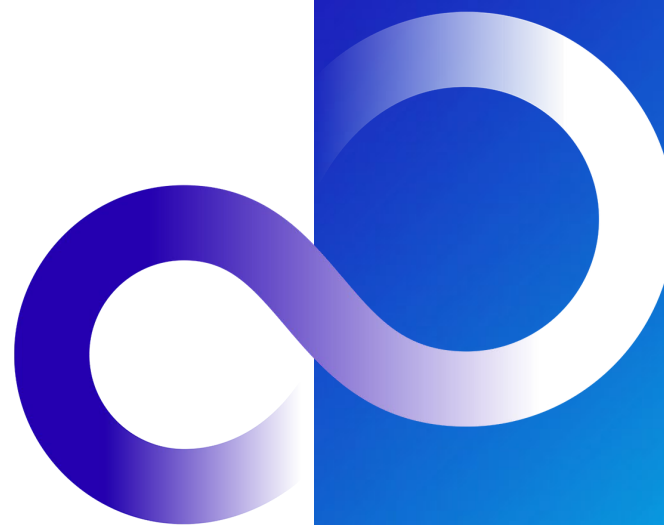


Linuxユーザのための Oracle Solarisの使い方

2016年3月（第1.1版）

富士通株式会社



- はじめに
- 1. OS環境の起動と停止
- 2. パッケージ管理
- 3. ユーザ管理
- 4. ネットワーク管理
- 5. サービス管理
- 6. ファイルシステムとストレージ管理
- 7. 監視
- 8. 仮想環境

■ 目的

- 本書は、Linux 環境のシステムを運用しているユーザ向けに、Oracle Solaris の使い方について解説しています。

■ 対象読者

- Linuxの基礎知識を有している方。
- Oracle Solarisのシステムを運用する予定の方。

■ ドキュメントの位置付け



■ 留意事項

- 本書ではOracle Solaris を「Solaris」と記載することがあります。
- 本書ではOracle VM Server for SPARCを「Oracle VM」と記載することがあります。
- 本書で解説しているコマンド等は以下の環境を元にしています。
 - Linux : Red Hat Enterprise Linux 6.5、Red Hat Enterprise Linux 7.1
 - Solaris: Oracle Solaris 11.3、ESF5.1
- Solarisの機能についてのスライドには、右記の印を表示しています。

Solaris

- SolarisはLinuxと同じようにコマンドベースで操作できるから、それほど違いはないだろう、と思っていたが・・・

再起動の方法が
わからない。

ZFSって何？
UFSは利用できないの？

パッチを適用したい
が、コマンドは・・・。



サービスの状態を
確認する方法は？

IPアドレスの変更
は、どうやるの？

本書は、そのようなLinuxユーザ向けにSolarisの運用シーンで必要となる操作や機能について解説しています。

1. OS環境の起動と停止

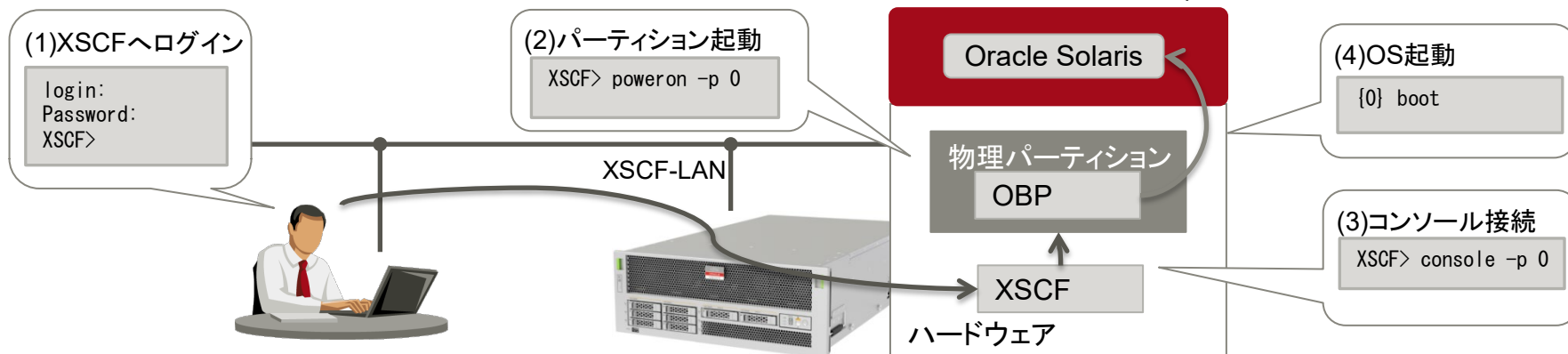
■ SPARC M10サーバはXSCFから操作

■ XSCF (eXtended System Control Facility) とは

- PRIMERGYのiRMC (Remote Management Controller) に相当します。
- 本体処理装置とは独立した専用プロセッサで稼働し、ハードウェア／OSの状態監視や通知機構などを持っています。
- リモート(XSCF-LAN)で接続し、物理パーティションの電源ON／OFFが可能です。

■ ネットワーク経由でのOS起動の流れ

- (1) ターミナルソフトウェアを使用し、XSCFへログインします。
 - (2) XSCFのコマンドで物理パーティションを起動します。
 - (3) XSCFのコマンドでコンソールログインします。
 - (4) OBPのコマンドでSolaris OSを起動します(※)。
- ※ OBPのパラメータ(auto-boot?)がfalseの場合に実行します。
パラメータがtrueの場合、poweronの後自動的にOSが起動します。



■ ランレベルの比較

- LinuxのRHEL6と同様にSolarisにも0から6までのランレベルが存在しますが、一部の意味が異なります。（RHEL7では、targetとなりました）

ランレベル／target		意味		備考
RHEL6 Solaris	RHEL7	Red Hat Enterprise Linux	Oracle Solaris	
0	poweroff.target	電源停止状態	OS停止状態(*1)	*1: OBPは起動状態
s(またはS)		-	シングルユーザ	
1	rescue.target	シングルユーザ	システム管理状態	
2		未使用	マルチユーザ(NFS無し)	
3	multi-user.target	マルチユーザ	マルチユーザ(*2)	*2: Solarisデフォルト
4		未使用	マルチユーザ(*3)	*3: 必要に応じて使用可
5	graphical.target	Xウィンドウ(*4)	電源停止状態	*4: Linuxデフォルト
6	reboot.target	OS再起動	OS再起動	

- ・Solaris OSのランレベルで覚えておきたいのは、0(OS停止)、s(シングルユーザ)、3(マルチユーザ)、5(電源停止)、6(OS再起動)です。
- ・ランレベルによって変わるサービス起動については、5. サービス管理 を参照してください。

■ LinuxとSolarisのOS起動

■ Linuxの場合

- ハードウェアの電源投入後、GRUB2環境から起動カーネルを選択します。
(RHEL6はGRUB)
- GRUB2環境からオプションの修正で、シングルユーザモードでの起動も可能です。

■ Solarisの場合

- ハードウェアの電源投入後、OBP (Open Boot PROM) 環境から起動します。
- OBPはPCサーバのBIOSとLinuxのGRUBを合わせたような環境です。
- ok と表示されるプロンプト (通称「okプロンプト」) 上で専用コマンドを使用します。

実行例

- ✓ Solaris OSの起動

```
{0} ok boot
```

・OBP環境では起動ディスクの指定や起動オプションの指定 (シングルユーザモード等) が可能です。
他にも、接続デバイスの確認や各種OBPパラメータの設定などを実施します。

■ SolarisのGUIデスクトップの起動

■ VNCを利用したGUI接続(※VNCはフリーソフトです)

実行例

1. solaris-desktopパッケージのインストール

```
# pkg install solaris-desktop
```

2. /etc/gdm/custom.confファイルの編集

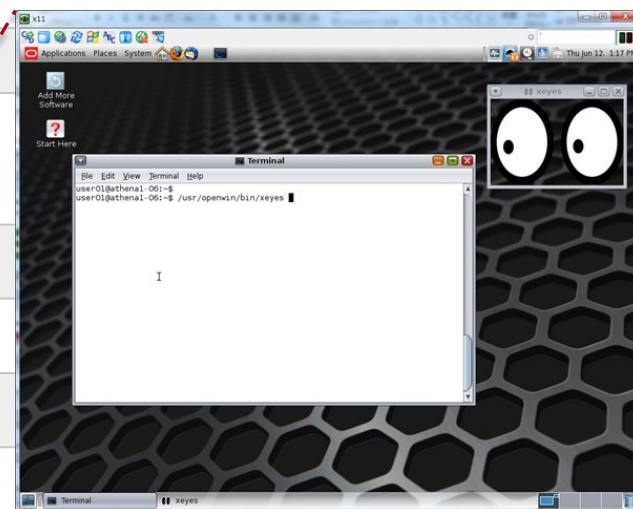
3. システムの再起動

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

4. xvnc inetdサービスを有効にします。

```
# inetadm -e xvnc-inetd
```

5. PCからVNCクライアントを利用してログインします。



・詳しくは「Oracle Solaris 11.3 デスクトップ管理者ガイド」を参照してください。

http://docs.oracle.com/cd/E62101_01/html/E62852/gmdah.html#scrolltoc

■ LinuxとSolarisのOS停止

■ Linuxの場合

• RHEL6

実行例

✓ Linux OSの停止

```
# shutdown -h now
```

✓ Linux OSの再起動

```
# shutdown -r now
```

• RHEL7

実行例

✓ Linux OSの停止

```
# systemctl poweroff
```

✓ Linux OSの再起動

```
# systemctl reboot
```

・RHEL7では互換性を維持するため、引き続きshutdownコマンドも使用可能ですが、systemctlコマンドの使用が推奨されます。

■ Solarisの場合

- Solarisでは、RHEL6と同様にshutdownコマンドを使用しますが、オプションの指定に違いがあります。
- -gオプションで停止までの時間(秒)を指定し、-iオプションでOSの停止や再起動等の操作を指定します。

実行例

✓ Solaris OSの停止

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

✓ Solaris OSの再起動

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

■ 覚えておきたいOS起動／停止のコマンド

基本操作	Red Hat Enterprise Linux		Oracle Solaris
	RHEL6	RHEL7	
OS起動	GRUBのメニューから起動。	GRUB 2のメニューから起動。	OBP (Open Boot PROM) 環境から起動コマンドを実行。 {0} ok boot
シングルユーザモードで起動	ランレベルを追記。 kernel /vmlinuz-... root=... 1	ランレベルを追記。 systemd.unit=rescue.target	OBP (Open Boot PROM) 環境からシングルユーザモードで起動。 {0} ok boot -s
OS停止	shutdownコマンドで停止オプション(-h)指定。 # shutdown -h now	systemctlコマンドで停止オプション(poweroff)指定。 # systemctl poweroff	shutdownコマンドを実行。停止オプション(-i5)を指定。 # shutdown -y -g0 -i5
OS再起動	shutdownコマンドで再起動オプション(-r)を指定。 # shutdown -r now	systemctlコマンドで再起動オプション(reboot)指定。 # systemctl reboot	shutdownコマンドを実行。再起動オプション(-i6)を指定。 # shutdown -y -g0 -i6
その他の実行例	<ul style="list-style-type: none"> ・10時に停止 # shutdown -h 10:00 ・5分後に停止 # shutdown -h +5 	<ul style="list-style-type: none"> ・OSをサスペンド # systemctl suspend ・OSを休止 # systemctl hibernate 	<ul style="list-style-type: none"> ・ネットワーク経由で起動 {0} ok boot net:dhcp ・30秒後に再起動 # shutdown -y -g30 -i6

詳細は「LinuxユーザのためのOracle Solarisコマンド実例集」を参照。

■ Solaris標準のロケール(文字コード)はUTF-8

■ システムロケールは以下のサービスのプロパティで設定

「svc:/system/environment:init」

■ システムロケールの変更

実行例

1. system/locale/extraパッケージのインストール

※標準ロケール以外を使用する場合のみ

```
# pkg install system/locale/extra
```

2. svccfgコマンドでロケールを変更(LANG=Cに設定)

```
# svccfg -s system/environment:init setprop environment/LANG = astring: C
```

3. サービスのプロパティの再読み込み

```
# svcadm refresh system/environment:init
```

4. 設定反映の確認

```
# svcprop system/environment:init | grep environment/LANG
environment/LANG astring C
```

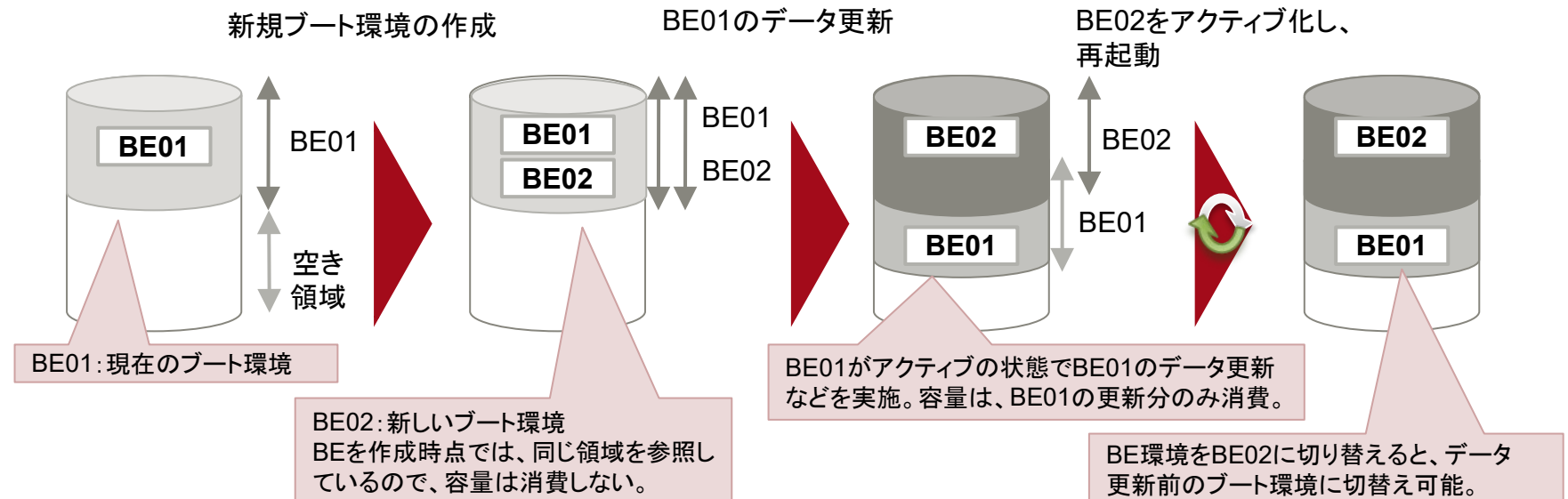
主なロケール(文字コード)	設定値
英語	C
日本語(EUC)	ja_JP.eucJP、ja
日本語(Shift-JIS)	ja_JP.PCK
日本語(UTF-8)	ja_JP.UTF-8

・設定可能なロケールは、locale -aコマンドで確認できます。

■ ブート環境の管理

- Boot Environment (BE) は、ブート環境を作成・削除・複製する機能です。
 - ・ 既存のブート環境から取得したスナップショットを元に、新しいBEを作成します。
 - ・ BEを選択してOSを再起動するだけで、新しいブート環境に切替えることができます。
 - ・ スナップショットを利用するので、ディスク容量はデータ更新分しか消費しません。
 - ・ Linuxでもアップデート前のカーネルをGRUBで選択・起動することができますが、SolarisではBEの機能によって実現しています。

BEのイメージ

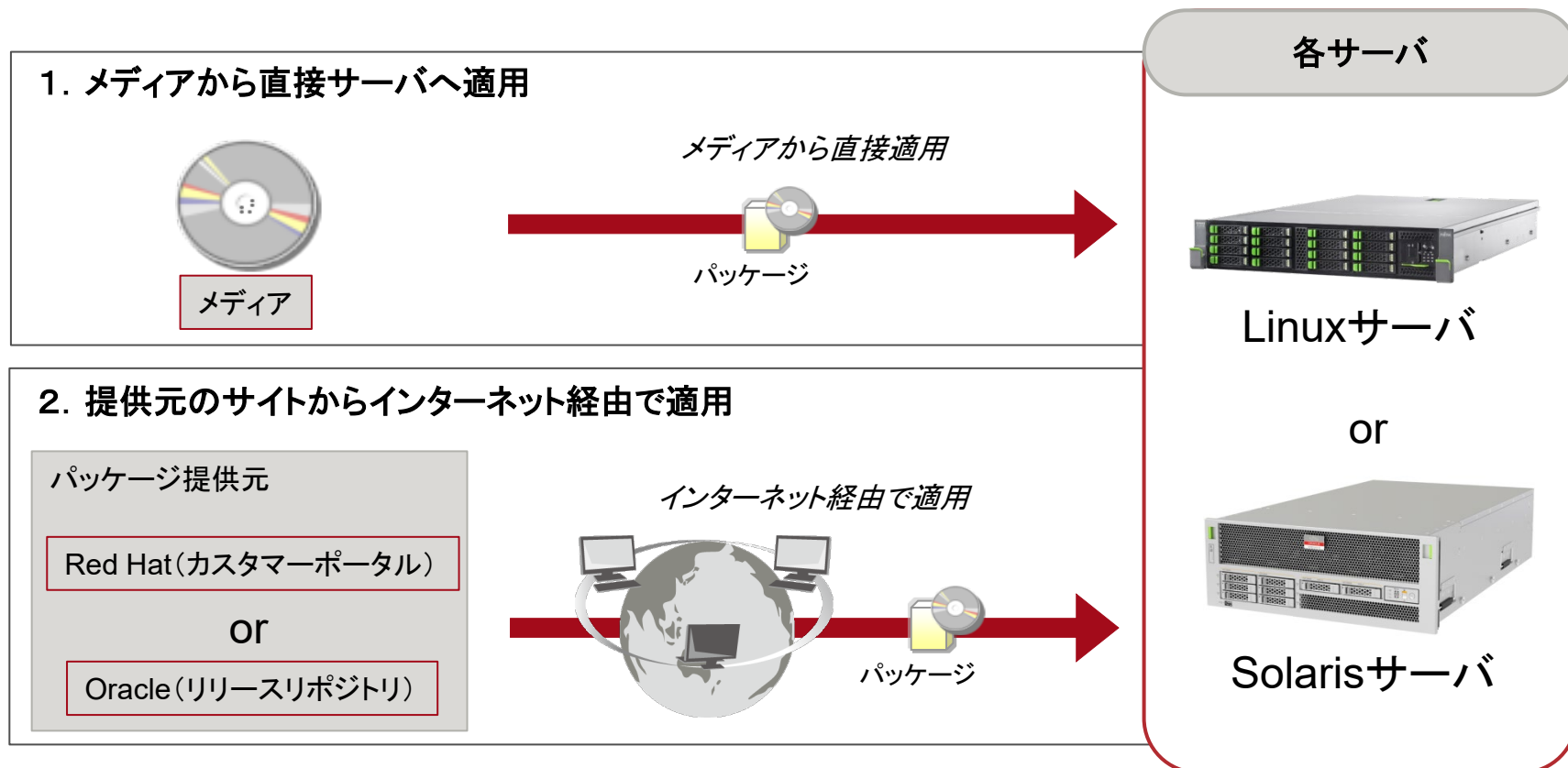


2.パッケージ管理

■ LinuxとSolarisのパッケージ適用方法

■ パッケージ適用の方法は基本的には同じであり、以下の2通りがあります。

- ・メディアから直接サーバへ適用。
- ・提供元のサイトからインターネット経由で適用。



■ ローカル環境のパッケージ提供サーバ(ローカルリポジトリサーバ)

■ ローカルリポジトリサーバの利用

- ・ 外部ネットワーク(リリースリポジトリ)に接続できないサーバが利用します。
- ・ 各サーバ毎にメディアを使用する必要がなく、パッケージの一元管理ができます。
- ・ ローカルリポジトリサーバにSRU(サポートリポジトリ)を適用し提供用パッケージを最新化します。

※ リリースリポジトリ … Oracle Solarisリリースごとに更新されるパッケージを提供
SRU (Support Repository Update) … 修正パッケージ

リポジトリサーバの利用



※SRUを入手するには富士通サポートデスク契約が必要です。

■ 覚えておきたいパッケージ管理のコマンド

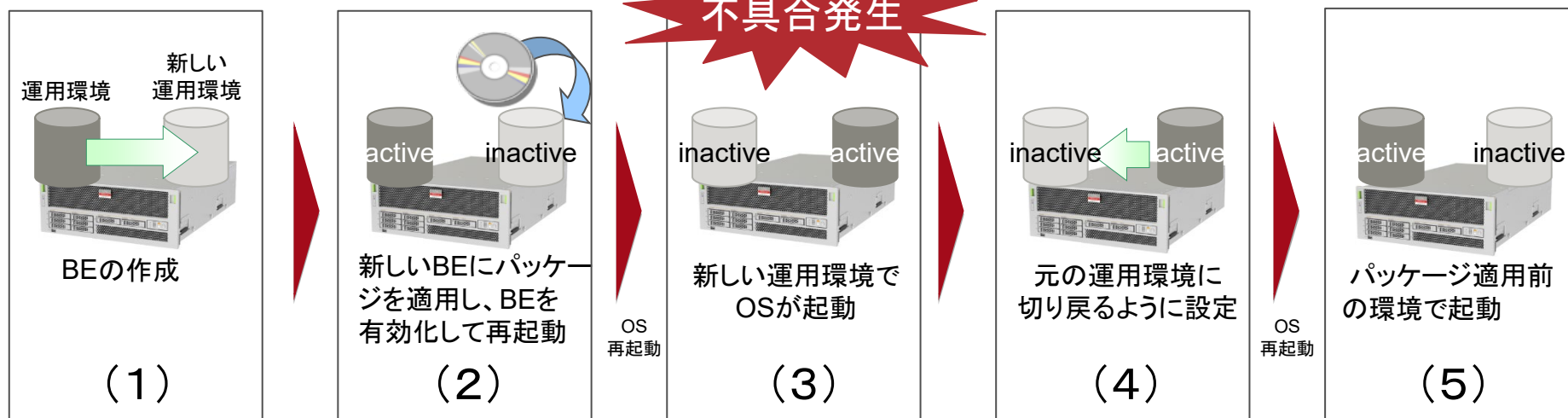
基本操作	Red Hat Enterprise Linux		Oracle Solaris
	RHEL6	RHEL7	
更新の確認	yum check-update		pkg update -nv
パッケージの更新	yum update パッケージ名		pkg update パッケージ名
全パッケージの更新	yum update		pkg update
インストール	yum install パッケージ名		pkg install パッケージ名
検索	yum search 検索文字列		pkg search 検索文字列
インストール済みの一覧	yum list all		pkg list
パッケージ情報の表示	yum info パッケージ名		pkg info パッケージ名

詳細は「LinuxユーザのためのOracle Solarisコマンド実例集」を参照。

■ Boot Environment (BE) による環境復元

- SolarisのBEを利用すると、パッケージ適用前の環境に戻すことができます。
- パッケージ適用が原因でOSが起動しなくなった場合は、OBP上からパッケージ適用前のBEを選択して起動することができます。

BE切り戻しのイメージ



・BE環境はシステムバックアップではありません。ディスク故障などの障害に備え、システムバックアップの取得が必要です。

■ BEによるOS環境の復元操作

実行例 (OS環境上でBEを指定する場合)

1. BEの一覧から起動するBE名を確認します。

```
# beadm list
BE                Active Mountpoint Space Policy Created
--                -
solaris-1         -      -          9.67M static 2012-11-06 15:08
solaris-2         NR      /          3.82G static 2012-11-06 15:30
```

Activeのステータス
N: 現在のブート環境
R: 次回のブート環境

2. 元の運用環境を有効にして、OSを再起動します。

```
# beadm activate solaris-1
# shutdown -y -g0 -i6
```

実行例 (OBP環境上でBEを指定する場合)

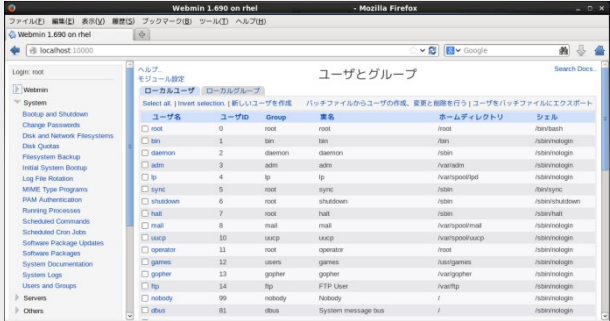
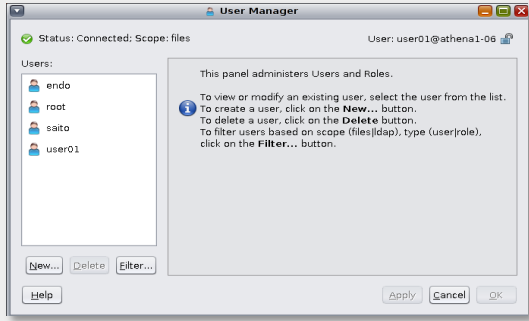
1. BEの一覧から起動するBEの番号を指定します。(boot -Lを実行)

```
{0} ok boot -L
Boot device: /virtual-devices@100/channel-devices@200/disk@1 File and args: -L
1 solaris-1
2 solaris-2
Select environment to boot: [ 1 - 2 ]:1
```

2. 表示された起動コマンドを実行します。

```
{0} ok boot -Z rpool/ROOT/solaris-1
```

3. ユーザ管理

基本操作	Red Hat Enterprise Linux	Oracle Solaris
操作	<p>コマンドラインでの管理が一般的 ※GUI管理も可能</p> 	<p>コマンドラインでの管理が一般的 ※GUI管理も可能</p> 
root ユーザ/役割	<ul style="list-style-type: none"> ユーザ 全ての一般ユーザがrootユーザに変更可能。 	<ul style="list-style-type: none"> 役割(ユーザにすることも可能) 特定の一般ユーザのみrootの役割を引き受けることができる。
グループ	<ul style="list-style-type: none"> デフォルトでユーザ名と同じ名前のグループに所属している。 	<ul style="list-style-type: none"> ユーザはグループの他に、プロジェクトという分類にも属しており、IPCパラメータ(※)などの資源制御の単位となる。

※ IPC (Inter Process Communication)

- ユーザ管理のコマンド (useradd, usermod, userdel) はLinux、Solarisともに同じ名称ですが、オプションにより意味が異なる場合がありますので注意が必要です。
- IPCパラメータの詳細は、「Oracle Solaris IPCパラメーター設定ガイド」を参照ください。

<http://www.fujitsu.com/jp/sparc-technical/document/#solaris11-ipc>

■ ローカル認証用のファイル

- アカウント情報は/etc/passwdファイルに保存されています。
- パスワード情報は/etc/shadowファイルに暗号化され保存されています。
 - ・暗号化方式: Hash (Solaris: SHA-256、Linux: SHA-512)

■ ログイン時のパラメータ

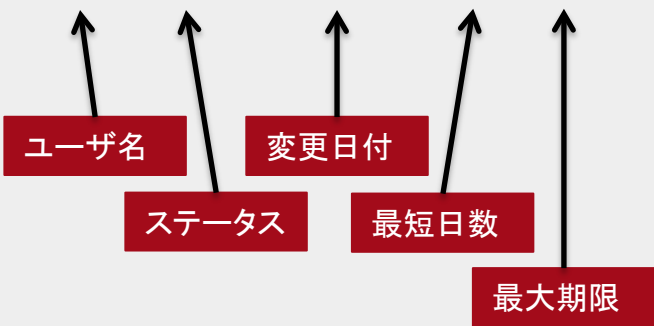
- ログイン時のパラメータは/etc/default/loginファイルに設定されます。

変数	説明
CONSOLE	設定するとスーパーユーザはそのデバイスにのみログイン可能
PATH	初期シェルのPATH変数
SUPATH	スーパーユーザ用初期シェルのPATH変数
TIMEOUT	ログインセッションを終了するまでの待ち時間(秒)
SLEEPTIME	ログイン失敗メッセージを画面に表示するまでに待つ秒数
RETRIES	ログインを再試行する回数

- ・アカウントやパスワードを設定/変更する際には、認証用のファイル(/etc/passwd、/etc/shadow)を直接編集するのではなく、各コマンド(usermod、passwd など)を使用して、設定/変更します。
- ・暗号化方式(Hash)はデフォルト設定であり、変更も可能です。

■ パスワードの有効期限の設定

- Linuxではユーザパスワードの有効期限の設定は、chageコマンドを使用しますが、Solarisではpasswdコマンドを使用します。

基本操作	Red Hat Enterprise Linux	Oracle Solaris
最大の期限を設定	# chage -M 90 ユーザ名	# passwd -x 90 ユーザ名
最短日数を設定	# chage -m 30 ユーザ名	# passwd -n 30 ユーザ名
パスワードの期限情報を表示する	<pre># chage -l ユーザ名 最終パスワード変更日 : 7月 29, 2015 パスワード期限: : なし パスワード無効化中 : なし アカウント期限切れ : なし パスワードが変更できるまでの最短日数 : 30 パスワードを変更しなくてよい最長日数 : 90 パスワード期限が切れる前に警告される日数: 7</pre>	<pre># passwd -s ユーザ名 admin PS 07/29/15 30 90</pre> 

- ・ユーザパスワードの有効期限設定については、コマンド、オプション共に異なりますが、Solarisでも同様に設定できます。

■ 覚えておきたいユーザ管理のコマンド

基本操作	Red Hat Enterprise Linux		Oracle Solaris
	RHEL6	RHEL7	
ユーザの追加	useradd ユーザ名		useradd ユーザ名
ユーザIDの更新	usermod -u 新UID ユーザ名		usermod -u 新UID ユーザ名
ユーザの削除	userdel ユーザ名		userdel ユーザ名
パスワードの有効期限表示	chage -l ユーザ名		passwd -s ユーザ名
パスワードの有効期限設定	chage -m 60 ユーザ名		passwd -x 60 ユーザ名
アカウント情報(フルネームなど)を変更する	chfn		passwd -g

詳細は「LinuxユーザのためのOracle Solarisコマンド実例集」を参照。

・ユーザ管理コマンドに関して、基本的な追加/更新/削除はSolarisでも同様に実行可能ですが、相当するコマンドが存在しないものもあります。

■ デフォルトシェルの変更

デフォルトのシェルはLinux,Solarisともにbashです。

■ Linuxの場合は、chshコマンドを使用

実行例

1. ユーザ(user01)のデフォルトシェルを変更する。

```
# chsh user01
user01 のシェルを変更します。
新しいシェル [/bin/bash]: /bin/sh
シェルを変更しました。
```

■ Solarisの場合は、passwdコマンドを使用

実行例

1. ユーザ(user01)のデフォルトシェルを変更する。

```
# passwd -e user01
古いシェル: /usr/bin/bash
新しいシェル: /bin/sh
passwd: user01 のパスワード情報が変更されました
```

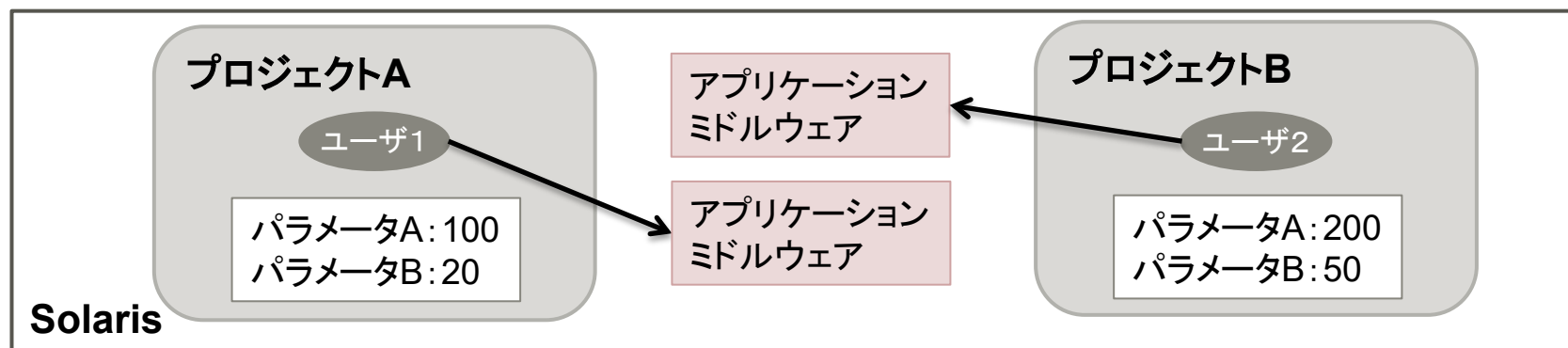
■ プロジェクトによる資源制御

■ プロジェクトとは

- Solarisでは資源制御を実行する単位としてプロジェクトという概念があります。
- ユーザはいずれかのプロジェクトに所属しており、ユーザが実行するアプリケーションやプロセスには、所属するプロジェクトに設定された資源制御が実行されます。

■ 資源制御の単位

- プロジェクト単位に設定するため、アプリケーションやミドルウェアを実行するユーザ別に資源制御が可能です。



- 資源にはプロセスに対するCPU使用時間、コアファイルサイズ、最大ヒープサイズやIPCパラメータなどが含まれます。プロジェクトのパラメータ設定はOS稼働中に設定／変更が可能です。

4. ネットワーク管理

	Red Hat Enterprise Linux		Oracle Solaris
	RHEL6	RHEL7	
IPアドレスの設定	<ul style="list-style-type: none"> 設定ファイルを編集し、networkサービスを再起動する。 	<ul style="list-style-type: none"> nmcli コマンド 	<ul style="list-style-type: none"> ipadmコマンドで設定 設定後、定義ファイルが自動で更新される。
ネットワークインタフェース名	<ul style="list-style-type: none"> ethXX 例: eth0, eth1 	<ul style="list-style-type: none"> 物理的なデバイスの接続情報に基づいて、割り当てられる 例: ens1p1, enp0s25 	<ul style="list-style-type: none"> 物理デバイス名を元に作成された論理デバイス名 (netXX) 例: net0, net1
データリンク層の管理	<ul style="list-style-type: none"> ethtool コマンド 		<ul style="list-style-type: none"> dladm コマンド
IP層の管理	<ul style="list-style-type: none"> ifconfig コマンド 	<ul style="list-style-type: none"> ip コマンド 	<ul style="list-style-type: none"> ipadm コマンド
冗長化機能	<ul style="list-style-type: none"> Bonding 		<ul style="list-style-type: none"> IPMP

• Solarisでは、Solaris 11からネットワーク管理の仕組みやコマンド体系が大きく変わっています。ネットワークの仮想化や、OS標準機能によるネットワーク冗長構成も可能です。

■ Solaris

■ dladmコマンド

- データリンク層を管理するコマンド。
- 仮想インタフェースの作成やネットワークの冗長化を設定する。

■ ipadmコマンド

- IP層を管理するコマンド。
- IPアドレスは、“インタフェース名/任意の文字列” 形式のアドレスオブジェクトとして管理され、設定、削除はアドレスオブジェクトを対象として行う。

■ Linux

■ ifconfigコマンド(RHEL6)

- ネットワークの設定や状態を表示するコマンド。

■ nmcliコマンド(RHEL7)

- ネットワークの設定や管理を行うコマンド。
- ホスト名やIPアドレスの設定、ネットワークのUp/Down、インタフェースの作成を行う。

■ ipコマンド(RHEL7)

- RHEL6の各コマンド(ifconfig、route、netstat、arp)を統合したコマンド。

■ Linuxのコマンドとの比較

1. インタフェースの作成とIPアドレスの設定

(RHE6の場合)

```
# ifconfig <interface> <addr>  
netmask <netmask>
```

(RHEL7の場合)

```
# nmcli c add type eth ifname  
<interface> con-name <string>  
# nmcli c mod <interface>  
ipv4.method manual ipv4.addresses  
"<addr>/<prefixlen> <gateway>"
```

(Solarisの場合)

```
# ipadm create-ip <interface>  
# ipadm create-addr -T static -a  
local=<addr>/<prefixlen>  
<interface>/<string>
```



2. IPアドレス情報の確認

(RHEL6の場合)

```
# ifconfig
```

(RHEL7の場合)

```
# ip address
```

(Solarisの場合)

```
# ipadm show-addr
```



- ・Linux: nmcliコマンドで設定すると、自動的に定義ファイルが更新されます。
- ・Solaris: ipadmコマンドで設定すると、自動的に定義ファイルが更新されます。

■ Solarisのネットワーク設定

実行例

■ インタフェースの状態確認

```
# dladm show-link
```

LINK	CLASS	MTU	STATE	OVER
net0	phys	1500	up	--
net1	phys	1500	up	--

STATEのステータス
up:リンクアップ
down:リンクダウン

■ ネットワークインタフェースの作成

(書式: ipadm create-ip インタフェース名)

```
# ipadm create-ip net1
```

■ IPアドレスの設定

(書式: ipadm create-addr -T static -a local=IPアドレス/ネットマスク長 インタフェース名/任意の文字列)

```
# ipadm create-addr -T static -a local=192.168.1.10/24 net1/v4
```

■ IPアドレスの確認

```
# ipadm show-addr
```

覚えておきたいネットワーク管理のコマンド

基本操作	Red Hat Enterprise Linux		Oracle Solaris
	RHEL6	RHEL7	
IPアドレスの設定	# vi /etc/sysconfig/network-scripts/ifcfg-デバイス IPADDR=IPアドレス # service network restart	# nmcli con mod デバイス ipv4.method manual ipv4.addresses "IPアドレス/マスク ゲートウェイ"	# ipadm create-addr -T static -a local=IPアドレス/マスク デバイス
DHCPの設定	# vi /etc/sysconfig/network-scripts/ifcfg-デバイス BOOTPROTO=dhcp # service network restart	# nmcli con mod デバイス ipv4.method auto	# ipadm create-addr -T dhcp デバイス
IPアドレスの確認	# ifconfig	# ip address	# ipadm show-addr
ゲートウェイの設定	# vi /etc/sysconfig/network GATEWAY=IPアドレス # service network restart	# nmcli con mod デバイス ipv4.gateway IPアドレス	# route -p add ネットワークアドレス IPアドレス
ゲートウェイ情報の確認	# netstat -rn	# ip route	# netstat -rn
ネットワークデバイスの状態表示	# ethtool eth0	# nmcli device status	# dladm show-link # dladm show-phys (物理NIC)

詳細は「LinuxユーザのためのOracle Solarisコマンド実例集」を参照。

■ SolarisのIPネットワークマルチパス (IPMP)

■ NICを冗長化しネットワークを高信頼化するSolarisの標準機能

- LinuxのBonding機能に相当

■ IPMPの特長

- 障害検出
 - NICの障害を検出し、通信経路を自動的に切り替える。
- 回復検出
 - 障害の発生したNICが回復したことを検出し、通信経路を自動的に元の状態に戻します。
- 送信負荷分散
 - 送信パケットを複数のNICに分散することで、全体のデータ送信性能を向上できます。

Oracle Solarisマニュアル

「管理：ネットワークインタフェースとネットワーク仮想化 (Oracle Solaris11用)」

http://docs.oracle.com/cd/E26924_01/html/E25834/gfkcy.html#scrolltoc

■ IPMPの設定手順

実行例

■ IPMP(検査信号ベース)の設定例

1. インタフェースの作成

```
# ipadm create-ip net0  
# ipadm create-ip net1
```

冗長構成にするインタフェース(net0、net1)を作成

2. IPMPインタフェースの設定

```
# ipadm create-ipmp ipmp0  
# ipadm add-ipmp -i net0 -i net1 ipmp0  
# ipadm create-addr -T static -a local=192.168.1.10/24 ipmp0/v4
```

ipadmコマンドでIPMPインタフェースを作成

3. 待機インタフェースの指定

```
# ipadm set-ifprop -p standby=on -m ip net1
```

運用待機構成にする設定を実行

・上記の通り、IPMPは全てipadmコマンドを使用して設定が可能です。サブコマンドを組み合わせそれぞれの設定を実行します。

5. サービス管理

■ LinuxとSolarisのサービス管理の仕組み

■ RHEL6の場合

- /etc/init.dディレクトリのサービス起動用スクリプト(rcスクリプト)を、OS起動時のランレベルに応じて順番に実行する仕組み。
- サービス間の依存関係はサービス単位で管理(起動用スクリプトによる制御)する必要がある。

■ RHEL7の場合

- systemdというサービスマネージャにより、オンデマンドのサービス起動を提供し、ランザクションの依存関係管理を改善することで、起動時間を大きく削減した。
- 重要なサービスの起動優先順位をあまり重要でないサービスのものよりも高くすることができる。

・RHEL6のサービス管理はSolaris 9以前の旧サービス管理に相当します。

■ LinuxとSolarisのサービス管理の仕組み

■ Solarisの場合

- SMF (Service Management Facility) というサービス管理機能により、依存関係が管理されているので、サービス起動時や停止時に依存するサービスを事前に起動／停止できる。
- 障害等でサービスが停止した場合は、自動的に再起動が可能 (セルフヒーリング)。
- サービス停止の原因や、影響のある他のサービスの確認が可能。
- rcスクリプトはレガシースクリプトと呼ばれ、旧サービス管理の仕組みとして互換性があります。

• SolarisのSMFはサービスの起動だけではなく、実行中のサービスの状態を常に監視しているため、障害発生時のサービス停止の原因究明や影響範囲をすぐに確認することができます。

■ LinuxとSolarisのサービスの自動起動の違い

■ RHEL6の場合

- ・サービスの起動と自動起動の設定は別々のコマンドで設定します。

実行例

# service httpd start	…サービスの起動
# service httpd stop	…サービスの停止
# chkconfig httpd on	…サービス自動起動の有効化
# chkconfig httpd off	…サービス自動起動の無効化

■ RHEL7の場合

- ・systemctlコマンド一つで管理します。
- ・サービスの起動と自動起動の設定は別々のサブコマンドで設定します。

実行例

# systemctl start postfix.service	…サービスの起動
# systemctl stop postfix.service	…サービスの停止
# systemctl enable postfix.service	…サービス自動起動の有効化
# systemctl disable postfix.service	…サービス自動起動の無効化

・RHEL7では従来のサービスに相当する処理は「.service」という拡張子がつくファイルとして管理されており、サービス関連の処理を行うときは、この名前を指定します。

■ LinuxとSolarisのサービスの自動起動の違い

■ Solarisの場合

- svcadmコマンド一つで管理します。
- 起動／停止したサービスの状態は次回OS起動時も引き継がれます。

実行例

# svcadm enable -t httpd	…サービスの起動
# svcadm disable -t httpd	…サービスの停止
# svcadm enable httpd	…サービスの起動と自動起動の有効化
# svcadm disable httpd	…サービスの停止と自動起動の無効化

- Solarisではサービスの起動／停止の状態が、次回のOS起動時にも引き継がれます。
- 次回OS起動時にこの状態を引き継ぎたくない場合は、-tオプションを使用し、一時的にサービスを起動／停止します。

■ 覚えておきたいサービス管理のコマンド

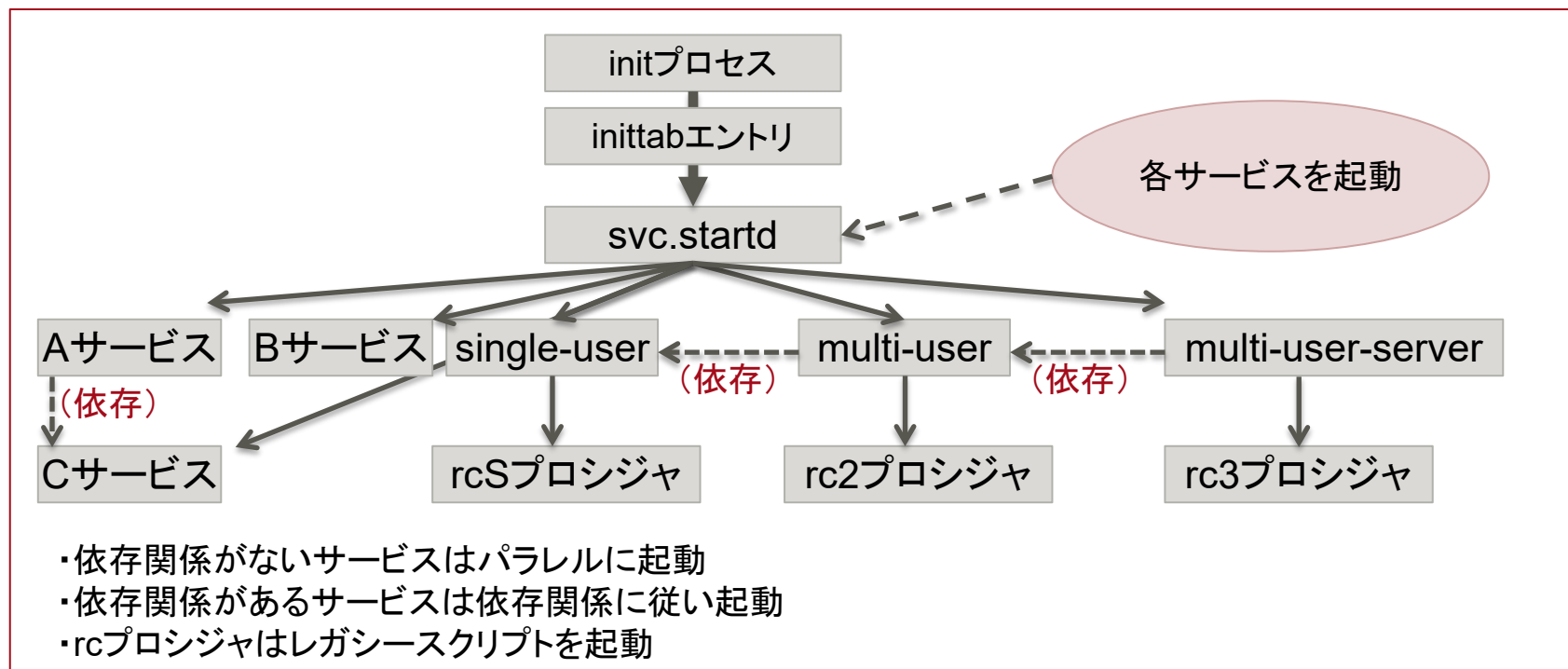
基本操作	Red Hat Enterprise Linux		Oracle Solaris
	RHEL6	RHEL7	
サービスの起動	service サービス名 start	systemctl start サービス名	svcadm enable -t サービス名 (FMRI)
サービスの起動と自動起動の有効化	service サービス名 start chkconfig サービス名 on	systemctl start サービス名 systemctl enable サービス名	svcadm enable サービス名 (FMRI)
サービスの停止	service サービス名 stop	systemctl stop サービス名	svcadm disable -t サービス名 (FMRI)
サービスの停止と自動起動の無効化	service サービス名 stop chkconfig サービス名 off	systemctl stop サービス名 systemctl disable サービス名	svcadm disable サービス名 (FMRI)
サービスの一覧表示	chkconfig --list	systemctl list-unit-files --type service	svcs -a
サービスの状態表示	service サービス名 status	systemctl status サービス名	svcs サービス名 (FMRI)

※FMRI(Fault Managed Resource Identifier)

詳細は「LinuxユーザのためのOracle Solarisコマンド実例集」を参照。

■ SMF (Service Management Facility)

- SMFで管理するサービスの起動スクリプトは、`/lib/svc/method`ディレクトリにあり、`svc.startd` (マスターリスタータデーモン) により制御されます。
- サービス間の依存関係を定義したマニフェストファイルは、グループ別に `/var/svc/manifest` ディレクトリに存在し、`svc.configd` (リポジトリデーモン) によって参照／変更されます。
- SMFによる管理のログは `/var/svc/log` 配下にサービス毎に出力されます。



6. ファイルシステムとストレージ管理

	Red Hat Enterprise Linux		Oracle Solaris
	RHEL6	RHEL7	
ストレージ (ボリューム)管理	<ul style="list-style-type: none"> LVM(Logical Volume Manager) 		<ul style="list-style-type: none"> ZFS(Zetta byte File System) zpool コマンド
ファイルシステム	<ul style="list-style-type: none"> ext3 (推奨) ,ext4 	<ul style="list-style-type: none"> XFS 	<ul style="list-style-type: none"> ZFS zfs コマンド
最大ファイル システムサイズ	<ul style="list-style-type: none"> 16TB 	<ul style="list-style-type: none"> 500TB 	<ul style="list-style-type: none"> 25京6000兆ZB
最大ファイルサイズ	<ul style="list-style-type: none"> 2TB (ext3) ,16TB (ext4) 	<ul style="list-style-type: none"> 500TB 	<ul style="list-style-type: none"> 25京6000兆ZB
冗長構成	<ul style="list-style-type: none"> 基本的にサーバ本体や外付けストレージ装置のハードRAIDを使用 <p>※富士通のPRIMERGYシリーズでは、RAIDカードを標準搭載しており、SVOM(ServerView Operations Manager)を使用してシステムを監視します。</p>		<ul style="list-style-type: none"> ZFSによるソフトRAIDや外付けストレージ装置のハードRAIDを使用 <p>※富士通のSPARC M10でも、ハードRAIDを標準機能としてサポートしています。</p>

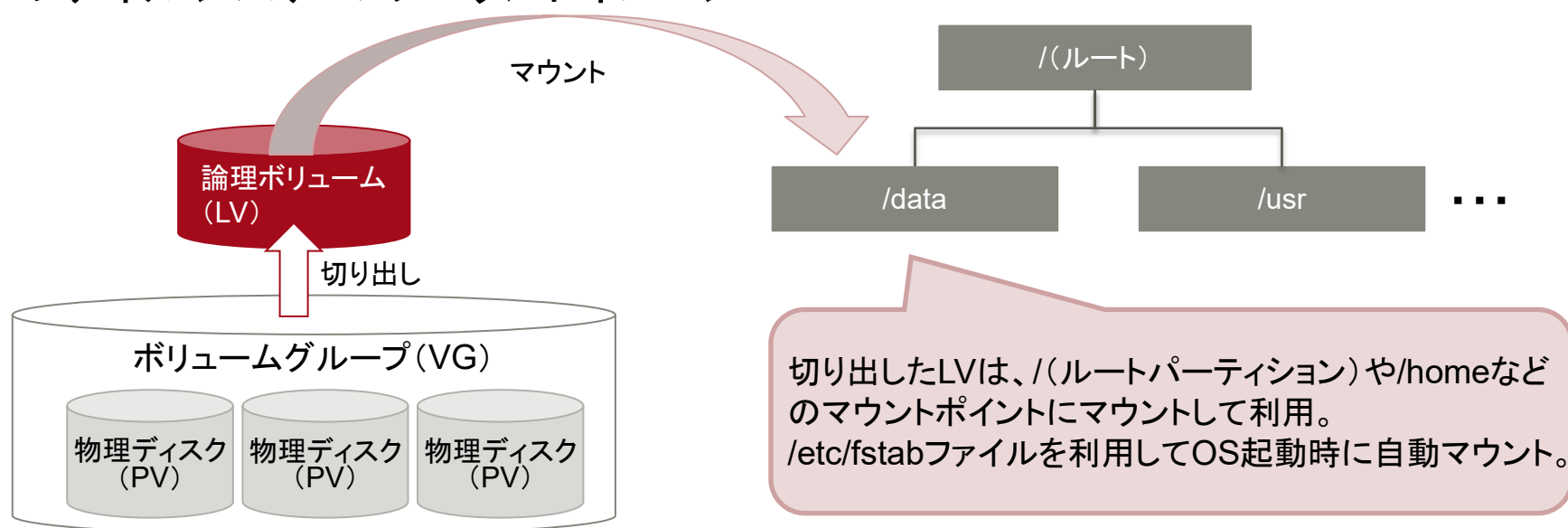
- Solarisのシステムボリューム領域はZFSで構成されます。
- ユーザボリューム領域でのみ Solaris 10以前のUFS(UNIX File System)も使用可能です。

■ LVM (Logical Volume Manager)

■ 特長

- 複数の物理ディスク(PV)を1つのボリュームグループ(VG)にまとめます。
- ボリュームグループ(VG)から論理ボリューム(LV)を切り出します。
OS稼働中に論理ボリューム(LV)の追加や拡張が可能です。
- 論理ボリューム(LV)に対してファイルシステムを作成後、マウントします。
- OS稼働中にスナップショット機能によりバックアップの取得が可能です。

■ ファイルシステムのマウントイメージ

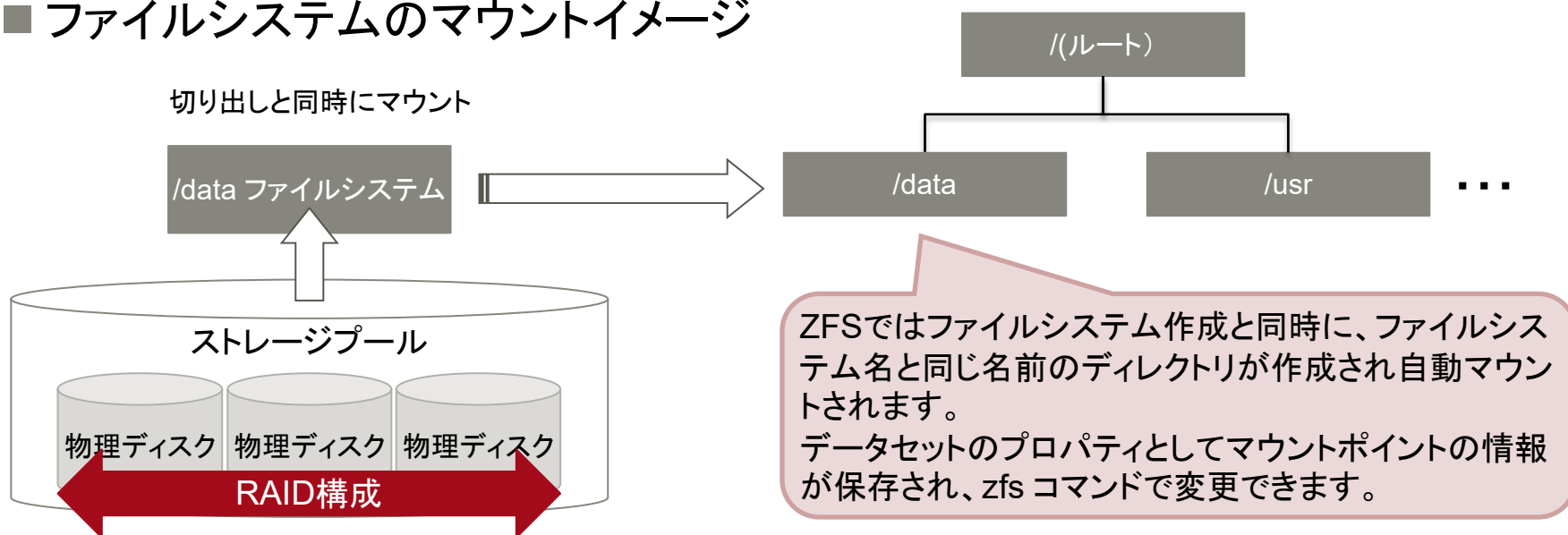


■ ZFS (Zetta byte File System)

■ 特長

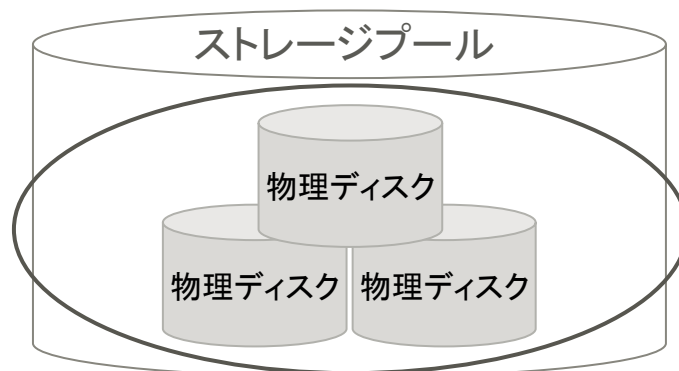
- 複数の物理ディスクから1つのストレージプールを構成します。ストレージプールをRAID構成にすることが可能です。
- ストレージプールからデータセットとして領域を切り出します。
- データセットは切り出しと同時にZFS形式のファイルシステムとしてマウントされます。ファイルシステムはストレージプールの容量の範囲で自動的に拡張されます。
- OS稼働中にスナップショット機能によりバックアップの取得が可能です。

■ ファイルシステムのマウントイメージ



■ 標準機能でディスクの冗長化が可能

・ZFSでサポートするRAID構成



ストレージプールにディスクを登録
するときに"RAID構成"を指定
※RAID1+0も構成可能

非冗長

◆ストライピング(RAID-0)

mirror(ミラー)

◆複数面ミラーも可能(RAID-1)

RAID-Z

◆シングルパリティ(RAID-5に類似)

RAID-Z2

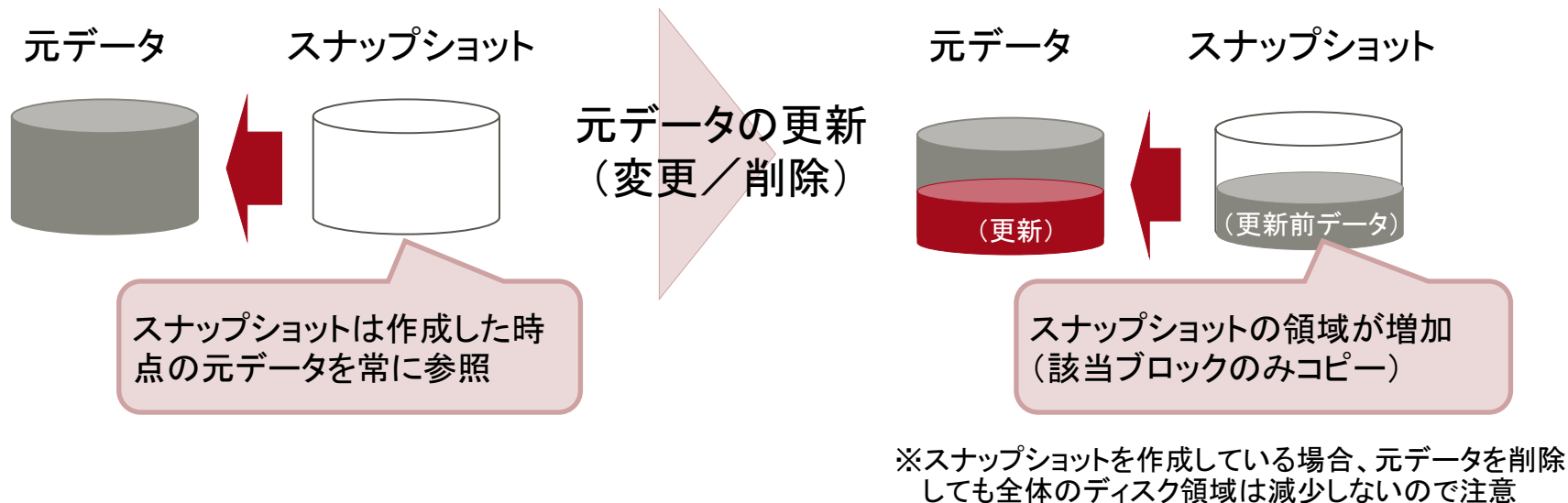
◆ダブルパリティ(RAID-6に類似)

RAID-Z3

◆トリプルパリティ

・SolarisのストレージプールはRAIDを構成して物理ディスクを登録することで、ファイルシステムの性能や信頼性を向上させることができます。

■ ZFSスナップショットの仕組み



■ ZFSスナップショットの特長

- 空き領域を自動で使用するため、スナップショットの領域を指定する必要があるない。
- データ更新時は該当ブロックのみコピーするのでディスク消費量が少ない。
- スナップショット作成時点に戻すことができる。【ロールバック機能】
- ZFSスナップショットを元にファイルシステムの退避ができる。【バックアップ機能】
- スナップショットを利用して、ファイルシステムの複製ができる。【クローン機能】

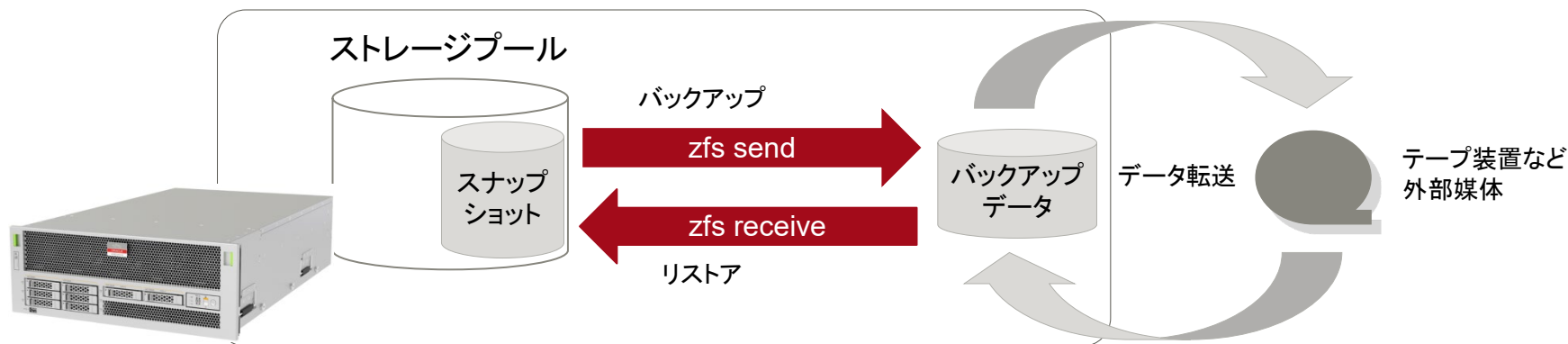
■ ZFSコマンドでバックアップ／リストアを実行

■ バックアップ: zfs send

- zfs send コマンドにより、スナップショットをバックアップデータとして標準出力に書き込みます。
- OS停止は必要ありません。

■ リストア: zfs receive

- バックアップデータを標準入力から読み込んで、ファイルシステムを復元します。
- リストア対象がシステムボリューム（ルートプール）ではない場合、OS停止の必要はありません。



・ZFSについて、詳しくはOracle Solaris 11 ZFSを使ってみよう(概要・設計ガイド)をご確認ください

<http://www.fujitsu.com/jp/sparc-technical/document/#solaris11>

■ 覚えておきたいファイルシステムとストレージ管理のコマンド

基本操作	Red Hat Enterprise Linux		Oracle Solaris
	RHEL6	RHEL7	
Linux: ボリュームグループ作成 Solaris: ストレージプール作成	vgcreate ボリュームグループ名 デバイス名		zpool create プール名 RAID デバイス名
Linux: 論理ボリューム作成 Solaris: ZFSファイルシステム作成	lvcreate -L サイズ -n 論理ボリューム名 ボリュームグループ名		zfs create ファイルシステム名
スナップショットの作成	lvcreate -s -L サイズ -n スナップショット名 元のデバイス		zfs snapshot スナップショット名
スナップショットからのロールバック	lvconvert --merge スナップショット名		zfs rollback スナップショット名
ファイルシステムのバックアップ	dump -0u -f バックアップファイル パーティション	xfsdump -l レベル -f バックアップファイル パーティション	zfs send スナップショット名
ファイルシステムのリストア	restore -r -f バックアップファイル	xfsrestore -f バックアップファイル -s セッションID 復元先	zfs receive スナップショットストリーム
パーティション操作	parted（推奨）,fdisk		format

詳細は「LinuxユーザのためのOracle Solarisコマンド実例集」を参照。

■ Solarisのディスクラベル

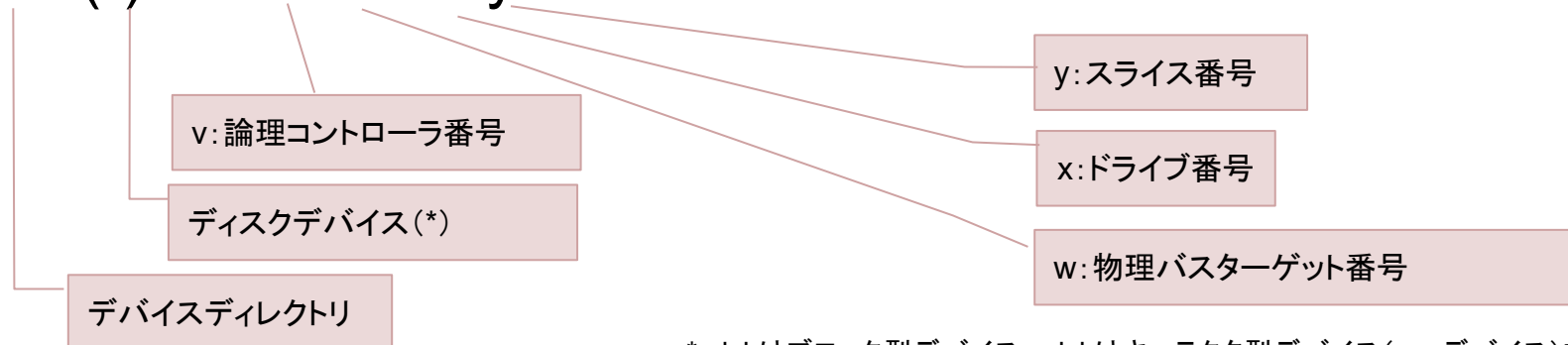
■ Solarisでは、次の2つのディスクラベルをサポートしています。

- SMI (Sun Microsystems Inc.)
 - システムボリューム用のディスクラベル。サイズが2TBに満たないディスク用です。
- EFI (GPT) (Extensible Firmware Interface GUID Partition Table)
 - システムボリューム以外のZFS用のディスクラベルです。

■ Solarisのデバイスパス

■ `/dev/(r)dsk` ディレクトリ配下にある、特定のコントローラ、ディスク、およびスライスを識別する文字列へのパスになります。

`/dev/(r)dsk/cvtdwxsy`



*: dskはブロック型デバイス、rdskはキャラクタ型デバイス(rawデバイス)を表します。

■ LinuxとSolarisのパーティションの違い

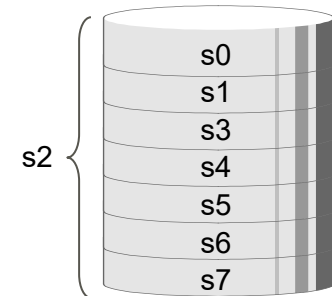
■ Linuxの場合

- 1つのディスクを複数の領域(パーティション)に分け、ファイルシステムやrawデバイスとして使用します。
- パーティションの設定はfdiskコマンドを使用します。

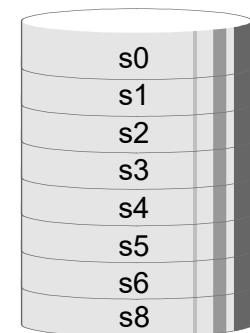
■ Solarisの場合

- Solarisのパーティションは「スライス」とも呼ばれます。
- スライスの作成数はディスクラベルによって異なります。
- デバイス名の後にs0～s7を付けてデバイスパスを表現します。(例:/dev/rdisk/c2t0d1s0)
- SMIラベルのs2はディスク全体を表す特殊なスライスです。
- スライスの設定はformatコマンドを使用します。

SMIラベルのディスク



EFIラベルのディスク



■ LinuxとSolarisのISOイメージファイルのファイル形式

■ Linuxの場合

- ファイルシステム形式として iso9660 を指定してマウントします。

実行例

/ISO/media.iso を/mnt ディレクトリにマウントする

```
# mount -o loop -t iso9660 /ISO/media.iso /mnt
```

 ...ISOファイルを/mntにマウントする

■ Solarisの場合

- ファイルシステム形式として hsfs を指定してマウントします。

実行例

/ISO/media.iso を/mnt ディレクトリにマウントする

```
# mount -F hsfs /ISO/media.iso /mnt
```

 ... ISOファイルを/mntにマウントする

■ Solarisのファイルシステムの構成

- ZFS以外のファイルシステムは、OS起動時に/etc/vfstabというファイルを参照して自動的にマウントされます。

■ 主なディレクトリ

名称	概要
/dev	特殊デバイスファイルを含むディレクトリ
/devices	devfsファイルシステムのマウントポイントディレクトリ
/etc	システム固有の管理およびコンフィギュレーションファイルを含むディレクトリ
/bin	一般ユーザも使用できる実行可能ファイルを含むディレクトリ
/sbin	ブートプロセスや手動でシステムのリカバリで使用する実行可能ファイルを含むディレクトリ
/usr/lib	実行時にプログラムによって必要とされる追加のシステムライブラリ
/proc	プロセスファイルシステムのマウントポイントディレクトリ
/tmp	一時ファイル用のディレクトリ。メモリ上のファイルシステムである tmpfsを使用

- swapとdump デバイスは、ZFSボリュームで構成されています。
- /tmp に配置したファイルは物理メモリを消費します。また、再起動により消失します。

7. 監視

■ LinuxとSolarisのログ監視

■ Linuxの場合

- システムメッセージのログ出力は `/etc/rsyslog.conf` で定義されています。

■ Solarisの場合

- システムメッセージのログ出力は `/etc/syslog.conf` で定義されています。
- Linuxと同様にシステムメッセージをログインユーザや管理者に表示したり、他のサーバに転送することが可能です。
- Linuxと同様にシステムのパフォーマンス情報(CPU、メモリ使用量など)はOSコマンドで確認することが可能です。
- Solaris独自の監視用コマンドもあります。

- ・オンサイトでの監視の場合は専用のソフトウェアを導入して、運用監視のシステムを導入する方法が一般的です。
- ・リモート通報(REMCS)を利用し、富士通サポートセンタによるリモート監視を導入するシステムも多くあります。



■ LinuxとSolarisが出力するログの違い

■ ログの出力先

ログ	Red Hat Enterprise Linux	Oracle Solaris
システムが出力するログ	/var/log/messages	/var/adm/messages
メール送受信を記録するログ	/var/log/maillog	/var/log/syslog
cron実行時のログ	/var/log/cron	/var/cron/log

■ ログのローテーション

- Linuxと同様、Solarisもログファイル単位に項目を設定しcron により定期的に行われます。

Red Hat Enterprise Linux	Oracle Solaris
<ul style="list-style-type: none"> ・/etc/logrotate.conf ファイルで定義します。 weekly ...1週間単位 rotate 4 ...4世代 compress ...圧縮する /var/log/messages { sharedscripts postrotate /bin/kill -HUP `cat /var/run/syslogd.pid 2> /dev/null` 2> /dev/null true endscript } 	<ul style="list-style-type: none"> ・/etc/logadm.confファイルにローテーションするログファイルごとに設定を定義します。 /var/adm/messages -C 4 -a '/usr/sbin/svccfg -s svc:/system/system-log refresh' -C : 保存する世代数 -a : ログファイル名の変更後に"内のコマンドを実行

■ 搭載リソースを確認するコマンドの例

確認項目	Red Hat Enterprise Linux	Oracle Solaris
CPU、メモリ、増設カードの確認	cat /proc/cpuinfo cat /proc/meminfo lspci	prtdiag

■ 異常を確認するコマンドの例

確認項目	Red Hat Enterprise Linux	Oracle Solaris
ネットワークの状態	ethtool、ping	dladm、ping
サービスの状態	service(RHEL6) systemctl(RHEL7)	svcs
プロセスの状態	pstree、ps	ptree、ps
プロセスのトレース	strace	truss

■ リソース使用量を確認するコマンドの例

確認項目	Red Hat Enterprise Linux	Oracle Solaris
システム稼働時間	uptime	uptime
CPU、メモリ、I/Oの負荷	mpstat、vmstat iostat、iotop、netstat	mpstat、pgstat、vmstat、iostat、 fsstat、netstat、flowstat、dlstat、 ipmpstat
プロセス単位の負荷	top、pidstat	top、prstat

■ 覚えておきたい監視のコマンド

基本操作	Red Hat Enterprise Linux		Oracle Solaris
	RHEL6	RHEL7	
CPU、メモリ、I/Oの負荷を確認する	mpstat vmstat iostat		mpstat vmstat iostat
CPU、メモリ、増設カードを確認する	cat /proc/cpuinfo cat /proc/meminfo lspci		prtdiag
ネットワークの疎通を確認する	ping		ping
プロセスの状態を確認する	ps -ef		ps -ef

詳細は「LinuxユーザのためのOracle Solarisコマンド実例集」を参照。

・プロセスの状態を監視するtopコマンドなどは、Solarisでも同様に使用することができます。

■ cronについて

■ cronの編集

- cronの編集はcrontabコマンドで行います。

```
# crontab -e
10 3 * * * /usr/sbin/logadm      . . .   書式
↑ ↑ ↑ ↑ ↑                      ↑
分 時 日 月 曜日      コマンド
```

■ 登録内容の表示

- cronに登録されている内容の表示はcrontabコマンドで行います。

```
# crontab -l
10 3 * * * /usr/sbin/logadm
```

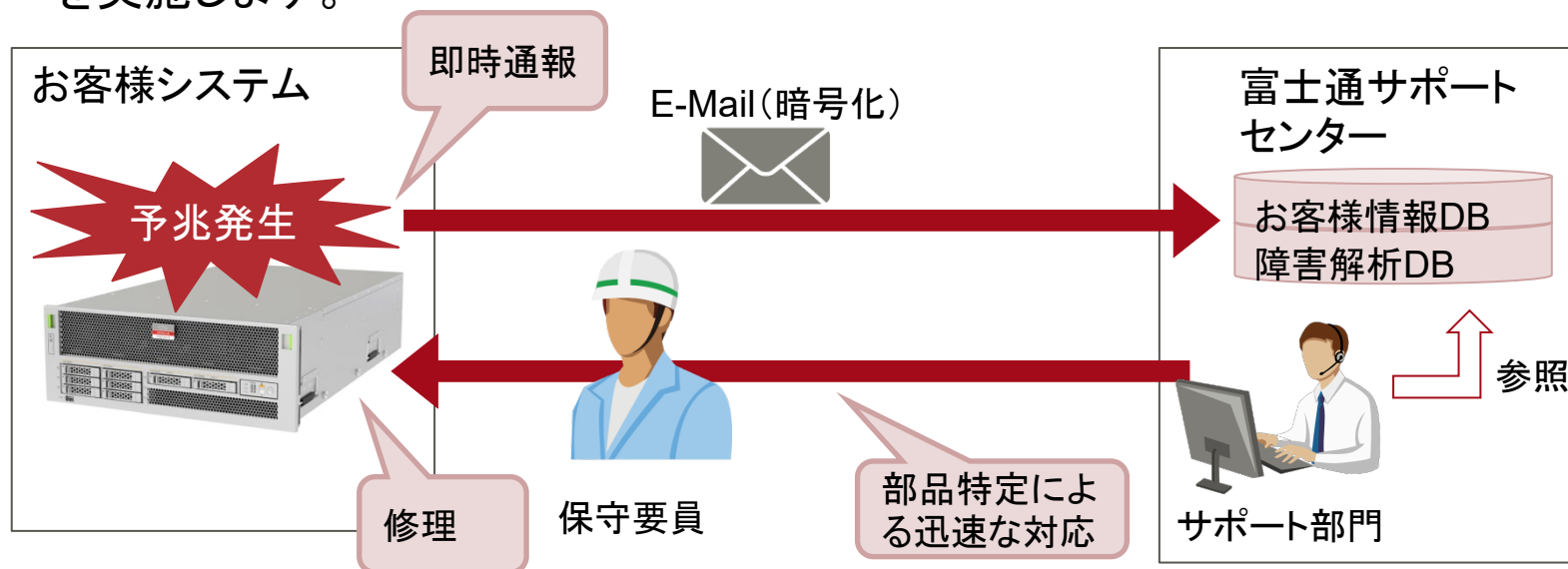
■ 実行結果の通知

- cronにより実行された結果は、登録したユーザ宛にメールで送信されます。

■ 富士通のリモート通報機能 (REMCS)

■ 特長

- ・ハードウェアの障害予兆情報を検知し、富士通サポートセンターに自動通報します。
- ・システム管理者に代わり専門スタッフが適切に対応し、トラブルを未然防止します。
- ・万一のトラブル発生時には、自動的に通報されたハードウェアの異常情報をもとに、センターの専門スタッフがトラブル箇所の特定、部品の手配、保守要員の派遣までを実施します。



・LinuxもSolarisも富士通のサポート体制に変わりはありませんが、REMCSによる通報の可否は対象装置の障害発生箇所により異なるため注意が必要です。

8. 仮想環境

■ Docker

- コンテナ型の仮想化ソフトウェアで、OS上にアプリケーション動作の隔離空間(コンテナ)を作成します。
- サーバ仮想化による性能面の劣化が小さいことや、ハードウェア資源の消費が少ないという特徴があります。
- コンテナ環境をイメージとして保存して、別のサーバのDockerに持ち込んでそのままの環境を動作させることができます。

■ KVM

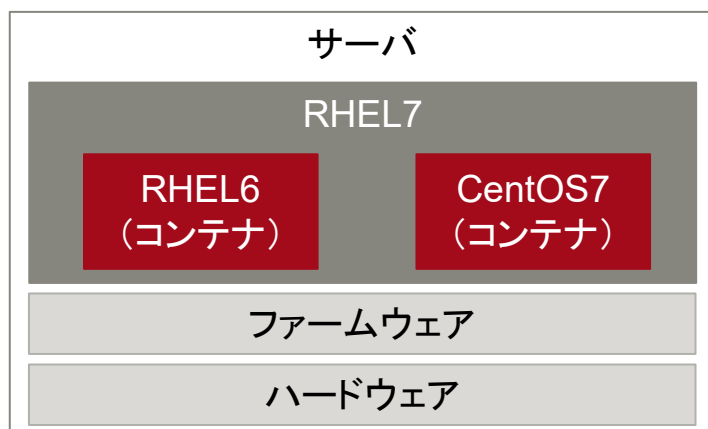
- ハイパーバイザー型の仮想化ソフトウェアで、ゲストOSという仮想環境を作成します。
- CPUの仮想化支援機能を利用し、ハードウェア自体を仮想化するため、Linuxに限らず様々なOS(Windowsなど)をゲストOSとすることができます。
- 独自のハイパーバイザーを持たず、Linux自体をホストOSとしてゲストOSの制御を行います。

■ Oracle Solaris ゾーン

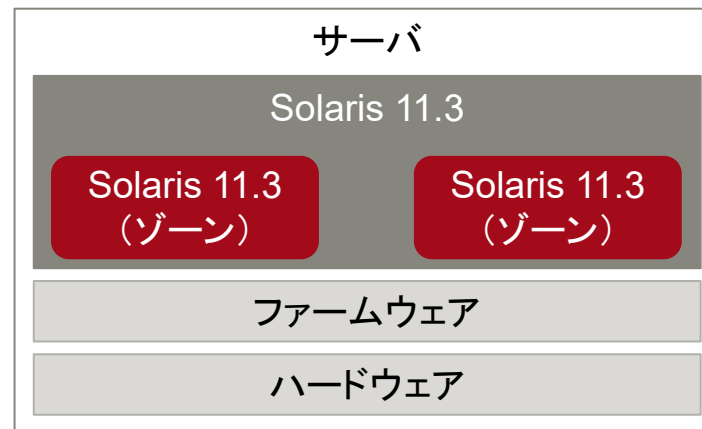
■ LinuxのDockerに相当するサーバ仮想化機能です。

- Solarisカーネル層でリソースを分割・管理します。
- コンテナ型の仮想化を実現します。
- 全てのゾーンは、共通のカーネルを使用するため、一括で各ゾーンにパッチや修正パッケージの適用ができます。

Docker



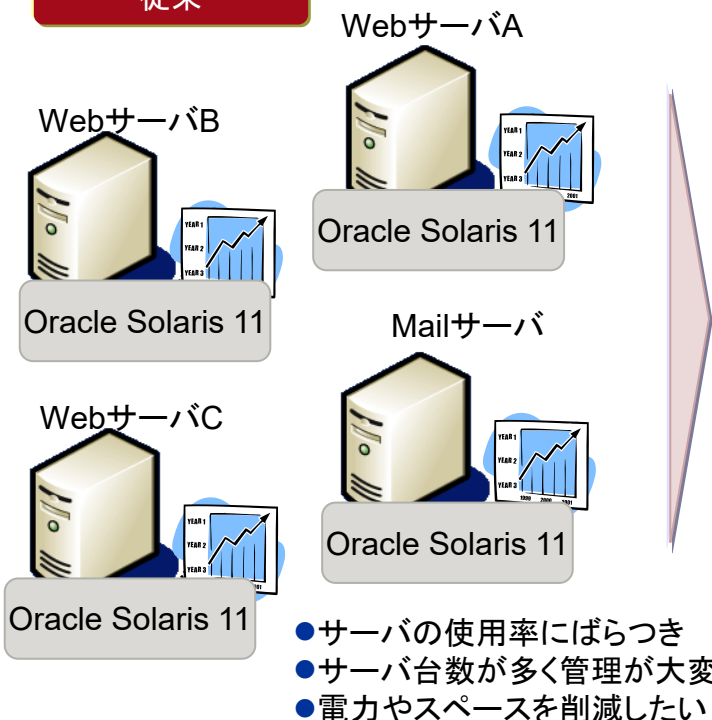
Oracle Solaris ゾーン



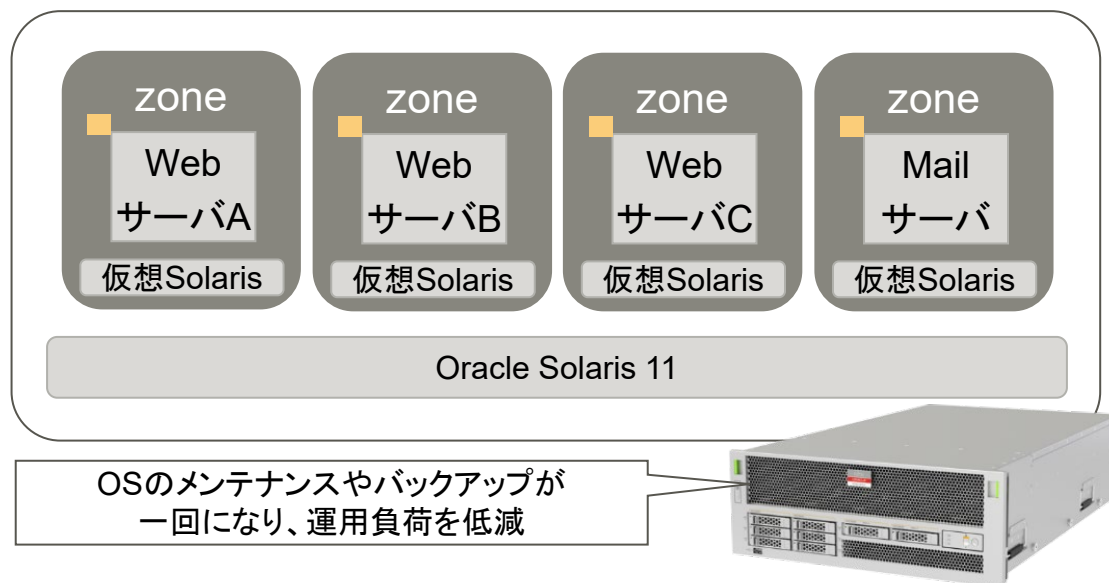
■ Solaris ゾーンの特長

- ハードウェア構成に依存せず、最大8191個の仮想OSを構築可能
- 仮想OSの追加・削除は簡単な作業で短時間に行うことが可能
- 仮想OS毎のOSインストールは不要
- CPU、メモリなどのハードウェアリソースを柔軟に配分可能
- 無償で使用可能

従来



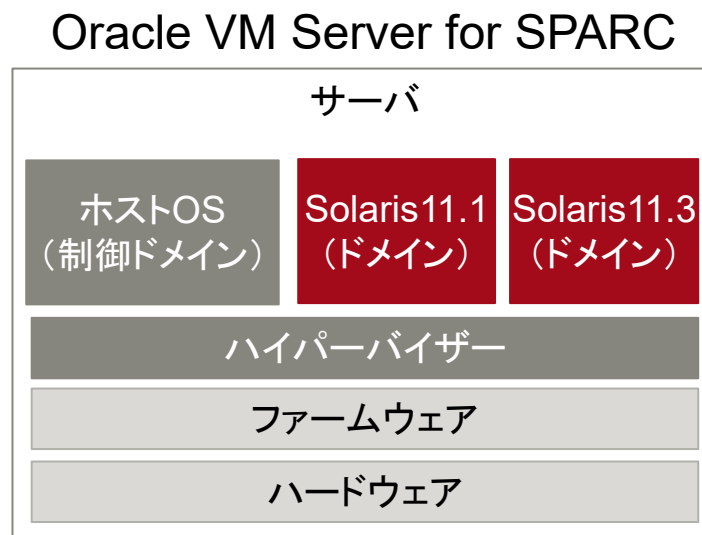
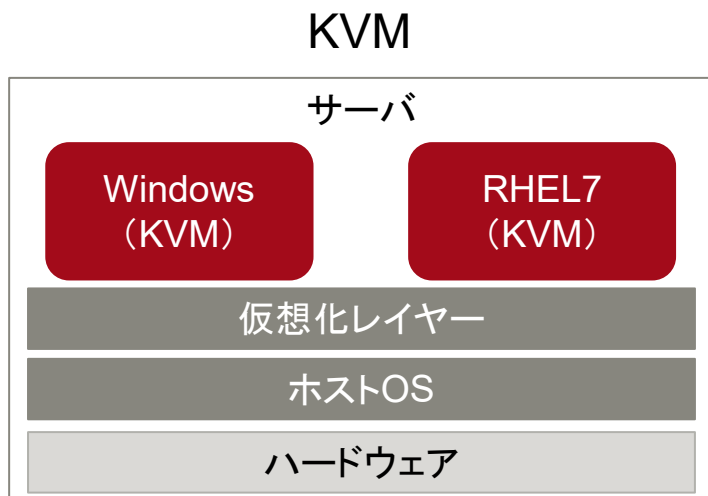
Oracle Solaris ゾーンで集約



■ Oracle VM Server for SPARC (Oracle VM)

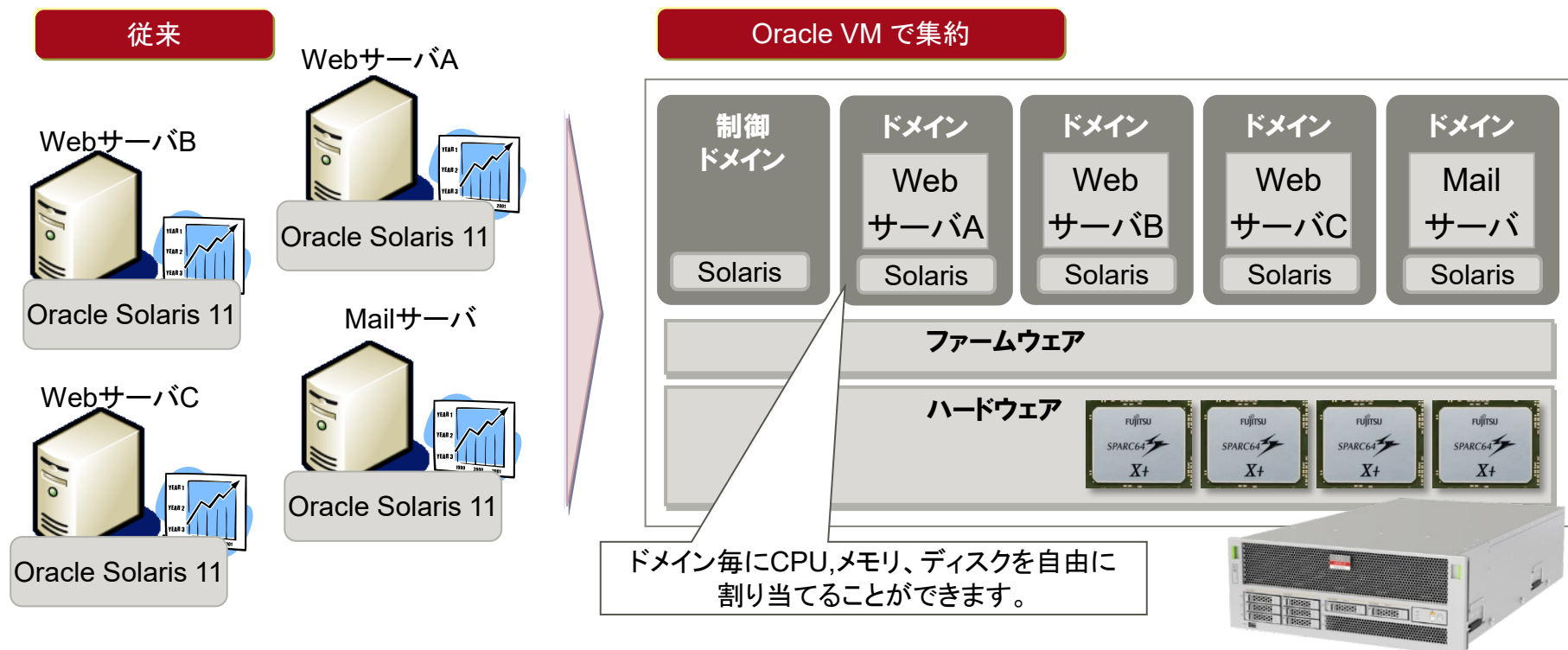
■ LinuxのKVMに相当する機能です。

- ファームウェア層でリソースを分割・管理します。
- ハイパーバイザー型の仮想化を実現します。
- ドメイン毎に別々のカーネルを使用するため、Solarisのアップデートリリースやパッチレベルの異なる環境を構築することが可能です。



■ Oracle VM Server for SPARC の特長

- 異なる版数・アップデートリリースのOS環境を統合可能
- ドメイン(仮想サーバ)ごとの柔軟なリソース(CPU, メモリ, I/O)の割り当てが可能
- ドメイン間のソフトウェアの障害隔離性を確立
- 無償で使用可能



■ Docker と Oracle Solaris ゾーン

	Docker	Oracle Solaris ゾーン
仮想環境の動作OS	様々なLinuxディストリビューション (RHEL, Cent OSなど)	Solaris 10, Solaris 11 ※ホストOSがSolaris 11の場合
性能オーバーヘッド	極めて小さい	極めて小さい
仮想環境への接続	専用のDockerクライアントが必要	専用クライアント不要 (telnet, ssh等で接続可能)

■ KVM と Oracle VM Server for SPARC

	KVM	Oracle VM Server for SPARC
仮想環境の動作OS	Windows, Linux, IA Solaris など 様々なOSを選択可能	Solaris 10, Solaris 11 ※ホストOSがSolaris 11の場合
性能オーバーヘッド	比較的大きい	極めて小さい

■ Docker : コンテナの管理

■ コンテナの作成／終了

実行例

# docker pull registry.access.redhat.com/rhel	…イメージのダウンロード
# docker run -it -h コンテナ名 rhel /bin/bash	…コンテナ名でコンテナを起動
[コンテナ名]#	…コンテナにログイン中
[コンテナ名]# exit	…コンテナの終了
# docker rm コンテナ名	…コンテナの削除

■ Solaris ゾーン : ゾーンの管理

■ ゾーンの作成／終了

実行例

# zonecfg -z ゾーン名 create	…ゾーンの設定ファイル作成
# zoneadm -z ゾーン名 install	…ゾーンのインストール
# zoneadm -z ゾーン名 boot	…ゾーンの起動
# zlogin -C ゾーン名	…ゾーンのコンソールにログイン
[ゾーン名]# ~.	…ゾーンからログアウト
# zoneadm -z ゾーン名 shutdown	…ゾーンの停止
# zoneadm -z ゾーン名 delete	…ゾーンの削除

■ KVM : ゲストOSの管理

■ ゲストOSの作成／終了

実行例

```
# virt-install -各種パラメータ
# virsh start ゲスト名
# virsh console ゲスト名
[ゲスト名]# Ctrlキーと]キーを押下
# virsh shutdown ゲスト名
```

- …ゲストの設定ファイル作成&インストール
- …ゲスト名でゲストを起動
- …ゲストにログイン
- …ゲストからログアウト
- …ゲストの終了

■ Oracle VM : ゲストドメインの管理

■ ゲストドメインの作成／終了

実行例

```
# ldm add-domain ドメイン名
# ldm set-core 2ドメイン名
# ldm set-memory 4096Mドメイン名
# ldm bindドメイン名
# ldm startドメイン名
# ldm stopドメイン名
# ldm unbindドメイン名
```

- …ドメイン名でドメインを作成
- …使用するコアを指定
- …使用するメモリを指定
- …ドメインをバインド
- …ドメインを起動
- …ドメインを停止
- …ドメインをアンバインド

■ Oracle Solaris 11の構築に役立つドキュメントが満載

- ハイパーバイザーベースの仮想化:
Oracle VM Server for SPARC
- Solarisベースの仮想化:
Oracle Solaris ゾーン
- 最新ファイルシステム:
ZFS (Zetta-byte File System)
- Solaris 8/9環境をそのままSolaris 10へ:
Oracle Solaris Legacy Containers

など

今すぐクリック!!



<http://www.fujitsu.com/jp/sparc-technical/>

■ Oracle社マニュアル

Oracle Solaris 11.3 Information Library

http://docs.oracle.com/cd/E62101_01/

⇒管理者、開発者向けのマニュアル全般とコマンドリファレンスが提供されています。

■ サーバマニュアル

「SPARC M10 システム システム運用・管理ガイド」

<http://www.fujitsu.com/jp/products/computing/servers/unix/sparc/lineup/m10-1/documents/>

⇒システムの運用管理に必要な基本操作を解説しています。

■ 技術情報 Technical Park

「Oracle Solaris コマンド集」

<http://www.fujitsu.com/jp/sparc-technical/command-reference/>

⇒使用目的からコマンドを探すことができます。

「UNIXサーバ SPARC M10 ドキュメント」

<http://www.fujitsu.com/jp/sparc-technical/document/>

⇒各種機能の紹介や手順書を記載しています。

■ Linux関連情報

「Product Documentation for Red Hat Enterprise Linux」

https://access.redhat.com/documentation/ja-jp/red_hat_enterprise_linux/

版数	更新日時	更新内容
第1.0版	2014年8月	新規作成
第1.1版	2016年3月	RHEL7、Oracle Solaris 11.3に対応

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

- 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。

