることを確認してください。

```
ldom1# dladm show-phys -L
LINK          DEVICE      LOC
net0          vnet0        --
```

9) 使用した物理 NIC を無効化します。

```
primary# ipadm delete-ip net2
primary# ipadm delete-ip net3
```

10) 無効になっていることを確認します。

net2 と net3 が表示されないことを確認してください。

```
primary# ipadm show-if
IFNAME  CLASS  STATE  ACTIVE OVER
lo0     loopback ok      yes   --
net0    ip      ok      yes   --
net1    ip      down    no    --
```

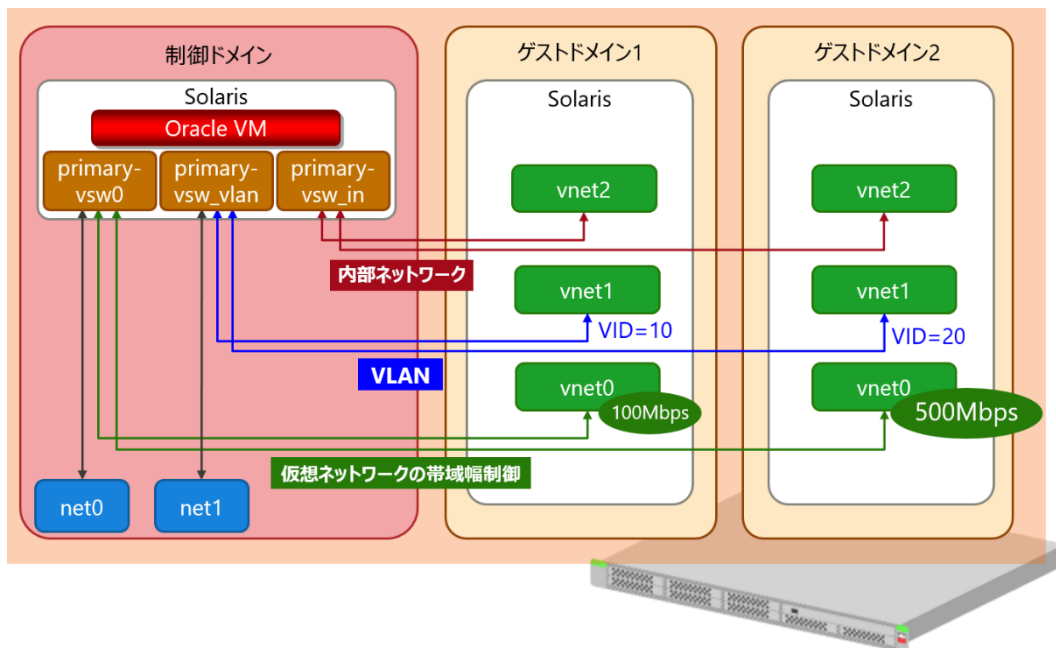
Point

Oracle VM の設定・構成を変更した場合は、以下のコマンドを実行して必ず構成情報を更新してください。

```
primary# ldm remove-spconfig config_initial
primary# ldm add-spconfig config_initial
```

5. 仮想ネットワークの応用

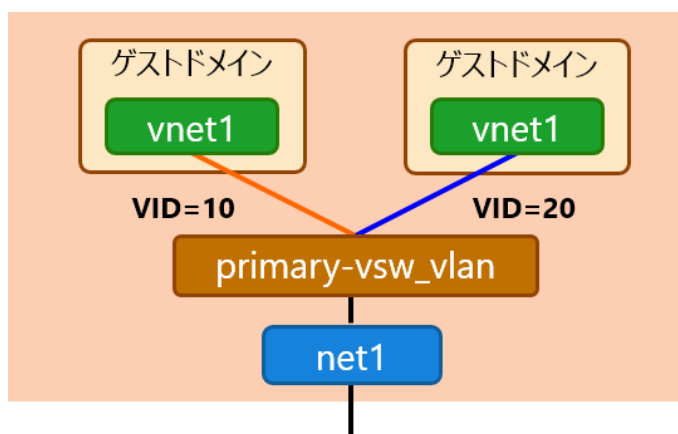
本章で作成するネットワーク構成を図に示します。



5.1. VLAN によるネットワークの統合

仮想ネットワーク(vnet)で VLAN の設定をすることにより、1 つの物理 NIC で異なるサブネットのネットワークを統合できます。

5.1.1. 異なるサブネットの業務を 1 つの NIC で統合



- 1) 仮想スイッチサービスを作成します。

```
primary# ldm add-vswitch net-dev=net1 primary-vsw_vlan primary
```

2) 仮想ネットワークデバイス(仮想 NIC)を割り当てます。

```
primary# ldm add-vnet vnet1 primary-vsw_vlan ldom1
primary# ldm add-vnet vnet1 primary-vsw_vlan ldom2
```

3) ldom1 の仮想 NIC(vnet1)に、VLAN ID「10」を設定します。

【書式】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-m ac-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>] <仮想ネットワークインターフェース名> <ドメイン名>

```
primary# ldm set-vnet pvid=10 vnet1 ldom1
```

4) ldom2 の仮想 NIC(vnet1)に、VLAN ID「20」を設定します。

```
primary# ldm set-vnet pvid=20 vnet1 ldom2
```

5) VLAN ID が設定されていることを確認します。

```
primary# ldm list-domain -o network
~ (省略) ~
ldom1
. . .
NAME          SERVICE          MACADDRESS          PVID|PVLAN|VIDs
----          -
vnet1         primary-vsw_vlan@primary 00:14:4f:f9:88:16  10|---|---
ldom2
. . .
NAME          SERVICE          MACADDRESS          PVID|PVLAN|VIDs
----          -
vnet1         primary-vsw_vlan@primary 00:14:4f:fb:31:53  20|---|---
```

6) ldom1 上で、仮想ネットワークデバイスが追加されたことを確認します。

vnet1(net1)が追加されていることを確認してください。

```
ldom1# dladm show-phys -L
LINK          DEVICE          LOC
net0          vnet0          --
net1          vnet1          --
```

7) ldom1 の net1 に IP アドレスを設定します。

```
ldom1# ipadm create-ip net1
ldom1# ipadm create-addr -T static -a 192.168.11.201/24 net1/v4
```

8) ldom1 の IP アドレスを確認します。

```
ldom1# ipadm show-addr
ADDROBJ      TYPE      STATE      ADDR
lo0/v4        static    ok          127.0.0.1/8
net0/v4        static    ok          192.168.10.201/24
net1/v4        static    ok          192.168.11.201/24
lo0/v6        static    ok          ::1/128
net0/v6        addrconf ok          fe80::214:4fff:fe8:2388/10
```

9) ldom2 上で、仮想ネットワークデバイスが追加されたことを確認します。

```
ldom2# dladm show-phys -L
LINK      DEVICE      LOC
net0       vnet0        --
net1       vnet1        --
```

10) ldom2 の net1 に IP アドレスを設定します。

```
ldom2# ipadm create-ip net1
ldom2# ipadm create-addr -T static -a 192.168.11.202/24 net1/v4
```

11) ldom2 の IP アドレスを確認します。

```
ldom2# ipadm show-addr
ADDROBJ      TYPE      STATE      ADDR
lo0/v4        static    ok          127.0.0.1/8
net0/v4        static    ok          192.168.10.202/24
net1/v4        static    ok          192.168.11.202/24
lo0/v6        static    ok          ::1/128
net0/v6        addrconf ok          fe80::214:4fff:fe8:77e7/10
```

12) ldom2 から ldom1 へ ping を実行します。VLAN ID が異なるため、疎通できないことを確認します。

```
ldom2# ping 192.168.11.201
no answer from 192.168.11.201
```

5.1.2. VLAN 設定の解除

1) ldom1 の仮想 NIC(vnet1)の VLAN 設定を解除します。

pvid に何も設定しないでコマンドを実行します。

```
primary# ldm set-vnet pvid= vnet1 ldom1
```

2) ldom2 の仮想 NIC(vnet1)の VLAN 設定を解除します。

```
primary# ldm set-vnet pvid= vnet1 ldom2
```

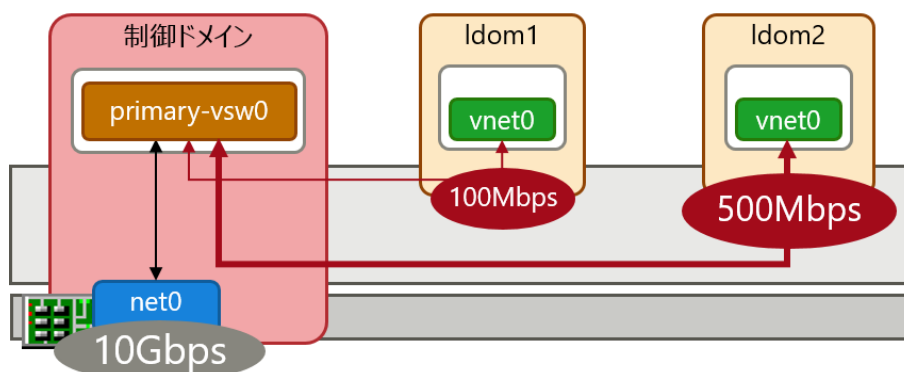
3) VLAN ID の設定が解除されていることを確認します。

VLAN の設定がされていない状態（デフォルト）では、PVID が「1」となっています。

```
primary# ldm list-domain -o network
~ (省略) ~
ldom1
. . .
NAME          SERVICE          MACADDRESS          PVID|PVLAN|VIDs
-----
vnet1         primary-vsw_vlan@primary 00:14:4f:f9:88:16   1|--|--
ldom2
. . .
NAME          SERVICE          MACADDRESS          PVID|PVLAN|VIDs
-----
vnet1         primary-vsw_vlan@primary 00:14:4f:fb:31:53   1|--|--
```

5.2. 仮想ネットワークの帯域制御

仮想ネットワークデバイスごとに、帯域幅の上限を設定できます。



1) ldom1 と ldom2 の仮想 NIC (vnet0) に帯域幅制限を設定します。

ここでは、ldom1 の vnet0 を 100 Mbps、ldom2 の vnet0 を 500 Mbps に制限します。

```
primary# ldm set-vnet maxbw=100m vnet0 ldom1
primary# ldm set-vnet maxbw=500m vnet0 ldom2
```

2) ldom1 と ldom2 の vnet0 に帯域幅の制限が設定されていることを確認します。

```
primary# ldm list-domain -o network
~ (省略) ~
NAME
ldom1

MAC
00:14:4f:fa:27:cc

NETWORK
NAME      SERVICE      MACADDRESS      PVID|PVLAN|VIDs
-----
vnet0     primary-vsw0@primary  00:14:4f:fa:e9:94  1|--|--
          DEVICE      :network@0      ID      :0
          LINKPROP   :phys-state     MTU     :1500
          MAXBW      :100M
~ (省略) ~
NAME
ldom2

MAC
00:14:4f:f8:0d:d9

NETWORK
NAME      SERVICE      MACADDRESS      PVID|PVLAN|VIDs
-----
vnet0     primary-vsw0@primary  00:14:4f:f8:61:8f  1|--|--
          DEVICE      :network@0      ID      :0
          LINKPROP   :phys-state     MTU     :1500
          MAXBW      :500M
~ (省略) ~
```

3) 設定した帯域幅制限の設定をクリアします。

maxbw に何も設定しないでコマンドを実行します。

```
primary# ldm set-vnet maxbw= vnet0 ldom1
primary# ldm set-vnet maxbw= vnet0 ldom2
```

4) ldom1 と ldom2 の帯域幅制限の設定が削除されていることを確認します。

```
primary# ldm list-domain -o network
~ (省略) ~
NAME
ldom1

MAC
00:14:4f:fa:27:cc

NETWORK
NAME      SERVICE      MACADDRESS      PVID|PVLAN|VIDs
-----
vnet0     primary-vsw0@primary  00:14:4f:fa:e9:94  1|---|---
          DEVICE      :network@0      ID      :0
          LINKPROP   :phys-state    MTU     :1500
          MAXBW      :---
          MODE      :---
~ (省略) ~
NAME
ldom2

MAC
00:14:4f:f8:0d:d9

NETWORK
NAME      SERVICE      MACADDRESS      PVID|PVLAN|VIDs
-----
vnet0     primary-vsw0@primary  00:14:4f:f8:61:8f  1|---|---
          DEVICE      :network@0      ID      :0
          LINKPROP   :phys-state    MTU     :1500
          MAXBW      :---
          MODE      :---
~ (省略) ~
```

《参考》 iperf3 を使用した帯域幅制限設定の確認

ネットワーク性能計測ソフト「iperf3」を使用し、スループット性能(1 秒あたりに転送可能なデータ容量)を計測できます。

※ iperf3 は、Solaris 11.4 のリポジトリからインストールできます。

※ iperf3 は、Solaris 11.4 から使用可能です。Solaris 11.3 以前は、iperf を使用します。

ここでは、制御ドメイン(primary)からゲストドメイン(ldom1)の転送速度を計測し、帯域幅制限設定が有効になっていることを確認します。

送信先と送信元それぞれで実行します。

※ オプションおよび設定値は例です。

(1) 送信先(ldom1)で iperf3 を実行します。

```
ldom1# iperf3 -s -p 6000
```

(2) 送信元(primary)で iperf3 を実行します。

```
primary# iperf3 -c 192.168.10.101 -t 20 -fm -i 4 -p 6000 -l 1024K -w 1024K -P1 -M 1472
```

● 帯域幅制限設定なしの場合

本環境の場合、送信先／送信元ともに、転送速度が**約 5.1 Gbps** となっています。

● 送信先(ldom1)

```
ldom1# iperf3 -s -p 6000
```

```
Server listening on 6000
```

```
Accepted connection from 192.168.10.101, port 43035
```

```
[ 5] local 192.168.10.201 port 6000 connected to 192.168.10.101 port 43207
```

```
[ ID] Interval          Transfer      Bandwidth
```

```
~ (省略) ~
```

```

[ ID] Interval          Transfer      Bandwidth
[ 5]  0.00-20.01 sec  11.8 GBytes  5.08 Gbits/sec
[ 5]  0.00-20.01 sec  11.8 GBytes  5.08 Gbits/sec

```

● 送信元(primary)

```
primary# iperf3 -c 192.168.10.201 -t 20 -fm -i 4 -p 6000 -l 1024K -w 1024K -P1 -M 1472
```

```
Connecting to host 192.168.10.201, port 6000
```

```
[ 4] local 192.168.10.101 port 43207 connected to 192.168.10.201 port 6000
```

```
[ ID] Interval          Transfer      Bandwidth
```

```
[ 4]  0.00-4.00 sec  2.27 GBytes  4868 Mbits/sec
```

```
[ 4]  4.00-8.00 sec  2.38 GBytes  5105 Mbits/sec
```

```
[ 4]  8.00-12.00 sec  2.40 GBytes  5149 Mbits/sec
```

```
[ 4] 12.00-16.00 sec  2.37 GBytes  5085 Mbits/sec
```

```
[ 4] 16.00-20.00 sec  2.42 GBytes  5196 Mbits/sec
```

```

[ ID] Interval          Transfer      Bandwidth
[ 4]  0.00-20.00 sec  11.8 GBytes  5081 Mbits/sec
[ 4]  0.00-20.00 sec  11.8 GBytes  5081 Mbits/sec

```

```
iperf Done.
```


- 帯域幅制限設定ありの場合 (ldom1 を 100 Mbps に制限)

送信先／送信元ともに、転送速度が約 **98.7 Mbps** となり、帯域幅制限の設定が有効になっていることが確認できます。

- 送信先 (ldom1)

```
ldom1# iperf3 -s -p 6000
-----
Server listening on 6000
-----
Accepted connection from 192.168.10.101, port 38714
[ 5] local 192.168.10.201 port 6000 connected to 192.168.10.101 port 39766
[ ID] Interval           Transfer     Bandwidth
~ (省略) ~
-----
[ ID] Interval           Transfer     Bandwidth
[ 5]  0.00-20.01  sec    0.00 Bytes    0.00 bits/sec           sender
[ 5]  0.00-20.01  sec   235 MBytes   98.6 Mbits/sec        receiver
```

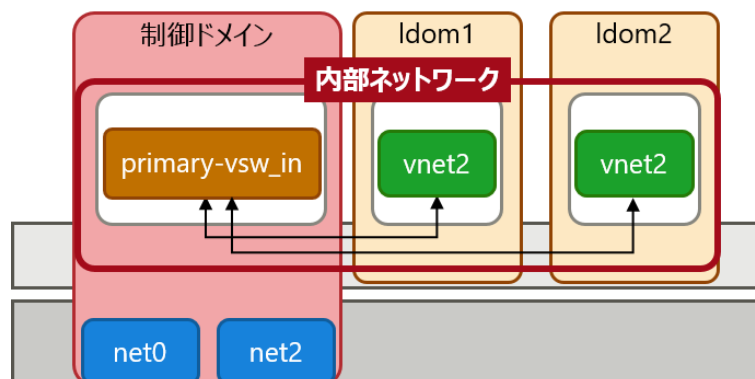
- 送信元 (primary)

```
primary# iperf3 -c 192.168.10.201 -t 20 -fm -i 4 -p 6000 -l 1024K -w 1024K -P1 -M
1472
Connecting to host 192.168.10.201, port 6000
[ 4] local 192.168.10.101 port 39766 connected to 192.168.10.201 port 6000
[ ID] Interval           Transfer     Bandwidth
[ 4]  0.00-4.00    sec    54.0 MBytes   113 Mbits/sec
[ 4]  4.00-8.00    sec    46.0 MBytes   96.5 Mbits/sec
[ 4]  8.00-12.00   sec    46.0 MBytes   96.5 Mbits/sec
[ 4] 12.00-16.00   sec    48.0 MBytes   101 Mbits/sec
[ 4] 16.00-20.00   sec    42.0 MBytes   88.1 Mbits/sec
-----
[ ID] Interval           Transfer     Bandwidth
[ 4]  0.00-20.00   sec   236 MBytes   99.0 Mbits/sec           Sender
[ 4]  0.00-20.00   sec   235 MBytes   98.7 Mbits/sec        receiver

iperf Done.
```

5.3. ゲストドメイン間の内部ネットワークの構築

物理 NIC を割り当てずに仮想スイッチを作成することで、ゲストドメイン間の内部ネットワークを構築できます。



- 1) 仮想スイッチサービスを設定します。

```
primary# ldm add-vswitch primary-vsw_in primary
```

▶ 物理 NIC[net-dev=<device>]を指定しません。

- 2) ldom1 に仮想ネットワークデバイスを割り当てます。

```
primary# ldm add-vnet vnet2 primary-vsw_in ldom1
```

- 3) ldom2 に仮想ネットワークデバイスを割り当てます。

```
primary# ldm add-vnet vnet2 primary-vsw_in ldom2
```

4) 作成した仮想ネットワークサービスと仮想ネットワークデバイスを確認します。

```
primary# ldm list-domain -o network
NAME
primary
~ (省略) ~
VSW
  NAME          MACADDRESS          NET-DEV  DVID|PVID|VIDs
  ----          -
  ~ (省略) ~
  primary-vsw_in 00:14:4f:fa:fe:03  --      1|1|--
            DEVICE      :switch@2      ID      :2
            LINKPROP    :--          MTU     :1500
            INTER-VNET-LINK :on/auto    MODE    :--
            VSW-RELAY-MODE :local
-----
NAME
ldom1
~ (省略) ~
  NAME          SERVICE          MACADDRESS          PVID|PVLAN|VIDs
  ----          -
  vnet2          primary-vsw_in@primary 00:14:4f:f8:61:28  1|--|--
~ (省略) ~
-----
NAME
ldom2
~ (省略) ~
  NAME          SERVICE          MACADDRESS          PVID|PVLAN|VIDs
  ----          -
  vnet2          primary-vsw_in@primary 00:14:4f:f8:83:2e  1|--|--
~ (省略) ~
```

5) ldom1 上で、仮想ネットワークデバイスが追加されたことを確認します。

vnet2(net2)が追加されていることを確認してください。

```
ldom1# dladm show-phys -L
LINK      DEVICE      LOC
net0      vnet0      --
net1      vnet1      --
net2      vnet2      --
```

▶ このデバイスを内部ネットワークとして利用できます。

6) ldom1 の net2 に IP アドレスを設定します。

```
ldom1# ipadm create-ip net2
ldom1# ipadm create-addr -T static -a 192.168.1.11/24 net2/v4
```

7) ldom1 の IP アドレスを確認します。

```
ldom1# ipadm show-addr
ADDROBJ      TYPE      STATE      ADDR
lo0/v4        static    ok          127.0.0.1/8
net0/v4        static    ok          192.168.10.201/24
net1/v4        static    ok          192.168.11.201/24
net2/v4        static    ok          192.168.1.11/24
lo0/v6         static    ok          ::1/128
net0/v6        addrconf  ok          fe80::214:4fff:fe8:2388/10
```

8) ldom2 上で、仮想ネットワークデバイスが追加されたことを確認します。

```
ldom2# dladm show-phys -L
LINK          DEVICE      LOC
net0           vnet0       --
net1           vnet1       --
net2           vnet2       --
```

9) ldom2 の net2 に IP アドレスを設定します。

```
ldom2# ipadm create-ip net2
ldom2# ipadm create-addr -T static -a 192.168.1.12/24 net2/v4
```

10) ldom2 の IP アドレスを確認します。

```
ldom2# ipadm show-addr
ADDROBJ      TYPE      STATE      ADDR
lo0/v4        static    ok          127.0.0.1/8
net0/v4        static    ok          192.168.10.202/24
net1/v4        static    ok          192.168.11.202/24
net2/v4        static    ok          192.168.1.12/24
lo0/v6         static    ok          ::1/128
net0/v6        addrconf  ok          fe80::214:4fff:fe8:f23e /10
```

11) ldom2 から ldom1 へ ping を実行し、内部ネットワークが有効であることを確認します。

```
ldom2# ping 192.168.1.11
192.168.1.11 is alive
```

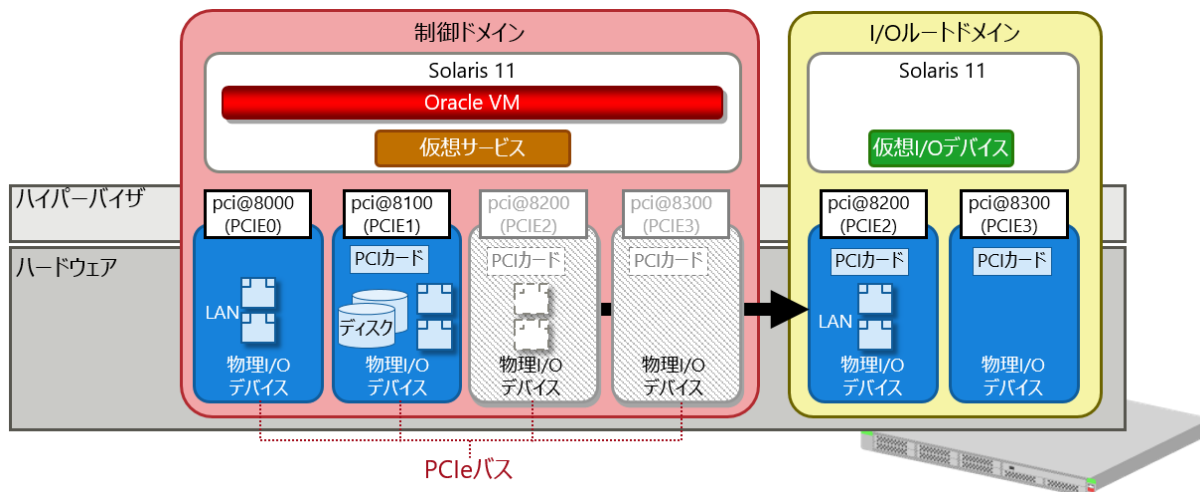
Point

Oracle VM の設定・構成を変更した場合は、以下のコマンドを実行して必ず構成情報を更新してください。

```
primary# ldm remove-spconfig config_initial
primary# ldm add-spconfig config_initial
```

6. I/O ルートドメインの構築

ドメインに物理 I/O デバイスを割り当てて構築します。物理 I/O デバイスには、PCIe バスを指定します。



本書では、ldom1 に PCIe バス「pci@8200」と「pci@8300」を割り当てます。

6.1. 物理 I/O デバイスの環境確認

1) ドメインの構成を確認します。

```
primary# ldm list-domain
NAME          STATE    FLAGS    CONS    VCPU    MEMORY    UTIL    NORM    UPTIME
primary       active  -n-cv-   UART    16      16G       0.1%    0.1%    22h 57m
ldom1         active  -n----- 5000     8       16G       0.1%    0.1%    22h 50m
ldom2         active  -n----- 5001     8       16G       0.1%    0.1%    7h 39m
```

2) 制御ドメイン・ゲストドメインに割り当てられている物理 I/O デバイス(LAN ポート)を確認します。

● 制御ドメインの確認

```
primary# dladm show-phys -L
LINK      DEVICE      LOC
net0      ixgbe0      /SYS/MBU/CMPO/PEX0
net1      ixgbe1      /SYS/MBU/CMPO/PEX0
net2      ixgbe2      /SYS/MBU/CMPO/PEX2
net3      ixgbe3      /SYS/MBU/CMPO/PEX2
~ (省略) ~
```

● ゲストドメインの確認(別のターミナル画面で確認)

```
ldom1# dladm show-phys -L
LINK      DEVICE      LOC
net0      vnet0      ---
net1      vnet1      ---
net2      vnet2      ---
```

《参考》 出力結果の見方

DEVICE の項目で、各ドメインのネットワークデバイスを確認します。

ixgbeX: 物理 LAN ポート(物理 I/O デバイス)

※ LAN ポートによってドライバ名は異なります。

vswX : 仮想スイッチ

vnetX : 仮想ネットワークデバイス

Point

ここでは LAN ポートのみを確認していますが、各ドメインの物理 I/O デバイスをすべて確認する場合は、prtdiag コマンドなどをご使用ください。

6.2. PCIe バスの割り当て

1) 物理 I/O デバイスの構成を確認します。

現在は、すべての物理 I/O デバイス(PCIe バス)が制御ドメインに割り当てられています。

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

NAME	TYPE	BUS	DOMAIN	STATUS
/SYS/MBU/CMP0/TDM0 [pci@8000]	BUS	PCIE0	primary	IOV
/SYS/MBU/CMP0/TDM1 [pci@8100]	BUS	PCIE1	primary	IOV
/SYS/MBU/CMP0/TDM2 [pci@8200]	BUS	PCIE2	primary	IOV
/SYS/MBU/CMP0/TDM3 [pci@8300]	BUS	PCIE3	primary	IOV
～ (省略) ～				

2) 制御ドメインから、PCIe バスの割り当てを解除します。

```
primary# ldm remove-io pci@8200 primary
primary# ldm remove-io pci@8300 primary
```

3) ゲストドメインに PCIe バスの割り当てを追加します。

```
primary# ldm add-io pci@8200 ldom1
primary# ldm add-io pci@8300 ldom1
```

4) 物理 I/O デバイスの構成を確認します。

PCIe バス「pci@8200」と「pci@8300」が、ドメインに割り当てられていることを確認してください。

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

NAME	TYPE	BUS	DOMAIN	STATUS
/SYS/MBU/CMP0/TDM0 [pci@8000]	BUS	PCIE0	primary	IOV
/SYS/MBU/CMP0/TDM1 [pci@8100]	BUS	PCIE1	primary	IOV
/SYS/MBU/CMP0/TDM2 [pci@8200]	BUS	PCIE2	ldom1	IOV
/SYS/MBU/CMP0/TDM3 [pci@8300]	BUS	PCIE3	ldom1	IOV

～（省略）～

5) 制御ドメインに割り当てられているネットワークインターフェースを確認します。

物理 LAN ポート ixgbe2, ixgbe3 が削除されていることを確認してください。

```
primary# dladm show-phys -L
```

LINK	DEVICE	LOC
net0	ixgbe0	/SYS/MBU/CMP0/PEX0
net1	ixgbe1	/SYS/MBU/CMP0/PEX0

～（省略）～

6) I/O ルートドメインに割り当てられているネットワークインターフェースを確認します。

i) I/O デバイスの構成を更新します。

```
ldom1# devfsadm
```

ii) ネットワークインターフェースを確認します。

ixgbe0 と ixgbe1 が追加されていることを確認してください。

```
ldom1# dladm show-phys -L
```

LINK	DEVICE	LOC
net5	ixgbe0	/SYS/MBU/CMP0/PEX2
net6	ixgbe1	/SYS/MBU/CMP0/PEX2
net0	vnet0	---
net1	vnet1	---
net2	vnet2	---

- iii) 追加されたネットワークインターフェースのデバイスパスを確認します。

I/O ルートドメインに、追加した PCIe バス「pci@8200」の物理 NIC が割り当てられていることを確認してください。

```
ldom1# ls -l /dev | grep ixgbe
lrwxrwxrwx 1 root root 31 Oct 31 13:11 ixgbe -
> ../devices/pseudo/clone0:ixgbe
lrwxrwxrwx 1 root root 54 Oct 31 13:11 ixgbe0 -
> ../devices/pci@8200/pci@4/pci@0/pci@0/network@0:ixgbe0
lrwxrwxrwx 1 root root 56 Oct 31 13:11 ixgbe1 -
> ../devices/pci@8200/pci@4/pci@0/pci@0/network@0,1:ixgbe1
```

6.3. PCIe バスの割り当ての解除

- 1) 物理 I/O デバイスの構成を元に戻します。

- i) PCIe バスを I/O ルートドメインから削除します。

```
primary# ldm remove-io pci@8200 ldom1
primary# ldm remove-io pci@8300 ldom1
```

- ii) PCIe バスを制御ドメインに追加します。

```
primary# ldm add-io pci@8200 primary
primary# ldm add-io pci@8300 primary
```

- iii) 物理 I/O デバイスの構成を確認します。

PCIe バス「pci@8200」と「pci@8300」が、制御ドメインに割り当てられていることを確認してください。

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

NAME	TYPE	BUS	DOMAIN	STATUS
/SYS/MBU/CMP0/TDM0 [pci@8000]	BUS	PCIE0	primary	IOV
/SYS/MBU/CMP0/TDM1 [pci@8100]	BUS	PCIE1	primary	IOV
/SYS/MBU/CMP0/TDM2 [pci@8200]	BUS	PCIE2	primary	IOV
/SYS/MBU/CMP0/TDM3 [pci@8300]	BUS	PCIE3	primary	IOV

～（省略）～

Point

Oracle VM の設定・構成を変更した場合は、以下のコマンドを実行して必ず構成情報を更新してください。

```
primary# ldm remove-spconfig config_initial
primary# ldm add-spconfig config_initial
```


7. ドメインの操作

7.1. ドメインの操作

7.1.1. ドメインの状態確認

1) 全ドメインの状態を確認します。

primary# ldm list-domain								
NAME	STATE	FLAGS	CONS	VCPU	MEMORY	UTIL	NORM	UPTIME
primary	active	-n-cv-	UART	16	16G	0.0%	0.0%	1d 21h 57m
ldom1	active	-n----	5000	8	16G	0.1%	0.1%	1d 21h 49m
ldom2	active	-n----	5001	8	16G	0.1%	0.1%	1d 6h 38m

①

②

③

④

⑤

⑥

《参考》 出力結果の見方

- ①NAME :ドメインの名称(制御ドメインは、「primary」で固定)
- ②STATE/FLAGS :ドメインの状態(起動、停止など)と種別
- ③CONS :ドメインの仮想コンソール接続のポート番号
- ④VCPU/MEMORY :ドメインの CPU・メモリリソース量
(CPU:スレッド単位、メモリ:ギガバイト単位(G)またはメガバイト単位(M))
- ⑤UTIL/NORM :ドメインの CPU 利用率
- ⑥UPTIME :ドメインの起動時間(起動中のドメインのみ表示)

STATE

- active : 起動中 (OS または OBP が起動)
- bound : 停止中 (リソース割り当ては完了)
- inactive : 停止中 (リソース未割り当て)

FLAGS

- : 可変部分
- c : 制御ドメイン
- d : 遅延再構成状態
- n : 通常(OS 起動)状態
- s : 起動中(停止中)
- t : 切り替え(OBP)状態
- v : サービスドメイン

7.1.2. ドメインの停止

1) ドメインを停止させます。

STATE が「bound」(停止中、リソース割り当て済み)に変わることを確認してください。

【書式】ldm stop-domain [[-f | -q] | [[-h | -r | -t sec] [-m <メッセージ>]]] (-a | <ドメイン名>...)

[SF] ldm stop-domain → ldm stop

```
primary# ldm stop-domain ldom1
primary# ldm list-domain
```

NAME	STATE	FLAGS	CONS	VCPU	MEMORY	UTIL	NORM	UPTIME
primary	active	-n-cv-	UART	16	16G	0.1%	0.1%	1d 22h 3m
ldom1	bound	-----	5000	8	16G			
ldom2	active	-n----	5001	8	16G	0.1%	0.1%	1d 6h 45m

2) ドメインをアンバインドします。

STATE が「inactive」(停止中、リソース未割り当て)に変わることを確認してください。

【書式】ldm unbind-domain <ドメイン名>

[SF] ldm unbind-domain → ldm unbind

```
primary# ldm unbind-domain ldom1
primary# ldm list-domain
```

NAME	STATE	FLAGS	CONS	VCPU	MEMORY	UTIL	NORM	UPTIME
primary	active	-n-cv-	UART	16	16G	0.8%	0.8%	1d 22h 5m
ldom2	active	-n----	5001	8	16G	0.1%	0.1%	1d 6h 46m
ldom1	inactive	-----		8	16G			

▶ 対象のドメイン上で shutdown コマンドを実行しても、ドメインを停止できます。

Point

ドメインの停止自体は、ldm stop-domain コマンドのみで完了しています。

さらに ldm unbind-domain コマンドを実行することで、そのドメインに割り当てていた CPU やメモリなどを、ほかのドメインに割り当てられるようになります。

7.1.3. ドメインの起動

1) ドメインをバインドします。

STATE が「bound」（停止中、リソース割り当て済み）に変わることを確認してください。

【書式】ldm bind-domain [-f] [-q] (-i <ファイル名> | <ドメイン名>)

SF ldm bind-domain → ldm bind

```
primary# ldm bind-domain ldom1
primary# ldm list-domain
```

NAME	STATE	FLAGS	CONS	VCPU	MEMORY	UTIL	NORM	UPTIME
primary	active	-n-cv-	UART	16	16G	1.7%	1.7%	1d 22h 14m
ldom1	bound	-----	5000	8	16G			
ldom2	active	-n----	5001	8	16G	0.4%	0.4%	1d 6h 55m

2) ドメインを起動させます。

STATE が「active」（起動中）に変わることを確認してください。

```
primary# ldm start-domain ldom1
primary# ldm list-domain
```

NAME	STATE	FLAGS	CONS	VCPU	MEMORY	UTIL	NORM	UPTIME
primary	active	-n-cv-	UART	16	16G	0.2%	0.2%	1d 22h 18m
ldom1	active	-t----	5000	8	16G	6.1%	5.6%	4s
ldom2	active	-n----	5001	8	16G	0.1%	0.1%	1d 6h 59m

7.1.4. ドメインの仮想コンソール接続

ゲストドメインのコンソールに接続するには、以下の 2 つの方法があります。

- telnet コマンド
- ldmconsole コマンド

※ ldmconsole コマンドは、Oracle VM Server for SPARC 3.6 / Oracle Solaris 11.4 から使用可能です。

1) 接続するドメインの名称と仮想コンソールポート番号を確認します。

ldom1 の仮想コンソールポートは「5000」です。

```
primary# ldm list-domain
```

NAME	STATE	FLAGS	CONS	VCPU	MEMORY	UTIL	NORM	UPTIME
primary	active	-n-cv-	UART	16	16G	0.1%	0.1%	1d 22h 18m
ldom1	active	-t----	5000	8	16G	12%	12%	1m
ldom2	active	-n----	5001	8	16G	0.0%	0.0%	1d 7h

2) 制御ドメインからドメインのコンソールにログインします。

● telnet コマンドの場合

【書式】telnet localhost <ドメインのコンソールポート番号>

```
primary# telnet localhost 5000
```

● ldmconsole コマンドの場合

【書式】ldmconsole <ドメインの名称>

```
primary# ldmconsole ldom1
```

3) OK プロンプトからドメインの OS を起動します。

- i) 以下のメッセージが出力されたあと、[Enter]キーを押します。

```
Connecting to console "ldom1" in group "ldom1" ....
Press ~? for control options ..
```

OK プロンプトが表示されます。

- ii) OS を起動します。

```
{0} ok boot
```

4) ゲストドメインのコンソールにログインします。

OS インストール時に設定したユーザとユーザパスワードで、ゲストドメインにログインします。

```
ldom1 console login: user-id
Password: *****
```

《参考》 使用可能なコンソール、およびコンソールグループの表示

ldmconsole (-l)で、使用可能なコンソールとコンソールグループを一覧表示します。

※ 制御ドメイン(primary)、および状態が inactive(リソース未割り当て)のドメインは表示されません。

```
primary# ldmconsole
GROUP          PORT          DOMAINS
ldom1          5000          ldom1
```

Point

ドメインに設定した auto-boot パラメーターの値によって、ドメインを起動・接続した際の状態が異なります。

auto-boot=true : OS が起動し、接続・OS 起動完了後に OS のログインプロンプトが表示

auto-boot=false : OBP が起動し、接続後に OK プロンプトが表示

7.1.5. ドメインの仮想コンソール接続の切断

- 1) 仮想コンソールを切断します。

```
ldom1# ~. ←チルダ+ドット
```

Point

「~.」は画面に表示されません。
表示される場合は、[Enter]キーを押してから「~.」を実行します。

7.1.6. 制御ドメインの停止

- 1) 制御ドメインで shutdown コマンドを使用します。

```
primary# shutdown -y -g0 -i0
```

▶ -i0 を指定して制御ドメインを停止します。停止後、OBP に移行します。

Point

制御ドメインを再起動・停止した場合、ゲストドメインや I/O ドメインなど、制御ドメインに依存しているドメインの I/O が停止しますのでご注意ください。

7.1.7. 制御ドメインの再起動

- 1) 制御ドメインで shutdown コマンドを使用します。

-i6 を指定して再起動します。

【書式】shutdown -y -g <シャットダウンまでの時間(秒)> -i <コマンド実行後のランレベル>

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

7.2. サーバ電源の操作

7.2.1. サーバの電源 OFF (制御ドメインで実行する場合)

- 1) 制御ドメイン以外のすべてのドメインを停止します。

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

- ▶ -a オプションで、制御ドメイン以外のすべてのドメインを停止させることができます。

- 2) 制御ドメイン以外のすべてのドメインをアンバインドします。

```
primary# ldm unbind-domain ldom1
primary# ldm unbind-domain ldom2
```

- 3) 制御ドメインを停止します。

-i5 を指定して停止します。

```
primary# shutdown -y -g0 -i5
```

- ▶ 「syncing file systems... done」が表示されたら停止完了です。#.(シャープ+ドット)を入力し、XSCF に遷移してください。
- ▶ 制御ドメイン以外のすべてのドメインを停止・アンバインドしていない場合は、再起動します。

7.2.2. サーバの電源 OFF (XSCF で実行する場合)

- 1) XSCF 上で、poweroff コマンドを実行します。

【書式】poweroff [[-q] [-y|n]] [-f] [-M] -p <物理パーティション ID>

```
XSCF> poweroff -p 0
```

- 2) 電源が停止していることを確認します。

Status が「Powered Off」であることを確認してください。

【書式】showpci [-v] -p <物理パーティション ID> [[-l <LSB 番号>]...]

```
XSCF> showpci -p 0
PPAR-ID  LSB  PSB  Status
00              00  00-0  Powered Off
```

停止しない場合は、-f オプションを指定して実行します。

```
XSCF> poweroff -f -p 0
PPAR-IDs to power off :00
The -f option will cause domains to be immediately reset.
Continue? [y|n] :y
```

Point

- SPARC M12/M10 の場合、シャットダウングループに設定した順序でドメインが停止されます。そのあと、電源 OFF が実行されます。
詳細は、以下を参照してください。
- 『Oracle VM Server for SPARC を使ってみよう（構築・運用ガイド）』—「Oracle VM Server for SPARC のその他の便利な機能」—「1. SPARC M12/M10 特有の機能 2/4」
<https://www.fujitsu.com/jp/sparc/technical/document/solaris/#ovm>
- サーバの電源 OFF は、Oracle VM の構成情報を保存 (ldm add-spconfig) したあとに実行してください。構成情報が保存されていない場合、一部の設定が削除されてしまうことがあります。

7.2.3. サーバの電源 ON**1) XSCF 上で、poweron コマンドを実行します。**

【書式】poweron [[-q] [-y|n]] [-M] -p <物理パーティション ID>

```
XSCF> poweron -p 0
PPAR-IDs to power on :00
Continue? [y|n] :y
00 : Powering on
```

2) OS のコンソールに接続します。

```
XSCF> console -p 0

Console contents may be logged.
Connect to PPAR-ID 0?[y|n] :y
```

《参考》ドメイン起動時に自動で OS もブートさせるための設定

(1) ドメインの状態を確認します。

```
primary# ldm list-domain
NAME          STATE      FLAGS    CONS    VCPU    MEMORY    UTIL    NORM    UPTIME
primary       active    -n-cv-   UART    16      16G       1.0%    1.0%    10m
ldom1         inactive -----      8       16G
ldom2         inactive -----      8       16G
```

(2) ドメインの起動パラメーターを確認します。

【書式】ldm list-variable <パラメーター> <ドメイン名>

☐ SF ldm list-variable → ldm list-var

```
primary# ldm list-variable auto-boot%? ldom1
auto-boot%=false
```

(3) ドメインの起動パラメーターを変更します。

```
primary# ldm set-variable auto-boot%?=true ldom1
Variables or keys have been set and those settings will be lost over a powercycle.
To persist variables or keys over a powercycle, a configuration must be saved to
the SP after updates.
```

(4) ドメインの起動パラメーターを確認します。

「true」(自動起動 ON)に設定されていることを確認してください。

```
primary# ldm list-variable auto-boot%? ldom1
auto-boot%=true
Variables or keys have been set and those settings will be lost over a powercycle.
To persist variables or keys over a powercycle, a configuration must be saved to
the SP after updates.
```

(5) ドメインを起動します。

```
primary# ldm bind-domain ldom1
primary# ldm bind-domain ldom2
primary# ldm start-domain -a
```

(6) ドメインの状態を確認します。

```
primary# ldm list-domain
NAME          STATE      FLAGS    CONS    VCPU    MEMORY    UTIL    NORM    UPTIME
primary       active    -n-cv-   UART    16      16G       0.3%    0.3%    15m
ldom1         active    -n----- 5000    8       16G       27%     27%     21s
ldom2         active    -t----- 5001    8       16G       12%     12%     21s
```

- ▶ OBP の状態ではなく、OS が起動していることが確認できます。
- ▶ FLAGS が「t」から「n」になるには、数秒かかります。

《参考》コンソール端末からゲストドメインに直接コンソール接続するための設定

デフォルトでは、ゲストドメインへの仮想コンソール接続は、制御ドメイン上で実施します。

以下の設定を行うことで、直接コンソール端末などからゲストドメインのコンソールに接続できるようになります。

- (1) 制御ドメインの/etc/hosts ファイルに、制御ドメインの IP アドレスとホスト名が記述されていることを確認します。

```
primary# cat /etc/hosts
#
# Copyright 2009 Sun Microsystems, Inc. All rights reserved.
# Use is subject to license terms.
#
# Internet host table
#
::1          localhost
127.0.0.1    localhost loghost
192.168.10.200 M12-1
```

▶ 記述されていない場合、または IP アドレスやホスト名が異なる場合は編集してください。

- (2) vntsd の設定を変更します。

【書式】setprop vntsd/listen_addr=<制御ドメインの IP アドレス>

```
primary# svccfg
svc:> select /ldoms/vntsd
svc:/ldoms/vntsd> setprop vntsd/listen_addr=192.168.10.200
svc:/ldoms/vntsd> end
```

- (3) vntsd を再起動します。

```
primary# svcadm refresh vntsd
primary# svcadm restart vntsd
```

- (4) ゲストドメインのコンソールポート番号を確認します。

```
primary# ldm list-domain
```

NAME	STATE	FLAGS	CONS	VCPU	MEMORY	UTIL	NORM	UPTIME
primary	active	-n-cv-	UART	16	16G	0.1%	0.1%	20m
ldom1	active	-n----	5000	8	16G	0.1%	0.1%	6m

～（省略）～

(5) コンソール端末から、telnet でゲストドメインのコンソールポート番号を指定して接続します。

ログイン画面が表示されることを確認してください。

【書式】telnet <制御ドメインの IP アドレス> <ゲストドメインのコンソールポート番号>

```
$ telnet 192.168.10.200 5000
Trying 192.168.10.200...
Connected to 192.168.10.200.
Escape character is '^]'.
Connecting to console "ldom1" in group "ldom1" ....
Press ~? for control options ..
ldom1 console login:
```

▶ ログアウトする場合は、「~.」(チルダ+ドット)を入力します。

※ vntsd の設定を変更したことにより、デフォルトの接続方法 (localhost 指定) は使用できなくなります。

```
primary# telnet localhost 5000
Trying ::1...
telnet: connect to address ::1: Connection refused
Trying 127.0.0.1...
telnet: Unable to connect to remote host: Connection refused
```

※ vntsd の設定を元に戻す場合は、以下の手順を実行してください。

(1) vntsd の設定を変更します。

listen_addr に localhost を指定します。

```
primary# svccfg
svc:> select /ldoms/vntsd
svc:/ldoms/vntsd> setprop vntsd/listen_addr=localhost
svc:/ldoms/vntsd> end
```

(2) vntsd を再起動します。

```
primary# svcadm refresh vntsd
primary# svcadm restart vntsd
```

8. リソースの操作

Point

ここでは、制御ドメインとゲストドメインそれぞれのコンソール画面を表示し、リソースの状態を確認しながらリソースの操作を実施します。

8.1. 使用リソースの確認

- 1) ドメインで利用できる全リソースを表示します。

```
primary# ldm list-devices -a
```

- 2) 未使用のリソースを表示します。

```
primary# ldm list-devices
```

8.2. ドメインへの CPU の追加／削減

- 1) CPU を確認します。

```
primary# ldm list-domain
```

NAME	STATE	FLAGS	CONS	VCPU	MEMORY	UTIL	NORM	UPTIME
primary	active	-n-cv-	UART	16	16G	0.1%	0.1%	47m
ldom1	active	-n----	5000	8	16G	0.0%	0.0%	33m

～（省略）～

- 2) ドメインに CPU を追加します。

ドメインの CPU を 1 コア追加します。

```
primary# ldm add-core 1 ldom1
```

- 3) CPU が追加されたことを確認します。

1 コア (8 VCPU) 追加されていることを確認してください。

- ・SPARC64 XII (SPARC M12) の場合、1 コアあたり 8 VCPU (スレッド)
- ・SPARC64 X+/X (SPARC M10) の場合、1 コアあたり 2 VCPU (スレッド)

```
primary# ldm list-domain
```

NAME	STATE	FLAGS	CONS	VCPU	MEMORY	UTIL	NORM	UPTIME
primary	active	-n-cv-	UART	16	16G	0.0%	0.0%	1h 18m
ldom1	active	-n----	5000	16	16G	0.0%	0.0%	1h 4m

～（省略）～

▶ psrinfo -vp コマンドを使用し、ドメイン上で確認することもできます。

4) ドメインから CPU を削減します。

ドメインの CPU を 1 コア削減します。

```
primary# ldm remove-core 1 ldom1
```

5) CPU が削減されたことを確認します。

```
primary# ldm list-domain
NAME          STATE    FLAGS    CONS    VCPU    MEMORY    UTIL    NORM    UPTIME
primary       active   -n-cv-   UART    16      16G       0.1%    0.1%    1h 22m
ldom1         active   -n----- 5000     8       16G       0.1%    0.1%    1h 8m
~ (省略) ~
```

《参考》 コアの最大数の設定 (max-cores プロパティ)

(1) ドメインを停止・アンバインドします。

```
primary# ldm stop-domain ldom1
primary# ldm unbind-domain ldom1
```

▶ コアの最大数を設定する際は、事前にドメインを停止し、アンバインドする必要があります。

(2) ドメインの制限の設定を確認します。

デフォルトで max-cores は、「unlimited」に設定されています。

```
primary# ldm list-domain -o resmgmt ldom1
NAME
ldom1

CONSTRAINT
  cpu=whole-core
  max-cores=unlimited
```

▶ このとき、ドメインに割り当てられる CPU コアの数に制限はありません。

(3) ドメインに割り当てられるコアの最大数 (max-cores) を設定します。

```
primary# ldm set-domain max-cores=2 ldom1
```

(4) 再度、制限の設定を確認します。

「max-cores」プロパティが「unlimited」から「2」に変更されていることを確認してください。

```
primary# ldm list-domain -o resmgmt ldom1
NAME
ldom1

CONSTRAINT
  cpu=whole-core
  max-cores=2
```

(5) ドメインをバインドし、起動します。

```
primary# ldm bind-domain ldom1
primary# ldm start-domain ldom1
```

(6) ドメインに CPU を 2 コア割り当てます。

```
primary# ldm set-core 2 ldom1
```

▶ 「max-cores」プロパティの範囲内であるため、CPU コアを割り当てられます。

(7) ドメインに CPU を 3 コア割り当てます(エラー)。

```
primary# ldm set-core 3 ldom1
Number of cores 3 must not exceed max-cores=2
```

▶ 「max-cores」よりも大きい数の CPU コアを指定したため、割り当てに失敗します(エラー出力)。

(8) ドメインの CPU リソースを確認します。

ドメインには 2 コア(16 スレッド)の CPU が割り当てられています。

```
primary# ldm list-domain
NAME          STATE    FLAGS   CONS   VCPU  MEMORY  UTIL  NORM  UPTIME
primary       active   -n-cv-  UART   16    16G     0.0%  0.0%  1h 33m
ldom1         active   -n----  5000   16    16G     0.0%  0.0%  4m
~ (省略) ~
```

※ コアの最大数の設定を解除する場合は、以下の手順を実行してください。

(1) ドメインを停止し、アンバインドします。

```
primary# ldm stop-domain ldom1
primary# ldm unbind-domain ldom1
```

(2) コアの最大数の設定を解除します。

```
primary# ldm set-domain max-cores=unlimited ldom1
```

(3) 制限の設定を確認します。

「unlimited」(制限なし)に変更されていることを確認してください。

```
primary# ldm list-domain -o resmgmt ldom1
NAME
ldom1

CONSTRAINT
  cpu=whole-core
  max-cores=unlimited
```

8.3. ドメインの CPU 割り当て方式の変更

ドメインへの CPU 割り当てには、以下の 3 種類の方式があります。

- コア数指定
- コア ID 指定
- スレッド数指定

CPU 割り当て方式によって、設定方法が異なります。

ここでは、「コア数指定」から「スレッド数指定」に変更する例を示します。

1) 現在の CPU リソースを確認します。

```
primary# ldm list-domain
NAME          STATE    FLAGS    CONS    VCPU    MEMORY  UTIL  NORM  UPTIME
primary       active   -n-cv-   UART    16      16G     0.1%  0.1%  1h 46m
ldom1         active   -n----- 5000    16      16G     0.0%  0.0%   8m
~ (省略) ~
```

2) 現在の割り当て方式を確認します。

「cpu=whole-core」が表示されているため、コア数指定です。

```
primary# ldm list-domain -o resmgmt ldom1
NAME
ldom1

CONSTRAINT
cpu=whole-core
max-cores=unlimited
```

▶ コア ID 指定の場合は、「physical-bindings=core」が表示されます。

3) 割り当て方式を変更します。

「コア数指定」から「スレッド数指定」に変更します。この場合、動的に変更できます。

「ldm set- core|vcpu」コマンドを使用します。

```
primary# ldm set-vcpu 12 ldom1
```

▶ SPARC M12 の場合、4 スレッド単位で割り当ててください。
SPARC M10 の場合は、1 スレッド単位で問題ありません。

4) CPU リソースを確認します。

```
primary# ldm list-domain
NAME          STATE    FLAGS    CONS    VCPU    MEMORY  UTIL  NORM  UPTIME
primary       active   -n-cv-   UART    16      16G     0.1%  0.1%  1h 50m
ldom1         active   -n----- 5000    12      16G     0.0%  0.0%  12m
~ (省略) ~
```

5) 割り当て方式を確認します。

「cpu=whole-core」が表示されていないため、スレッド数指定です。

```
primary# ldm list-domain -o resmgt ldom1
NAME
ldom1

CONSTRAINT
```

←何も表示されません。

6) 割り当て方式を元に戻します。

```
primary# ldm set-core 1 ldom1
```

7) 割り当て方式を確認します。

「cpu=whole-core」が表示されているため、コア数指定です。

```
primary# ldm list-domain -o resmgt ldom1
NAME
ldom1

CONSTRAINT
  cpu=whole-core
  max-cores=unlimited
```

8.4. ドメインへのメモリの割り当て

ドメインのメモリ容量の変更方法には、動的に変更する方法と静的に変更する方法の2種類があります。ここでは、メモリ容量を動的に変更する手順を示します。

1) メモリリソースを確認します。

```
primary# ldm list-domain
NAME          STATE    FLAGS    CONS    VCPU    MEMORY    UTIL    NORM    UPTIME
primary       active   -n-cv-   UART    16      16G       0.2%    0.2%    1h 53m
ldom1         active   -n----- 5000     8       16G       0.5%    0.5%    15m
~ (省略) ~
```

2) ドメインにメモリリソースを追加します。

```
primary# ldm add-memory 8g ldom1
```

3) メモリリソースを確認します。

```
primary# ldm list-domain
NAME          STATE    FLAGS    CONS    VCPU    MEMORY    UTIL    NORM    UPTIME
primary       active   -n-cv-   UART    16      16G       0.1%    0.1%    1h 55m
ldom1         active   -n----- 5000     8       24G       0.2%    0.2%    18m
~ (省略) ~
```

▶ prtconf -v |head コマンドを使用し、ドメイン上で確認することもできます。

4) ドメインからメモリリソースを削減します。

```
primary# ldm remove-memory 8g ldom1
```

5) メモリリソースを確認します。

```
primary# ldm list-domain
NAME          STATE    FLAGS  CONS  VCPU  MEMORY  UTIL  NORM  UPTIME
primary       active  -n-cv-  UART   16    16G     0.1%  0.1%  1h 57m
ldom1         active  -n----  5000    8    16G     0.2%  0.2%  20m
~ (省略) ~
```


8.5. ドメインへの仮想ネットワークの追加

- 1) 物理ネットワークデバイス(物理 NIC)のインターフェース名を確認します。

```
primary# dladm show-phys -L
LINK          DEVICE          LOC
net0          ixgbe0          /SYS/MBU/CMP0/PEX0
net1          ixgbe1          /SYS/MBU/CMP0/PEX0
net2          ixgbe2          /SYS/MBU/CMP0/PEX2
net3          ixgbe3          /SYS/MBU/CMP0/PEX2
```

- 2) 仮想ネットワークの状態を確認します。

```
primary# ldm list-domain -o network
NAME
primary
~ (省略) ~
VSW
NAME          MACADDRESS          NET-DEV  DVID|PVID|VIDs
-----
primary-vsw0  00:14:4f:f8:93:ac   net0     1|1|---
~ (省略) ~
primary-vsw_vlan  00:14:4f:f8:63:0f  net1     1|1|---
~ (省略) ~
primary-vsw_in  00:14:4f:fa:fe:03   --       1|1|---
~ (省略) ~
↑ 作成済みの仮想スイッチサービス
```

```
NAME
ldom1
~ (省略) ~
NETWORK
NAME          SERVICE          MACADDRESS          PVID|PVLAN|VIDs
-----
vnet0         primary-vsw0@primary  00:14:4f:fa:e9:94   1|---|---
~ (省略) ~
vnet1         primary-vsw_vlan@primary  00:14:4f:fb:b8:2c   1|---|---
~ (省略) ~
vnet2         primary-vsw_in@primary  00:14:4f:f8:61:28   1|---|---
~ (省略) ~
↑ 仮想 NIC として、vnet0, vnet1, vnet2 を ldom1 に割り当てています。
```

3) ドメイン上でネットワークインターフェースを確認します。

```
ldom1# dladm show-phys -L
LINK          DEVICE          LOC
net0          vnet0             --
net1          vnet1             --
net2          vnet2             --
```

4) 仮想スイッチサービスを作成します。

```
primary# ldm add-vswitch net-dev=net2 primary-vsw2 primary
```

5) ドメイン(ldom1)に、仮想ネットワークデバイス(仮想 NIC)を割り当てます。

```
primary# ldm add-vnet vnet3 primary-vsw2 ldom1
```

6) 仮想ネットワークの状態を確認します。

```
primary# ldm list-domain -o network
NAME
primary
~ (省略) ~
VSW
  NAME          MACADDRESS          NET-DEV  DVID|PVID|VIDs
  ----          -
  ~ (省略) ~
  primary-vsw2  00:14:4f:fb:03:40    net2     1|1|--
  ~ (省略) ~

NAME
ldom1
~ (省略) ~
NETWORK
  ~ (省略) ~
  NAME          SERVICE          MACADDRESS          PVID|PVLAN|VIDs
  ----          -
  vnet3         primary-vsw2@primary  00:14:4f:fb:6d:a4    1|--|--
  ~ (省略) ~
```

7) ドメイン上でネットワークインターフェースを確認します。

```
ldom1# dladm show-phys -L
LINK          DEVICE          LOC
net0          vnet0             --
net1          vnet1             --
net2          vnet2             --
net3          vnet3             --
```

8) インターフェースを有効にし、IP アドレスを設定します。

```
ldom1# ipadm create-ip net3
ldom1# ipadm create-addr -T static -a local=10.10.10.201/24 net3/v4
```

9) IP アドレスが設定されていることを確認します。

```
ldom1# ipadm show-addr
ADDROBJ      TYPE      STATE      ADDR
lo0/v4        static    ok          127.0.0.1/8
net0/v4        static    ok          192.168.10.201/24
net1/v4        static    ok          192.168.11.201/24
net2/v4        static    ok          192.168.1.11/24
net3/v4        static    ok          10.10.10.201/24
(～省略～)
```

8.6. ドメインの仮想ネットワークの削除

1) 仮想ネットワークの状態を確認します。

```
primary# ldm list-domain -o network
NAME
primary
~ (省略) ~
VSW
  NAME          MACADDRESS          NET-DEV  DVID|PVID|VIDs
  ----          -
  primary-vsw0  00:14:4f:f8:93:ac    net0     1|1|--
  ~ (省略) ~
  primary-vsw_vlan 00:14:4f:f8:63:0f    net1     1|1|--
  ~ (省略) ~
  primary-vsw_in 00:14:4f:fa:fe:03    --       1|1|--
  ~ (省略) ~
  primary-vsw2  00:14:4f:fb:03:40    net2     1|1|--
  ~ (省略) ~

-----
NAME
ldom1
~ (省略) ~
NETWORK
  NAME          SERVICE          MACADDRESS          PVID|PVLAN|VIDs
  ----          -
  vnet0          primary-vsw0@primary  00:14:4f:fa:e9:94    1|--|--
  ~ (省略) ~
  vnet1          primary-vsw_vlan@primary 00:14:4f:fb:b8:2c    1|--|--
  ~ (省略) ~
  vnet2          primary-vsw_in@primary  00:14:4f:f8:61:28    1|--|--
  ~ (省略) ~
  vnet3          primary-vsw2@primary  00:14:4f:fb:6d:a4    1|--|--
  ~ (省略) ~
```

2) 作成したネットワークインターフェースを無効にします。

```
ldom1# ipadm delete-ip net3
```

3) ネットワークインターフェースが無効であることを確認します。

```
ldom1# ipadm show-addr
ADDROBJ      TYPE      STATE      ADDR
lo0/v4       static    ok         127.0.0.1/8
net0/v4       static    ok         192.168.10.201/24
net1/v4       static    ok         192.168.11.201/24
net2/v4       static    ok         192.168.1.11/24
(~省略~)
```

4) 仮想 NIC の割り当てを解除します。

【書式】ldm remove-vnet [-f] <仮想ネットワークインターフェース名> <ドメイン名>

SF ldm remove-vnet → ldm rm-vnet

```
primary# ldm remove-vnet vnet3 ldom1
```

5) 仮想スイッチサービスを削除します。

【書式】ldm remove-vswitch [-f] <仮想スイッチサービス名>

SF ldm remove-vswitch → ldm rm-vsw

```
primary# ldm remove-vswitch primary-vsw2
```

6) 仮想ネットワークの状態を確認します。

```
primary# ldm list-domain -o network
```

NAME

primary

～（省略）～

VSW

NAME	MACADDRESS	NET-DEV	DVID PVID VIDs
primary-vsw0	00:14:4f:f8:93:ac	net0	1 1 --
～（省略）～			
primary-vsw_vlan	00:14:4f:f8:63:0f	net1	1 1 --
～（省略）～			
primary-vsw_in	00:14:4f:fa:fe:03	--	1 1 --
～（省略）～			

NAME

ldom1

～（省略）～

NETWORK

NAME	SERVICE	MACADDRESS	PVID PVLAN VIDs
vnet0	primary-vsw0@primary	00:14:4f:fa:e9:94	1 -- --
～（省略）～			
vnet1	primary-vsw_vlan@primary	00:14:4f:fb:b8:2c	1 -- --
～（省略）～			
vnet2	primary-vsw_in@primary	00:14:4f:f8:61:28	1 -- --
～（省略）～			

8.7. ドメインへの仮想ディスクの追加

1) 仮想ディスクの状態を確認します。

```
primary# ldm list-domain -o disk
NAME
primary

VDS
  NAME          VOLUME          OPTIONS          MPGROUP          DEVICE
  primary-vds0  vol0
/dev/dsk/c0t600000E00D000000000103460000000d0s2
      vol_iso      ro                /ISO/Sol11.4.iso
      vol1
/dev/dsk/c0t600000E00D0000000001034600010000d0s2
      vol_iso1     ro                /ISO/Sol11.4.iso

-----
NAME
ldom1

DISK
  NAME          VOLUME          TOUT ID    DEVICE  SERVER  MPGROUP
  vdisk0        vol0@primary-vds0  0        disk@0  primary
  vdisk_iso     vol_iso@primary-vds0  1        disk@1  primary

-----
~（省略）~
```

▶ 以降の操作で、ドメイン 1(ldom1)に仮想ディスクを割り当てます。

2) ドメイン上の仮想ディスクを確認します。

i) I/O デバイスの構成を更新します。

```
ldom1# devfsadm -C
```

ii) ドメインが認識しているディスク構成を確認します。

```
ldom1# ls /dev/dsk
c1d0s0 c1d0s2 c1d0s4 c1d0s6 c1d1s0 c1d1s2 c1d1s4 c1d1s6
c1d0s1 c1d0s3 c1d0s5 c1d0s7 c1d1s1 c1d1s3 c1d1s5 c1d1s7
```

iii) ドメインが認識しているディスクを確認します。

```
ldm1# format
Searching for disks...done

AVAILABLE DISK SELECTIONS:
  0. c1d0 <FUJITSU-ETERNUS_DXL-0000 cyl 6398 alt 2 hd 64 sec 256>
     /virtual-devices@100/channel-devices@200/disk@0
Specify disk (enter its number):
```

▶ [Ctrl]+[C]キーで format メニューを終了します。

3) 未使用の物理ディスクを確認します。

```
primary# format
Searching for disks...done

AVAILABLE DISK SELECTIONS:
  0. c0t5000039698132D40d0 <TOSHIBA-AL13SEB600-3703-558. 91GB>
     /scsi_vhci/disk@g5000039698132d40
     /dev/chassis/SYS/HDD0/disk
  1. c0t500003970830F9E1d0 <TOSHIBA-AL13SEB600AL14SE-3703-558. 91GB>
     /scsi_vhci/disk@g500003970830f9e1
     /dev/chassis/SYS/HDD1/disk
  2. c0t600000E00D000000000103460000000d0 <FUJITSU-ETERNUS_DXL-0000-50. 00GB>
     /scsi_vhci/disk@g600000e00d0000000001034600000000
  3. c0t600000E00D0000000001034600010000d0 <FUJITSU-ETERNUS_DXL-0000-50. 00GB>
     /scsi_vhci/disk@g600000e00d0000000001034600010000
  4. c0t600000E00D0000000001034600020000d0 <FUJITSU-ETERNUS_DXL-0000-50. 00GB>
     /scsi_vhci/disk@g600000e00d0000000001034600020000
~ (省略) ~
```

▶ 本環境では、以下のディスクをゲストドメインの仮想ディスクとして追加します。

4. c0t600000E00D0000000001034600020000d0

4) 物理ディスクを仮想ディスクデバイスとして登録します(排他設定)。

```
primary# ldm add-vdiskserverdevice options=excl
/dev/dsk/c0t600000E00D0000000001034600020000d0s2 vol_disk@primary-vds0
```

▶ s2 を指定します。

Point

excl オプションは「排他設定」のためのオプションです。

排他設定を行い、ゲストドメインに割り当てたディスクを制御ドメインから操作できないようにします。

5) 仮想ディスクデバイスを ldom1 に割り当てます。

```
primary# ldm add-vdisk vdisk1 vol_disk@primary-vds0 ldom1
```

6) 仮想ディスクの構成を確認します。

```
primary# ldm list-domain -o disk
NAME
primary

VDS
  NAME          VOLUME          OPTIONS          MPGROUP          DEVICE
  primary-vds0  vol0
/dev/dsk/c0t600000E00D000000000103460000000d0s2
                vol_iso          ro                /ISO/Sol11.4.iso
                vol1
/dev/dsk/c0t600000E00D0000000001034600010000d0s2
                vol_iso1         ro                /ISO/Sol11.4.iso
                vol_disk         excl
/dev/dsk/c0t600000E00D0000000001034600020000d0s2

-----
NAME
ldom1

DISK
  NAME          VOLUME          TOUT ID    DEVICE  SERVER    MPGROUP
  vdisk0         vol0@primary-vds0      0    disk@0  primary
  vdisk_iso      vol_iso@primary-vds0   1    disk@1  primary
  vdisk1         vol_disk@primary-vds0  2    disk@2  primary

-----
~（省略）~
```


7) 制御ドメイン上で物理ディスクの状態を確認します。

i) 制御ドメインが認識しているディスクを確認します。

```
primary# format
Searching for disks...done

AVAILABLE DISK SELECTIONS:
  0. c0t5000039698132D40d0 <TOSHIBA-AL13SEB600-3703-558. 91GB>
    /scsi_vhci/disk@g5000039698132d40
    /dev/chassis/SYS/HDD0/disk
  1. c0t500003970830F9E1d0 <TOSHIBA-AL13SEB600AL14SE-3703-558. 91GB>
    /scsi_vhci/disk@g500003970830f9e1
    /dev/chassis/SYS/HDD1/disk
  2. c0t600000E00D000000000103460000000d0 <FUJITSU-ETERNUS_DXL-0000-50. 00GB>
    /scsi_vhci/disk@g600000e00d0000000001034600000000
  3. c0t600000E00D0000000001034600010000d0 <FUJITSU-ETERNUS_DXL-0000-50. 00GB>
    /scsi_vhci/disk@g600000e00d0000000001034600010000
  4. c0t600000E00D0000000001034600020000d0 <FUJITSU-ETERNUS_DXL-0000-50. 00GB>
    /scsi_vhci/disk@g600000e00d0000000001034600020000
~ (省略) ~
```

▶ Oracle VM 3.5 までは、排他設定した物理ディスク「4. c0t600000E00D0000000001034600020000d0」が表示されません。排他設定をすることによって、物理ディスクが認識されなくなります。

ii) 仮想ディスクとして追加した物理ディスクの番号を入力します。

```
primary# format
searching for disks...done

AVAILABLE DISK SELECTIONS:
  0. c0t5000039698132D40d0 <TOSHIBA-AL13SEB600-3703-558. 91GB>
    /scsi_vhci/disk@g5000039698132d40
    /dev/chassis/SYS/HDD0/disk
  1. c0t500003970830F9E1d0 <TOSHIBA-AL13SEB600AL14SE-3703-558. 91GB>
    /scsi_vhci/disk@g500003970830f9e1
    /dev/chassis/SYS/HDD1/disk
  2. c0t600000E00D000000000103460000000d0 <FUJITSU-ETERNUS_DXL-0000-50. 00GB>
    /scsi_vhci/disk@g600000e00d0000000001034600000000
  3. c0t600000E00D0000000001034600010000d0 <FUJITSU-ETERNUS_DXL-0000-50. 00GB>
    /scsi_vhci/disk@g600000e00d0000000001034600010000
  4. c0t600000E00D0000000001034600020000d0 <FUJITSU-ETERNUS_DXL-0000-50. 00GB>
    /scsi_vhci/disk@g600000e00d0000000001034600020000
~ (省略) ~
Specify disk (enter its number): 4
```

- iii) 排他設定した物理ディスクにアクセスできないことを確認します。確認したあとに、[Ctrl]+[C]キーで終了します。

～（省略）～

Specify disk (enter its number): 4

Error: can't open selected disk '/dev/rdisk/c0t600000E00D00000000001034600020000d0s2'.

8) ドメイン上でディスクの状態を確認します。

- i) ドメインが認識しているディスク構成を確認します。

追加した仮想ディスクデバイスが認識されていることを確認してください。

```
ldom1# ls /dev/dsk
c1d0s0 c1d0s3 c1d0s6 c1d1s1 c1d1s4 c1d1s7 c1d2s2 c1d2s5
c1d0s1 c1d0s4 c1d0s7 c1d1s2 c1d1s5 c1d2s0 c1d2s3 c1d2s6
c1d0s2 c1d0s5 c1d1s0 c1d1s3 c1d1s6 c1d2s1 c1d2s4 c1d2s7
```

- ii) ドメインが認識しているディスクを確認します。

仮想ディスクを動的に追加したことが確認できます。

```
ldom1# format
Searching for disks...done

AVAILABLE DISK SELECTIONS:
  0. c1d0 <FUJITSU-ETERNUS_DXL-0000 cyl 6398 alt 2 hd 64 sec 256>
    /virtual-devices@100/channel-devices@200/disk@0
  1. c1d2 <FUJITSU-ETERNUS_DXL-0000 cyl 6398 alt 2 hd 64 sec 256>
    /virtual-devices@100/channel-devices@200/disk@2
Specify disk (enter its number):
```

- ▶ [Ctrl]+[C]キーで format メニューを終了します。

8.8. ドメインの仮想ディスクの削除

1) 仮想ディスクの構成を確認します。

削除するドメインの仮想ディスクを確認してください。

```
primary# ldm list-domain -o disk
NAME
primary

VDS
  NAME          VOLUME          OPTIONS          MPGROUP          DEVICE
  primary-vds0  vol0
/dev/dsk/c0t600000E00D000000000103460000000d0s2
                vol_iso          ro               /ISO/Sol11.4.iso
                vol1
/dev/dsk/c0t600000E00D0000000001034600010000d0s2
                vol_iso1         ro               /ISO/Sol11.4.iso
                vol_disk         excl
/dev/dsk/c0t600000E00D0000000001034600020000d0s2

-----
NAME
ldom1

DISK
  NAME          VOLUME          TOUT ID  DEVICE  SERVER  MPGROUP
  vdisk0        vol0@primary-vds0    0      disk@0  primary
  vdisk_iso     vol_iso@primary-vds0 1      disk@1  primary
  vdisk1        vol_disk@primary-vds0 2      disk@2  primary

-----
~（省略）~
```

2) 仮想ディスクの割り当てを解除します。

```
primary# ldm remove-vdisk vdisk1 ldom1
```

3) 仮想ディスクサービスから、仮想ディスクのデバイスを解除します。

```
primary# ldm remove-vdiskserverdevice vol_disk@primary-vds0
```

4) 仮想ディスクの構成を確認します。

仮想ディスクデバイス「vdisk1」が削除されていることを確認してください。

```
primary# ldm list-domain -o disk
```

5) 制御ドメイン上で物理ディスクの状態を確認します。

i) 制御ドメインが認識しているディスクを確認します。

```
primary# format
Searching for disks... done

AVAILABLE DISK SELECTIONS:
  0. c0t5000039698132D40d0 <TOSHIBA-AL13SEB600-3703-558. 91GB>
    /scsi_vhci/disk@g5000039698132d40
    /dev/chassis/SYS/HDD0/disk
  1. c0t500003970830F9E1d0 <TOSHIBA-AL13SEB600AL14SE-3703-558. 91GB>
    /scsi_vhci/disk@g500003970830f9e1
    /dev/chassis/SYS/HDD1/disk
  2. c0t600000E00D000000000103460000000d0 <FUJITSU-ETERNUS_DXL-0000-50. 00GB>
    /scsi_vhci/disk@g600000e00d0000000001034600000000
  3. c0t600000E00D0000000001034600010000d0 <FUJITSU-ETERNUS_DXL-0000-50. 00GB>
    /scsi_vhci/disk@g600000e00d0000000001034600010000
  4. c0t600000E00D0000000001034600020000d0 <FUJITSU-ETERNUS_DXL-0000-50. 00GB>
    /scsi_vhci/disk@g600000e00d0000000001034600020000
  ~ (省略) ~
```

ii) 仮想ディスクから削除した物理ディスクの番号を入力します。

```
primary# format
searching for disks... done

AVAILABLE DISK SELECTIONS:
  0. c0t5000039698132D40d0 <TOSHIBA-AL13SEB600-3703-558. 91GB>
    /scsi_vhci/disk@g5000039698132d40
    /dev/chassis/SYS/HDD0/disk
  1. c0t500003970830F9E1d0 <TOSHIBA-AL13SEB600AL14SE-3703-558. 91GB>
    /scsi_vhci/disk@g500003970830f9e1
    /dev/chassis/SYS/HDD1/disk
  2. c0t600000E00D000000000103460000000d0 <FUJITSU-ETERNUS_DXL-0000-50. 00GB>
    /scsi_vhci/disk@g600000e00d0000000001034600000000
  3. c0t600000E00D0000000001034600010000d0 <FUJITSU-ETERNUS_DXL-0000-50. 00GB>
    /scsi_vhci/disk@g600000e00d0000000001034600010000
  4. c0t600000E00D0000000001034600020000d0 <FUJITSU-ETERNUS_DXL-0000-50. 00GB>
    /scsi_vhci/disk@g600000e00d0000000001034600020000
  ~ (省略) ~
Specify disk (enter its number): 4
```

- iii) 物理ディスクにアクセスできることを確認します。確認したあとに、[Ctrl]+[C]キーで終了します。

```

~ (省略) ~
Specify disk (enter its number): 4
selecting c0t600000E00D0000000001034600020000d0 <FUJITSU-ETERNUS_DXL-0000 cyl 6398 alt
2 hd 64 sec 256>
[disk formatted]

FORMAT MENU:
    disk      - select a disk
    type      - select (define) a disk type
    partition - select (define) a partition table
    current   - describe the current disk
    format    - format and analyze the disk
    repair    - repair a defective sector
    label     - write label to the disk
    analyze   - surface analysis
    defect    - defect list management
    backup    - search for backup labels
    verify    - read and display labels
    inquiry   - show disk ID
    volname   - set 8-character volume name
    !<cmd>    - execute <cmd>, then return
    quit
format>

```

- iv) ドメインが認識しているディスク構成を確認します。

```

ldom1# ls /dev/dsk
c1d0s0 c1d0s3 c1d0s6 c1d1s1 c1d1s4 c1d1s7 c1d2s2 c1d2s5
c1d0s1 c1d0s4 c1d0s7 c1d1s2 c1d1s5 c1d2s0 c1d2s3 c1d2s6
c1d0s2 c1d0s5 c1d1s0 c1d1s3 c1d1s6 c1d2s1 c1d2s4 c1d2s7

```

6) 不要なデバイスファイルを削除します。

- i) I/O デバイスの構成を更新します。

```
ldom1# devfsadm -C
```

- ii) ドメインが認識しているディスク構成を確認します。

```

ldom1# ls /dev/dsk
c1d0s0 c1d0s2 c1d0s4 c1d0s6 c1d1s0 c1d1s2 c1d1s4 c1d1s6
c1d0s1 c1d0s3 c1d0s5 c1d0s7 c1d1s1 c1d1s3 c1d1s5 c1d1s7

```

- ▶ 「c1d2sX」のデバイスが削除されていることを確認します。

7) 構成情報を更新します。

```
primary# ldem remove-spconfig config_initial  
primary# ldem add-spconfig config_initial
```

- ▶ 既存の構成情報を削除したあと、最新の構成情報を SP（サービスプロセッサ）に保存します。

Point

Oracle VM は、CPU、メモリ、および I/O デバイスの動的な構成変更が可能です。

ただし、設定・構成を変更した場合は、以下のコマンドを実行して必ず構成情報を保存してください。

```
# ldem add-spconfig config_initial
```

9. ドメインのバックアップとリストア

SP(サービスプロセッサ)から制御ドメインへコンソール接続し、作業を実施します。

本書では、Oracle VM の構成情報を使用して各ドメインの定義を退避・復元します。

各ドメインの定義を退避(復元)後、必要に応じてシステムボリュームとデータボリュームも退避(復元)してください。

Point

- Oracle VM 環境では、通常の OS のシステムボリュームやデータボリュームに加え、「Oracle VM の構成情報」もバックアップする必要があります。
- システムボリュームの退避・復元については、『[Oracle Solaris 11 を使ってみよう\(構築・運用手順書\)](#)』の「8. システムボリュームのバックアップ／リストア」を参照してください。

9.1. ゲストドメインのバックアップとリストア

9.1.1. ゲストドメインのバックアップ

1) ゲストドメインの状態を確認します。

本手順では、ldom1 のみバックアップとリストアを行います。

```
primary# ldm list-domain
NAME          STATE    FLAGS  CONS  VCPU  MEMORY  UTIL  NORM  UPTIME
primary       active  -n-cv-  UART   16    16G     0.1%  0.1%  23h 8m
ldom1         active  -n----- 5000    8    16G     0.1%  0.1%  44m
ldom2         active  -n----- 5001    8    16G     0.2%  0.2%  22h 54m
```

2) 構成情報を更新します。

既存の構成情報を削除したあと、最新の構成情報を SP(サービスプロセッサ)に保存します。

```
primary# ldm remove-spconfig config_initial
primary# ldm add-spconfig config_initial
```

3) ゲストドメインの環境設定を確認します。

```
primary# ldm list-constraints -p ldom1
VERSION 1.21
DOMAIN|name=ldom1
UUID|uuid=bc78190b-1f4a-4991-b9ac-90dbe548d290
~（省略）~

VDISK|name=vdisk0|vol=vol0@primary-vds0|timeout=|id=0
VDISK|name=vdisk_iso|vol=vol_iso@primary-vds0|timeout=|id=1
VCONS|group=|service=|port=5000|log==
```

4) ゲストドメインの環境設定ファイルを保存します。

```
primary# ldm list-constraints -x ldom1 > /OVM/ldom1.xml
```

- ▶ 保存先のディレクトリとファイル名は任意です。必要に応じて格納先ディレクトリを作成してください。

Point

本書では格納先ディレクトリを制御ドメインの/OVMに保存していますが、テープ装置や退避先サーバなどへ保存することを推奨します。

5) ゲストドメインが認識しているディスク、およびディスクのボリューム名を確認します。

```
primary# ldm list-domain -o disk ldom1
NAME
ldom1

DISK
```

NAME	VOLUME	TOUT	ID	DEVICE	SERVER	MPGROUP
vdisk0	vol0@primary-vds0		0	disk@0	primary	
vdisk_iso	vol_iso@primary-vds0		1	disk@1	primary	

- ▶ 本書では2つのディスクを認識させています。

6) 手順5)で確認したボリューム名から、ゲストドメインのディスクのバックエンドを確認します。

```
primary# ldm list-domain -o disk primary
NAME
primary

VDS
```

NAME	VOLUME	OPTIONS	MPGROUP	DEVICE
primary-vds0	vol0			
	/dev/dsk/c0t600000E00D000000000103460000000d0s2			
	vol_iso	ro		/ISO/Sol11.4.iso
	vol1			
	/dev/dsk/c0t600000E00D0000000001034600010000d0s2			
	vol_iso1	ro		/ISO/Sol11.4.iso

7) ゲストメインのシステムボリュームやデータボリュームをバックアップします。

任意の方法で、テープ装置や退避先サーバなどへシステムボリュームやデータボリュームのバックアップを実施します。

9.1.2. ゲストメインのリストア

1) ゲストメインを停止し、アンバインドします。

```
primary# ldm stop-domain ldom1
primary# ldm unbind-domain ldom1
```

2) ゲストメイン「ldom1」を削除します。

ゲストメインのリストアが必要な状況を想定します。

```
primary# ldm remove-domain ldom1
```

3) ゲストメイン環境を確認します。

```
primary# ldm list-domain
NAME          STATE      FLAGS    CONS    VCPU  MEMORY  UTIL  NORM  UPTIME
primary       active    -n-cv-   UART    16    16G     0.1%  0.1%  23h 25m
ldom2         active    -n----- 5001     8    16G     0.0%  0.0%  23h 11m
```

4) 環境設定ファイルから、ゲストメインの再構築を行います。

```
primary# ldm add-domain -i /OVM/ldom1.xml
```

5) ドメインの構成を確認します。

ゲストメイン ldom1 が復元されていることを確認してください。

```
primary# ldm list-domain
NAME          STATE      FLAGS    CONS    VCPU  MEMORY  UTIL  NORM  UPTIME
primary       active    -n-cv-   UART    16    16G     0.0%  0.0%  23h 27m
ldom2         active    -n----- 5001     8    16G     0.0%  0.0%  23h 12m
ldom1         inactive  -----      8      16G
```

6) ゲストメインをバインドします。

```
primary# ldm bind-domain ldom1
```

7) ゲストメインを起動します。

```
primary# ldm start-domain ldom1
```

8) ゲストメインの状態を確認します。

```
primary# ldm list-domain
NAME          STATE      FLAGS    CONS    VCPU  MEMORY  UTIL  NORM  UPTIME
primary       active    -n-cv-   UART    16    16G     0.3%  0.3%  23h 28m
ldom1         active    -n----- 5000     8    16G     31%   31%   18s
ldom2         active    -n----- 5001     8    16G     0.2%  0.2%  23h 14m
```

9) ゲストドメインのシステムボリュームやデータボリュームをリストアします。

任意の方法で、テープ装置や退避先サーバなどからシステムボリュームやデータボリュームのリストアを実施します。

9.2. 制御ドメインのバックアップとリストア

9.2.1. 制御ドメインのバックアップ

1) 構成情報を更新します。

```
primary# ldmd remove-spconfig config_initial
primary# ldmd add-spconfig config_initial
```

2) 制御ドメインの環境設定を保存します。

```
primary# ldmd list-constraints -x primary > /OVM/primary.xml
```

3) ゲストドメインの環境設定を保存します。

```
primary# ldmd list-constraints -x ldom1 > /OVM/ldom1.xml
primary# ldmd list-constraints -x ldom2 > /OVM/ldom2.xml
```

▶ 制御ドメインのリストア後にゲストドメインもリストアするため、環境設定を保存します。

Point

本書では、格納先ディレクトリを制御ドメインの/OVM に保存していますが、テープ装置や退避先サーバなどへ保存することを推奨します。

4) 制御ドメインのシステムボリュームやデータボリュームをバックアップします。

任意の方法で、テープ装置や退避先サーバなどへシステムボリュームやデータボリュームのバックアップを実施します。

9.2.2. 制御ドメインのリストア

Point

SR-IOV 機能を使用している場合、構成情報ファイルによる制御ドメインの復元後に、手動で SR-IOV の設定を元に戻す必要があります。

詳細は、以下を参照してください。

- 『SR-IOV (Single Root I/O Virtualization) ドメイン環境 構築手順書』

<https://www.fujitsu.com/jp/sparc-technical/document/solaris/#ovm>

- 1) SP(サービスプロセッサ)の設定を初期状態に戻します。

```
primary# ldmd set-spconfig factory-default
primary# ldmd list-spconfig
factory-default [next poweron]
config_initial [current]
```

- 2) OS を停止します。

```
primary# shutdown -y -g0 -i0
```

- 3) XSCF に遷移します。

```
{0} ok #. ←シャープ+ドット
```

- 4) ハードウェアの電源を停止します。

```
XSCF> poweroff -p 0
PPAR-IDs to power off :00
Continue? [y|n] :y
```

- 5) ハードウェアの電源が停止していることを確認します。

Status が「Powered Off」であることを確認してください。

```
XSCF> showpci -p 0
PPAR-ID  LSB  PSB  Status
00
          00  00-0  Powered Off
```

停止しない場合は、-f オプションを指定して実行します。

```
XSCF> poweroff -f -p 0
PPAR-IDs to power off :00
The -f option will cause domains to be immediately reset.
Continue? [y|n] :y
```

6) ハードウェアの電源を起動します。

```
XSCF> poweron -p 0
PPAR-IDs to power on :00
Continue? [y|n] :y
```

7) OS のコンソールに接続します。

```
XSCF> console -p 0

Console contents may be logged.
Connect to PPAR-ID 0?[y|n] :y
```

8) 制御ドメインのシステムボリュームやデータボリュームをリストアします。

任意の方法で、テープ装置や退避先サーバなどからシステムボリュームやデータボリュームのリストアを実施します。

※ リストア後、一般ユーザでログインしたあと、「su -」を実行して root ユーザに切り替えてください。

9) SP の設定が初期状態であることを確認します。

```
primary# ldm list-spconfig
factory-default [current]
config_initial
```

10) ドメインの状態を確認します。

```
primary# ldm list-domain
NAME          STATE      FLAGS    CONS    VCPU    MEMORY    UTIL    NORM    UPTIME
primary       active    -n-c--   UART    48      128512M   0.0%    0.0%    4m
```

11) ドメインの環境設定ファイルから、制御ドメインを復元します。

```
primary# ldm init-system -r -i /OVM/primary.xml
```

▶ 「-r」オプションにより、システムが自動で再起動され、制御ドメインが復元されます。

12) 制御ドメインの状態を確認します。

制御ドメインのリソースが、元に戻っていることを確認してください。

```
primary# ldm list-domain
NAME          STATE      FLAGS    CONS    VCPU    MEMORY    UTIL    NORM    UPTIME
primary       active    -n-cv-   UART    16      16G      0.0%    0.0%    4m
```

《参考》 I/O ルートドメインを作成している場合

制御ドメインのリストアにより、PCIe バスは制御ドメインに割り当てられます。

そのため、I/O ルートドメインに割り当てる PCIe バスを制御ドメインから割り当て解除します。

PCIe バス名は事前に確認してください（以下のコマンド実行例は pci@8100 の場合）。

【書式】ldm remove-io (rm-io) [-n] <PCIe バス名> <PCIe バスを切り離すドメイン>

```
primary# ldm remove-io pci@8100 primary
```

13) ゲストドメインを復元します。

```
primary# ldm add-domain -i /OVM/ldom1.xml
primary# ldm add-domain -i /OVM/ldom2.xml
```

14) ゲストドメインの状態を確認します。

ゲストドメインのリソースが、元に戻っていることを確認してください。

```
primary# ldm list-domain
```

NAME	STATE	FLAGS	CONS	VCPU	MEMORY	UTIL	NORM	UPTIME
primary	active	-n-cv-	UART	16	16G	0.0%	0.0%	6m
ldom1	inactive	-----		8	16G			
ldom2	inactive	-----		8	16G			

15) ゲストドメインを起動します。

```
primary# ldm bind-domain ldom1
primary# ldm bind-domain ldom2
primary# ldm start-domain -a
```

16) 復元した構成情報を SP (サービスプロセッサ) に保存します。

```
primary# ldm add-spconfig config_restore
```

- ▶ 構成情報の名前は任意です。

付録 1. Oracle VM Server for SPARC の初期化

付録 1.1. ゲストドメインの削除

事前にゲストドメインを停止し、ゲストドメインのリソースを解除します。

すでにゲストドメインが停止している場合、手順 1)は不要です。

また、すでにゲストドメインのリソースを解除している場合、手順 2)は不要です。

- 1) すべてのゲストドメインを停止します。

```
primary# ldm stop-domain ldom1
primary# ldm stop-domain ldom2
```

- 2) ゲストドメインをアンバインドします。

```
primary# ldm unbind-domain ldom1
primary# ldm unbind-domain ldom2
```

- 3) ゲストドメインの定義情報を削除します。

【書式】ldm remove-domain (-a | <ドメイン名>...)

SF ldm remove-domain → ldm destroy

```
primary# ldm remove-domain ldom1
primary# ldm remove-domain ldom2
```

- 4) ゲストドメインが削除されている(表示されない)ことを確認します。

```
primary# ldm list-domain
NAME          STATE      FLAGS    CONS    VCPU  MEMORY  UTIL  NORM  UPTIME
primary       active     -n-cv-   UART    16    16G     0.1%  0.1%  13m
```

付録 1.2. 仮想環境の初期化

- 1) SP(サービスプロセッサ)に設定した構成情報を確認します。

```
primary# ldm list-spconfig
factory-default
config_restore
config_initial [next poweron]
```

- 2) SP(サービスプロセッサ)の設定を初期状態に戻します。

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

- 3) SP(サービスプロセッサ)に設定した構成情報を削除します。

「factory-default」以外の構成情報をすべて削除します。

```
primary# ldm remove-spconfig [config 名]
```

4) 再起動後の Oracle VM Server for SPARC の構成を確認します。

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

5) 仮想ネットワークターミナルサーバデーモン(vntsd)を停止します。

```
primary# svcadm disable vntsd
```

6) OS を停止します。

```
primary# shutdown -y -g0 -i0
```

7) XSCF に遷移します。

```
{0} ok #. ←シャープ+ドット
```

8) ハードウェアの電源を停止します。

```
XSCF> poweroff -p 0
PPAR-IDs to power off :00
Continue? [y|n] :y
```

9) ハードウェアの電源が停止していることを確認します。

Status が「Powered Off」であることを確認します。

```
XSCF> showpci -p 0
PPAR-ID  LSB   PSB   Status
00              00-0  Powered Off
```

停止しない場合は、-f オプションを指定して実行します。

```
XSCF> poweroff -f -p 0
PPAR-IDs to power off :00
The -f option will cause domains to be immediately reset.
Continue? [y|n] :y
```

Point

構成情報をすべて削除しても、XSCF のリセット(電源 ON/OFF)を行わないと、Oracle VM の環境は完全に削除されません。

《参考》XSCF 上 (SPARC M12/M10) で構成情報を変更する方法

(1) ドメインの状態を一括で確認できます。

【書式】showdomainstatus -p <物理パーティション ID> [-v] [-M] [-g <ドメイン名>]

```
XSCF> showdomainstatus -p 0
Logical Domain Name  Status
primary              Solaris running
```

▶ 制御ドメインのステータスを確認できます。

(2) SP (サービスプロセッサ) に保存されている構成情報を確認します。

```
XSCF> showdomainconfig -p 0
PPAR-ID      :0
Booting config
(Current)    :config_restore
(Next)       :config_restore

-----

Index        :1
config_name   :factory-default
domains      :1
date_created:-

-----

Index        :2
config_name   :config_initial
domains      :3
date_created:' 2018-10-26 16:29:55'

-----

Index        :3
config_name   :config_restore
domains      :3
date_created:' 2018-10-26 17:07:20'
```

※ 構成情報の一覧、現在の構成情報(Current)、および電源再投入後に有効になる構成情報(Next)を確認できます。

(3) 次回電源投入時の構成を出荷時の状態(factory-default)にします。

【書式】setdomainconfig [[-q] [-y|n]] -p <物理パーティション ID> -c default

【書式】setdomainconfig [[-q] [-y|n]] -p <物理パーティション ID> -i [構成情報番号(index)]

```
XSCF> setdomainconfig -y -p 0 -c default
```

※ 変更する構成は factory-default 以外も選択可能です。

例:XSCF> setdomainconfig -y -p 0 -i 3

（4）現在の構成情報を確認します。

（Next）が「factory-default」であることを確認してください。これで電源再投入後、ドメインの構成が初期化されます。

```
XSCF> showdomainconfig -p 0
PPAR-ID      :0
Booting config
  (Current)   :config_restore
  (Next)      :factory-default
```

～（省略）～

改版履歴

改版日	版数	改版内容
2012 年 1 月	1.0 版	新規作成
2015 年 5 月	2.0 版	Oracle Solaris 11.2 および Oracle VM Server for SPARC 3.2 に対応
2016 年 5 月	3.0 版	レイアウトデザインと構成を更新 Oracle Solaris 11.3 および Oracle VM Server for SPARC 3.3 に対応
2017 年 12 月	3.1 版	レイアウトデザインと構成を更新 SPARC M12 に対応 Oracle VM Server for SPARC 3.5 に対応
2019 年 6 月	4.0 版	Oracle Solaris 11.4 および Oracle VM Server for SPARC 3.6 に対応

