

Oracle Solaris Clusterを 使ってみよう（概要・構築ガイド）

2017年 9月（第2.0版）
富士通株式会社

■ 目的

- ・ 本書は、SPARC / Solaris サーバ用のクラスタソフトウェアである、Oracle Solaris Cluster の概要や機能、構築に関する情報を記載します。

■ 対象読者

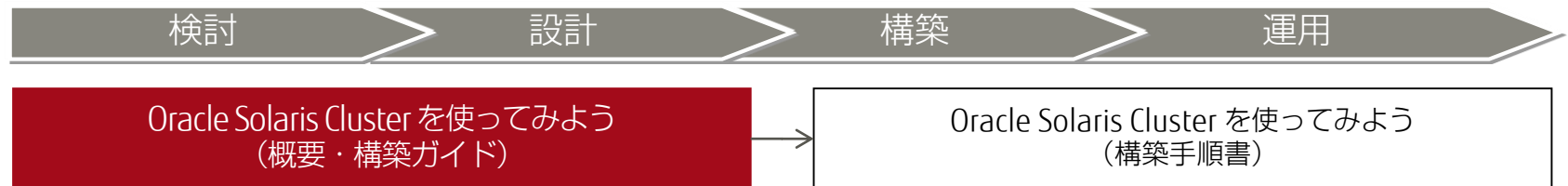
- ・ Oracle Solaris Cluster の導入を検討されている方
- ・ Oracle Solaris の基礎知識を有している方

■ 留意事項

- ・ 本書は、Oracle Solaris Cluster 4.3 / Oracle Solaris 11.3に基づいています。
 - これらのバージョンが異なる場合、一部の機能が使用できなかったり、構築手順が異なる場合があります。
- ・ Oracle Solaris Cluster の最新情報は、Oracle 社のマニュアルも併せてご確認ください。
 - Oracle Solaris Cluster 4.3 Documentation
http://docs.oracle.com/cd/E62281_01/

■ ドキュメントの位置づけ

- Oracle Solaris Clusterを使ってみよう
http://www.fujitsu.com/jp/sparc-technical/document/middleware/index.html#solaris_cluster



■ 本書での表記

- ・ コマンドのセクション番号は省略しています。
例：
 - ls(1) ⇒ lsコマンド
 - shutdown(1M) ⇒ shutdownコマンド
- ・ 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
Solaris Cluster	Oracle Solaris Cluster
Solarisゾーン	Oracle Solarisゾーン
Oracle VM	Oracle VM Server for SPARC
Oracle DB	Oracle Database
HA for Oracle DB	HA for Oracle Database

- Oracle Solaris Clusterの概要
- Oracle Solaris Clusterの特長
- Oracle Solaris Clusterを用いたクラスタシステムの構築
 1. サーバの環境確認・OSの設定
 2. ディスクの冗長化設定
 3. Oracle Solaris Clusterのインストール
 4. クラスタへのノードの登録
 5. クラスタの初期設定
 6. リソースの設定
 7. アプリケーション／サービスのリソース登録
- 参考情報
 - Oracle Solaris Cluster Managerの導入
- 付録

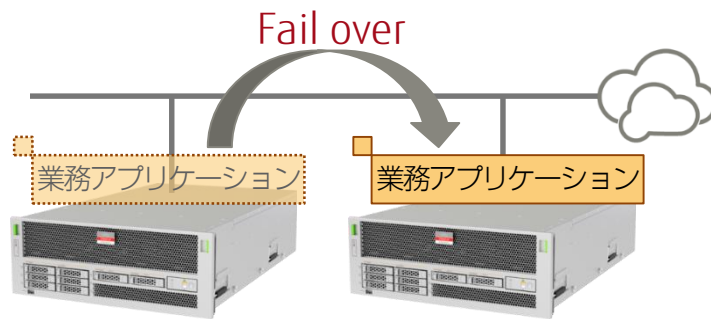
Oracle Solaris Clusterの概要

Oracle Solaris Clusterの概要、対応するクラスタ構成、Solaris標準機能との連携について説明します。

Oracle Solaris Clusterとは

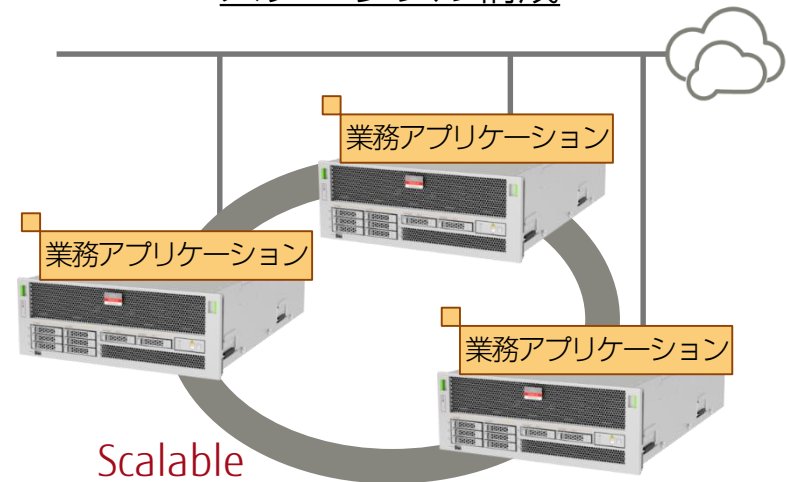
- Solaris標準の高可用性サーバクラスタソフトウェアです。
 - ・複数のサーバでクラスタシステムを構築し、可用性を高め、サーバで障害が発生したときの業務稼動に対応できます。
- 2種類のクラスタ構成に対応しています。
 - ・稼動系サーバと待機系サーバからなる基本的なノード（サーバ）切替型構成（フェイルオーバー構成）
 - ・複数のサーバを同時に稼動させる負荷分散型構成（スケーラブル構成）

フェイルオーバー構成



1台のサーバで業務システムを稼動させておき、サーバの障害発生時には、別のサーバに業務システムを引き継ぎます。

スケーラブル構成



複数のサーバで業務システムを動作させて、業務負荷を各サーバに分散させます。

Solaris標準機能との連携

Solaris Clusterは常に様々なSolarisの標準機能と連動して動作します。
そして、ハードウェア・ソフトウェアの障害の迅速な検知・復旧が可能です。

アプリケーション

Oracle Solaris Cluster Data Service

アプリケーションの状態を監視し、障害に応じた復旧処理を自動実行

OS

SMF (※1)

各種サービス動作（起動・停止・再起動）・サービス間の依存関係の管理

ハードウェア

FMA (※2)

ハードウェア部品の故障管理・故障部品の自動オフライン

ZFS (※3)

複数のディスクの一元管理（プール化）・ディスク冗長化・データ保護

MPxIO

主にFC（ファイバチャネル）などのI/Oコントローラ・パスの冗長化

IPMP (※4)

ネットワークインターフェースの冗長化・フェイルオーバー・負荷分散



- ※1 SMF : Service Management Facility
- ※2 FMA : Fault Management Architecture
- ※3 ZFS : Zettabyte File System
- ※4 IPMP : IP Networking Multi Pathing

Oracle Solaris Clusterの特長

Oracle Solaris Clusterの特長について説明します。

■ クラスタ環境の柔軟性

- ・ クラスタの構成・サーバの機種などを自由に選択できます。

■ リーズナブルなクラスタシステムの確立

- ・ Solaris標準機能の使用や、仮想サーバ環境への対応により、安価なクラスタシステムを構築できます。

■ 容易な構築・管理インターフェース

- ・ クラスタシステムの構築や管理を支援する環境が充実しています。

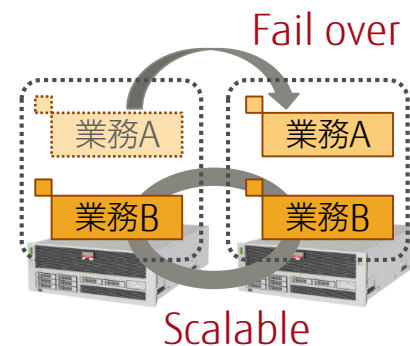
■ ZFSファイルシステムによるデータ保護

- ・ Solaris Clusterで構築するクラスタシステムのディスク管理に使用するZFSファイルシステムには、信頼性・可用性を高める様々な仕組みが備わっています。

クラスタの構成・サーバの機種などを自由に選択できます。

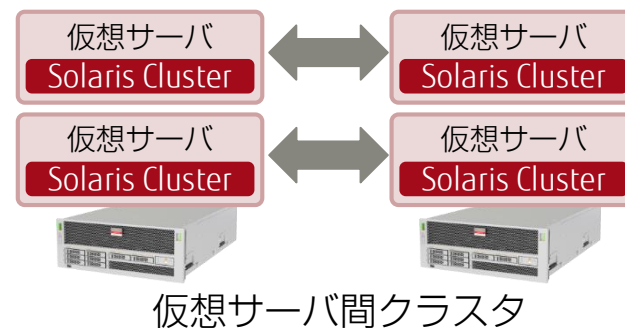
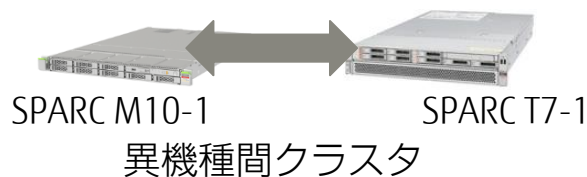
■ クラスタ構成の使い分け

- ・ Solaris Clusterが対応するクラスタ構成は、「フェイルオーバー構成」と「スケーラブル構成」の2つです。
- ・ これらの2つの構成を業務（アプリケーション）ごとに使い分けることもできます。



■ 様々なプラットフォームによる構成

- ・ 異なるサーバ（機種）間でもクラスタシステムを構築できます。
- ・ 仮想サーバに対しても導入できます。
 - Oracle VM Server for SPARCとSolarisゾーンの両方に対応しています。



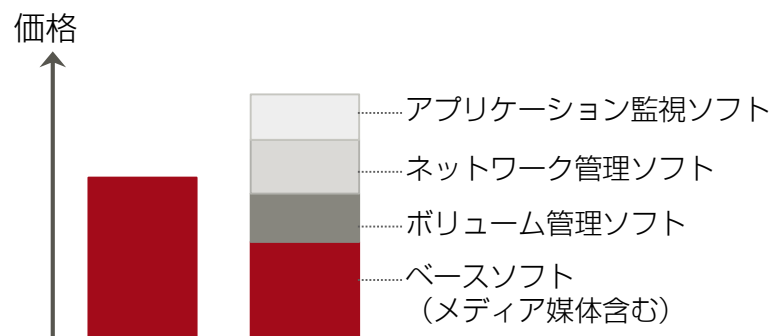
リーズナブルなクラスタシステムの確立

Solaris標準機能の使用や、仮想サーバ環境への対応により、安価なクラスタシステムを構築できます。

■ 追加ソフトウェア不要

- ・ ディスクのボリュームやネットワークなどの管理はすべてSolaris標準機能で実現します。
- ・ さらにアプリケーションの監視機能もSolaris Clusterのライセンス契約に含まれているため、特別なオプションソフトウェアを購入する必要はありません。

クラスタソフトウェアの価格イメージ（※1）

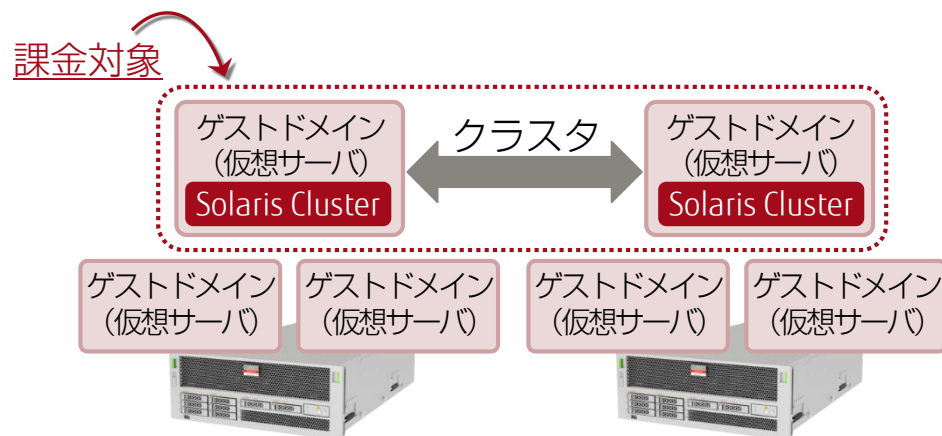


Solaris Cluster ほかのクラスタソフトウェア

※1：本グラフはイメージです。厳密なデータではありません。

■ ライセンスコストの削減

- ・ Solaris Clusterのライセンスコストは「サーバのCPU数（CPUコア数）」によって決定されます。
- ・ 仮想サーバを使用する場合、ライセンスコストは仮想サーバに割り当てたCPU分のみとなるため、安価なクラスタ環境を実現できます（※2）。



※2：Oracle社認定の「Hard Partitioning」に該当する、仮想化構成である必要があります。

クラスタシステムの構築や管理を支援する環境が充実しています。

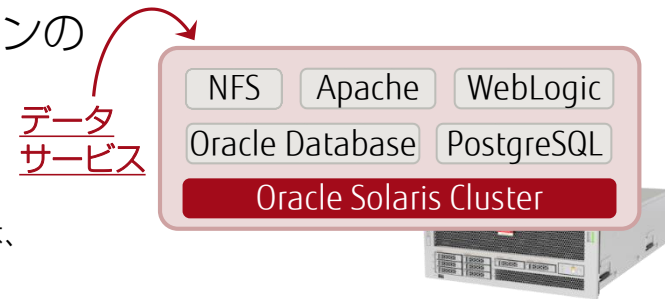
■ 標準ファイルシステムによるディスク管理

- ・ ディスク・ファイルシステムの管理は、Solaris標準のファイルシステム（ZFS/UFS）をそのまま使用できます。これらの操作は単一のサーバ環境（Solaris Clusterを導入していない環境）と同じです。

■ 豊富なアプリケーションサポート 「データサービス」

- ・ Solaris Clusterには、Oracle Databaseをはじめとした、主要な業務アプリケーションの監視・管理パッケージが標準で用意されています。このパッケージのことを「データサービス」と呼びます。
- ・ データサービスの導入により、業務アプリケーションのクラスタシステムを簡単に構築できます。特別な開発作業（スクリプト作成など）は必要ありません。

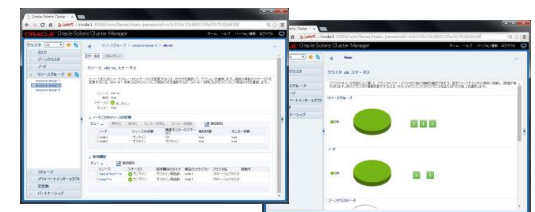
※ データサービスの導入方法やサポートしている主要アプリケーションについては、[「7.アプリケーション／サービスのリソース登録」](#)をご参照ください。



■ GUIベースの管理 「Oracle Solaris Cluster Manager」

- ・ Webブラウザを使用して、クラスタの管理や構成変更が可能です。

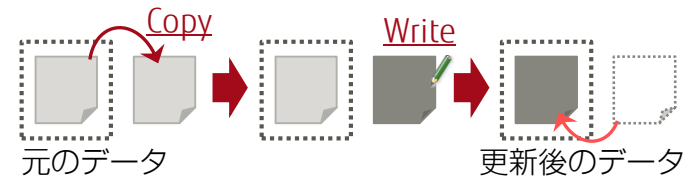
※ 詳しくは、「[参考情報 - Oracle Solaris Cluster Manager の導入](#)」をご参照ください。



Solaris Clusterで構築するクラスタシステムのデータ管理に使用するZFSファイルシステムには、信頼性・可用性を高める様々な仕組みが備わっています。

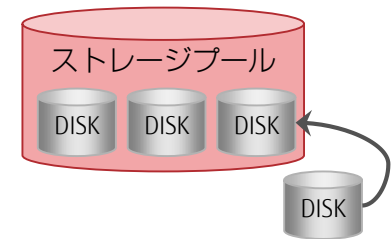
■ コピーオンライト（Copy On Write）によるデータ更新

- ・ ZFSはデータの更新処理を実行するとき、既存のデータのコピーを変更して、その後元のデータと入れ替えます。
- ・ データの更新途中で障害が発生しても、更新前の状態に戻り、データの破損・不整合は起きません。



■ RAID機能 - ストレージプール -

- ・ 複数のディスクをまとめて、「ストレージプール」と呼ばれる仮想ストレージとして管理します。
- ・ ストレージプールによってディスクの冗長化を確立し、ディスク追加による容量拡張も無停止で実行できます。
 - RAID-1, RAID-1+0, RAID-Z (RAID-5に類似) などの構成が可能です。



動的拡張可能

■ チェックサム - データの自己修復 -

- ・ データとそのチェックサムを異なるブロックに保存して、チェックサムの信頼性を向上させています。
- ・ チェックサムによってデータの不整合を検出した場合、自動で修復処理が実行されます。



・ ZFSファイルシステムの詳細は、以下の資料をご参照ください。

『Oracle Solaris 11 ZFSを使ってみよう』 <http://www.fujitsu.com/jp/sparc-technical/document/solaris/index.html#zfs>

Oracle Solaris Clusterを用いた クラスタシステムの構築

Oracle Solaris Clusterを用いたクラスタシステムの構築のながれや
ポイントを説明します。

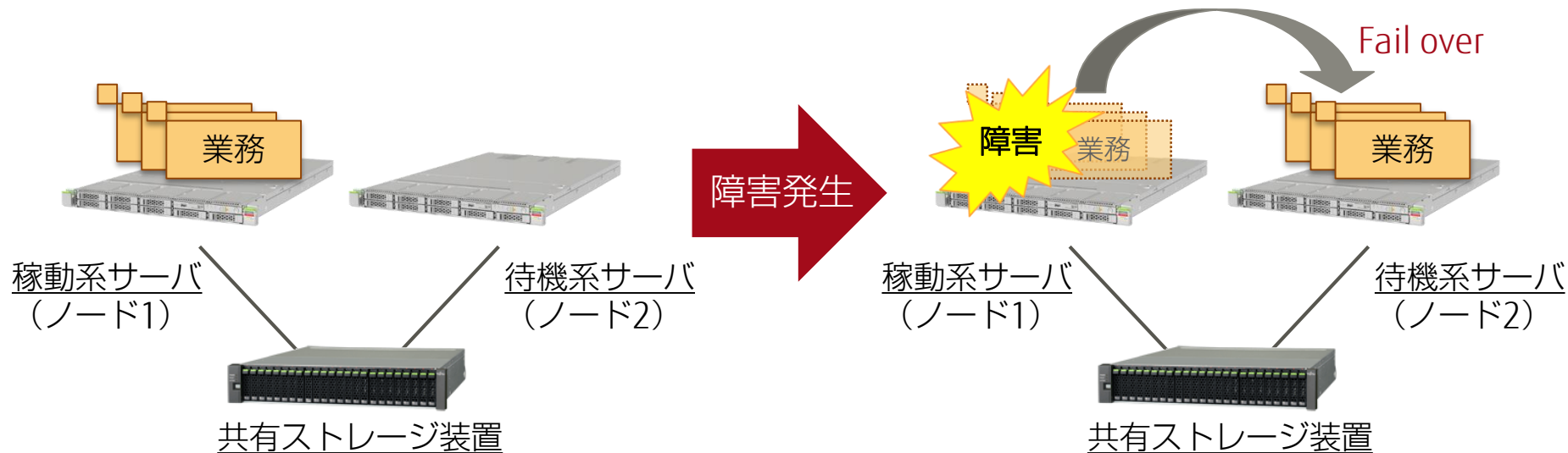
別資料の

『Oracle Solaris Clusterを使ってみよう（構築手順書）』
と併せて、ご参照ください。

■ 構成概要

■ ハードウェア構成

- ・ 通常時は稼動系サーバですべての業務アプリケーションを実行
- ・ 稼動系サーバの障害発生時に、待機系サーバに業務を切替（非対象系 運用待機）



ハードウェア

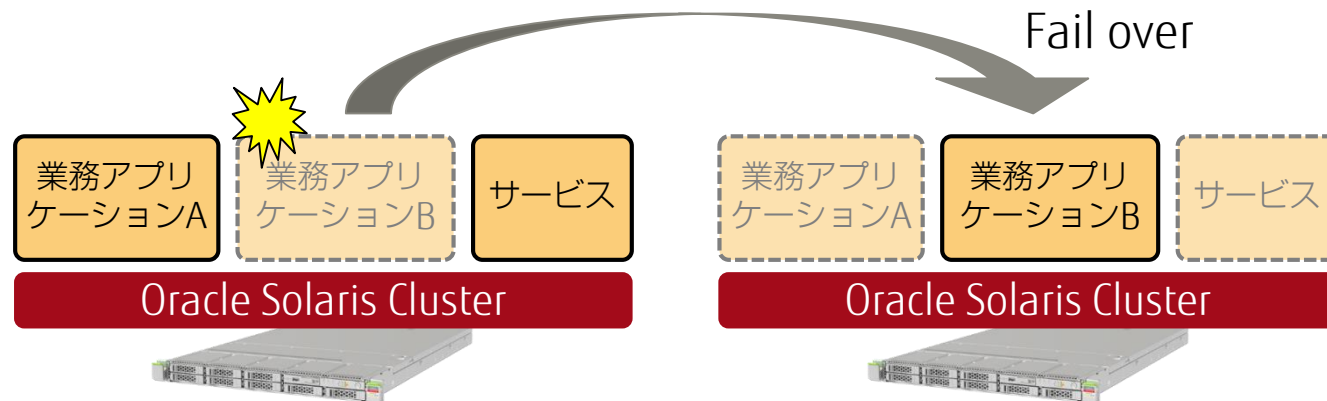
- ・ サーバ： SPARC M10-1
- ・ 共有ストレージ： ETERNUS DX100 S3

ソフトウェア

- ・ OS/SRU： Oracle Solaris 11.3 / SRU 16091 (SRU 11.3.4.0)
- ・ Solaris Cluster： Solaris Cluster 4.3 (SRU 4.3.5.3.1)

■ ソフトウェア構成

- ・ Solaris Cluster に様々なソフトウェアを登録することで、ソフトウェアを含めた障害監視も可能になります。
- ・ 稼動系サーバの一部のソフトウェアに障害が発生した場合、対象ソフトウェアのみを待機系サーバに切り替えることができます。



- ・ 本書では、以下のソフトウェア（業務アプリケーション／サービス）を登録します。

業務アプリケーション／サービス

- ・ Apache（Webサービス）
- ・ NFSサービス
- ・ Oracle Database

仮想サーバ

- ・ Oracle VM Server for SPARC

Solaris Clusterを用いたクラスタシステムの構築には、2種類のネットワークが必要です。

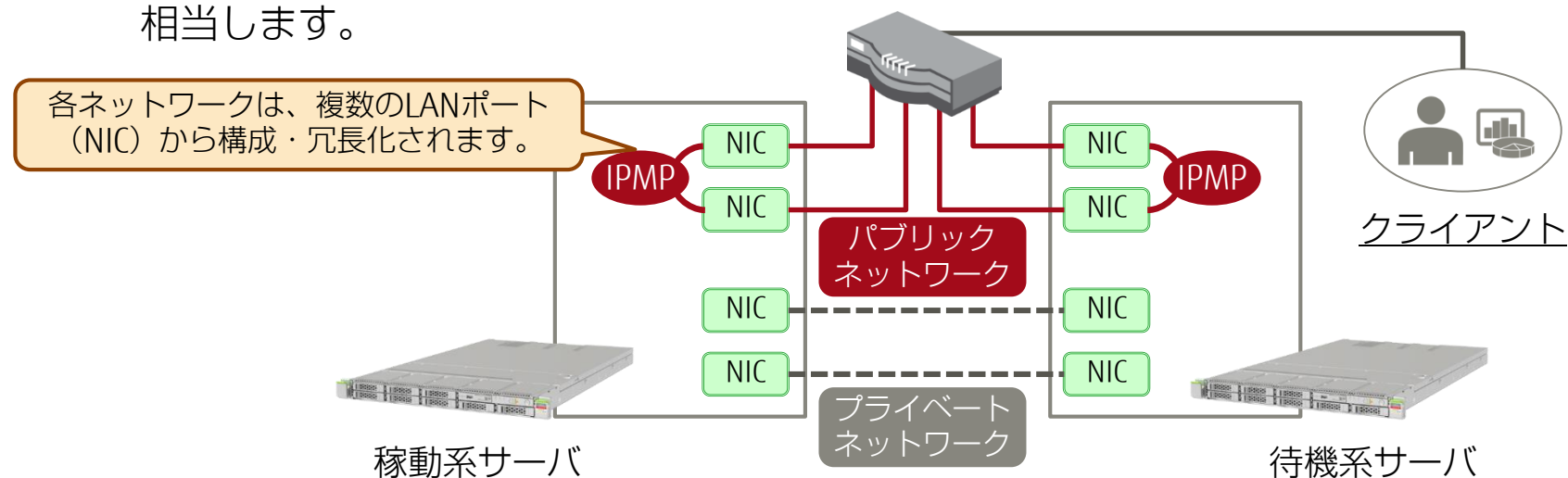
■ 必要なネットワーク（前提条件）

■ パブリックネットワーク（Public Network）

- ・アプリケーションやクライアント（システム）がアクセスするネットワークです。
- ・一般的（ほかのクラスタソフトウェアなど）には業務LANなどと呼ばれるものです。

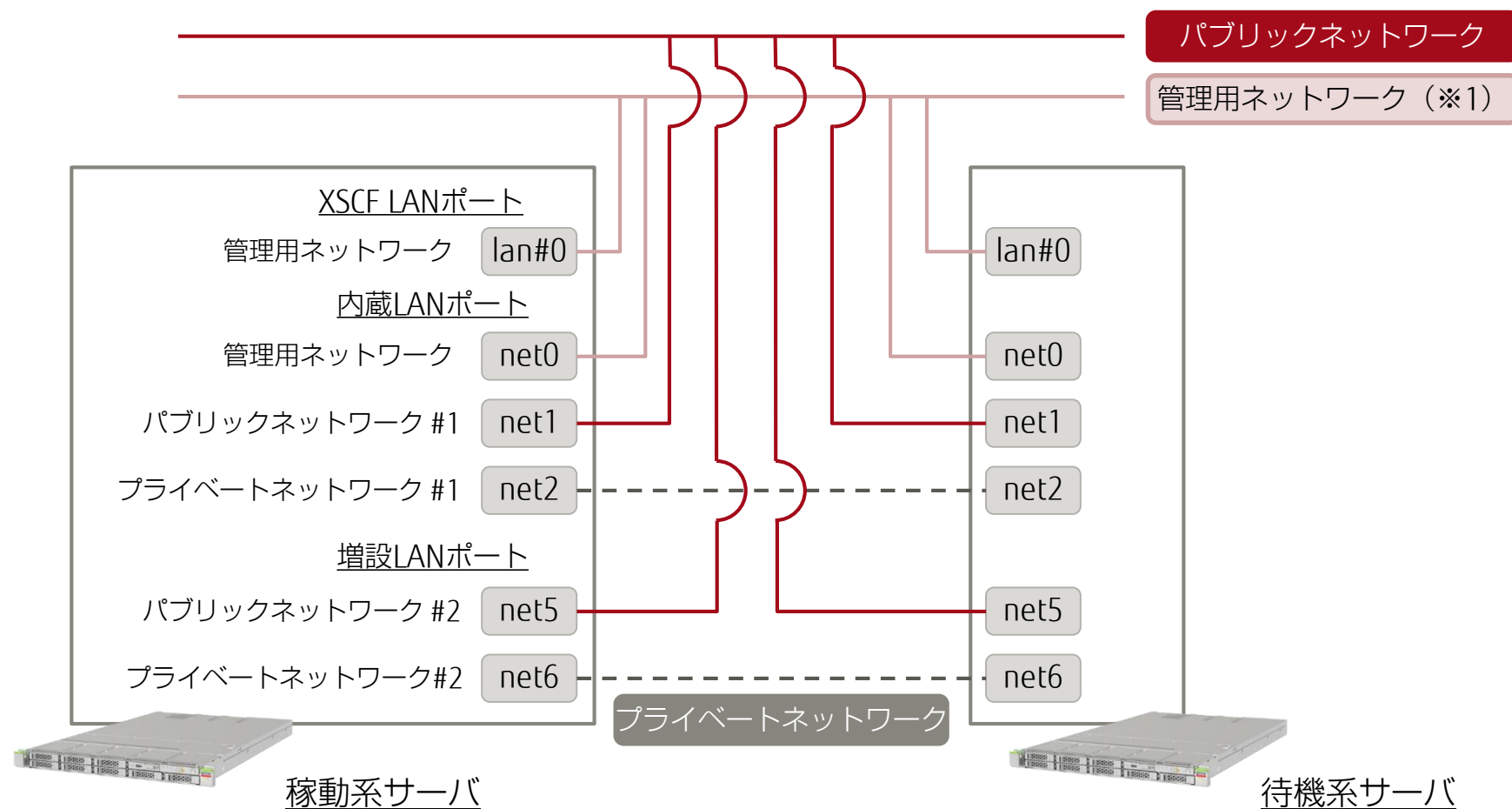
■ プライベートネットワーク（Private Network）

- ・ノード間監視（正常に起動しているかどうか）やクラスタファイルシステムに使用します。
 - クラスタファイルシステムについては、「[②リソースの作成 -Solaris Clusterで扱うファイルシステム-](#)」参照
- ・一般的には、クラスタインターコネクトや系間パスと呼ばれるネットワークに相当します。



■ ネットワーク構成

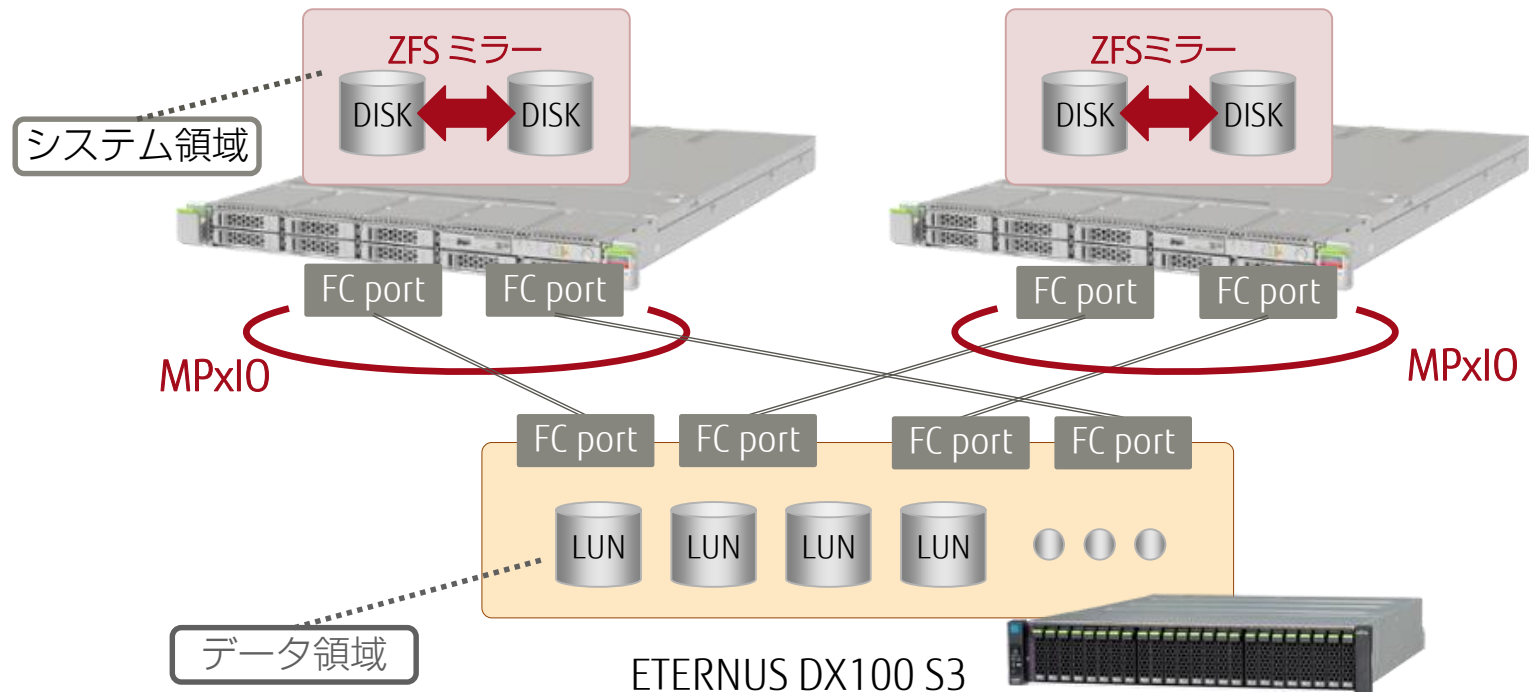
■ 2台のサーバ（ノード）で同一の構成



※1：本書で構築する環境では、サーバ管理や内部通信用（NTPなど）として、「管理用ネットワーク」を別途構築しています。

■ ディスク構成

- システム（ブート）領域：サーバ本体の内蔵ディスク（HDD）を使用
 - ・ ZFSによるミラーリング
- データ領域：共有ストレージ（ETERNUS）を使用
 - ・ MPxIOによるFCのマルチパス



1. サーバの環境確認・OSの設定

1. サーバの環境確認・OSの設定

2. ディスクの冗長化設定

3. Oracle Solaris Cluster のインストール

4. クラスタへのノードの登録

5. クラスタの初期設定

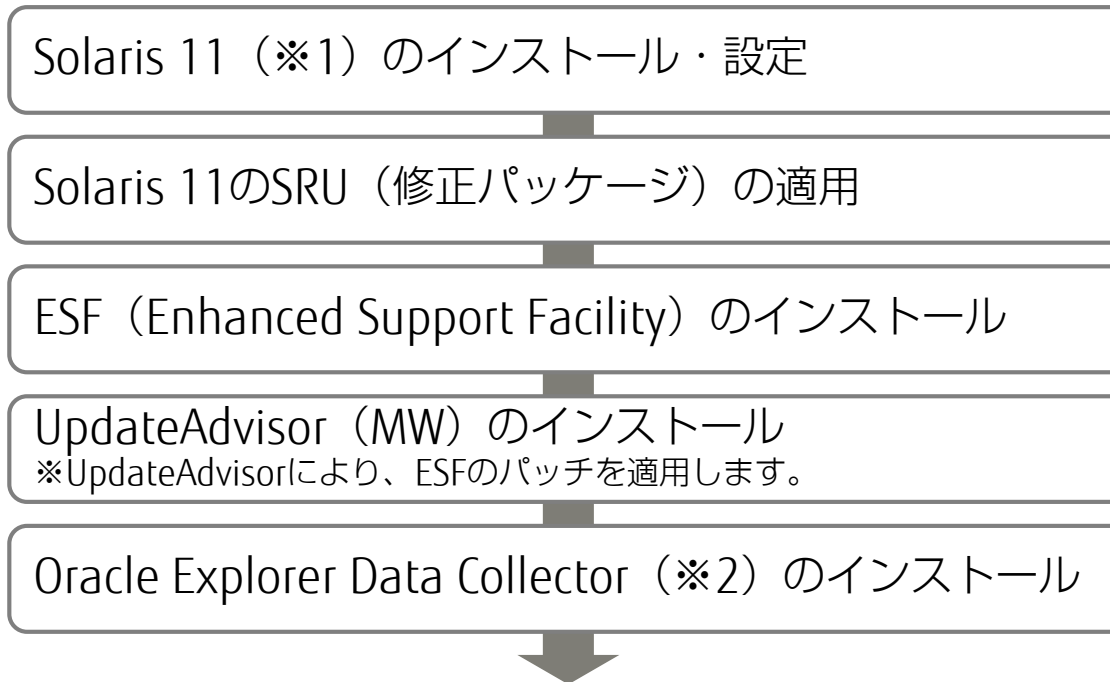
6. リソースの設定

7. アプリケーション／サービスのリソース登録



各サーバで以下の作業を完了させておく必要があります。

■ サーバの環境確認のながれ



※1：本書で扱うOracle Solaris Cluster 4.3の対応OSは、Solaris 11.2以降です。

※2：Oracle Explorer Data Collector は、障害解析のためにシステム構成情報を採取するツールです。



- ここでは、ストレージの構築・設定内容は記述していません。
- 上記のソフトウェアのインストールや設定については、以下の資料をご参照ください。
『Oracle Solaris 11を使ってみよう』 <http://www.fujitsu.com/jp/sparc-technical/document/solaris/index.html#solaris11-use>

■ OBPの設定

■ 環境変数 「local-mac-address?」

- ・デフォルト値である、「true」に設定されている必要があります。

■ 環境変数 「auto-boot?」

- ・「false」に設定することを推奨します。

■ OSの設定 1/2

■ カーネルパラメータ 「/etc/system」

- ・「set halt_on_panic=1」の記述を追加します。

```
# vi /etc/system
*ident  "%Z%%M% %I%      %E% SMI" /* SVR4 1.5 */

~ (省略) ~
set halt_on_panic=1
~ (省略) ~
set kmem_lite_maxalign = 8192
* End FJSSVssf (do not edit)
```

記述追加

- ・本設定により、OSパニック発生時にノードをOBPで停止させます。
- ・クラスタの構築完了後に、本設定は削除します。

■ OSの設定 2/2

■ rootの環境設定ファイル「/root/.profile」

- ・ Solaris Cluster のコマンドのPATH (/usr/cluster/bin) を追加します。
- ・ LANGの設定を追加します。
 - ・ 本書では、「ja_JP.UTF-8」を選択しています。

■ ホスト設定ファイル「/etc/hosts」

- ・ 各ノードの管理用ネットワーク（環境構築用）のIPアドレスとホスト名を記述します。

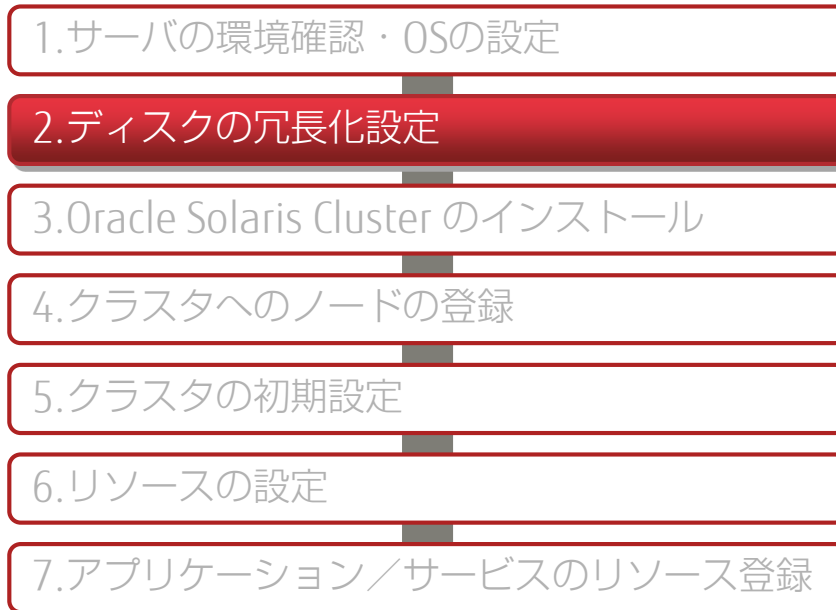
■ ネットワーク設定

- ・ RPC（※1）通信サービス
 - ・ 外部アクセスを有効に設定しておきます。
 - ・ Solaris Clusterでは、クラスタシステムの構築やノード間通信にRPCを使用します。

※1： RPC（Remote Procedure Call）・・・ ネットワーク上のほかのコンピュータでプログラムを実行する技術・プロトコル

- ・ 手動ネットワーク構成（DefaultFixed NCP）
 - ・ 有効に設定しておきます。
 - ・ 自動ネットワーク構成「NWAM（Network Auto-Magic）」は使用できません。

2. ディスクの冗長化設定

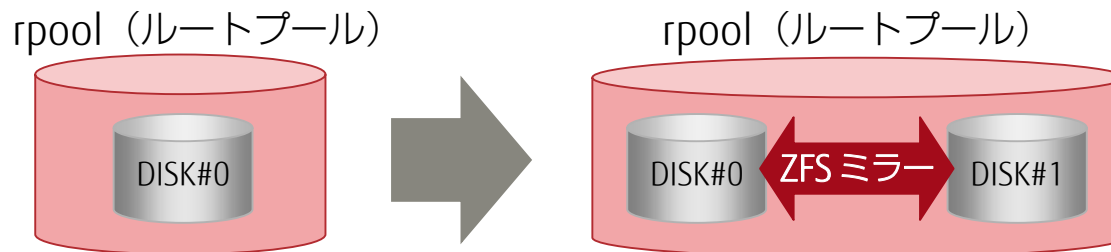


Solarisやサーバの標準機能を使用して、ディスクとFC（ファイバチャネル）の冗長構成を構築します。

■ サーバ内蔵ディスク（システム領域として使用する場合のみ）

- ・主に以下の方法で、ディスクの冗長構成を構築します。
 - ・サーバのハードRAID機能
 - ・ZFSのミラーリング機能
 - ZFSの場合、システム領域のRAID構成はRAID1（ミラーリング）のみとなります。

ZFSによるミラーリング（システム領域：rpool）



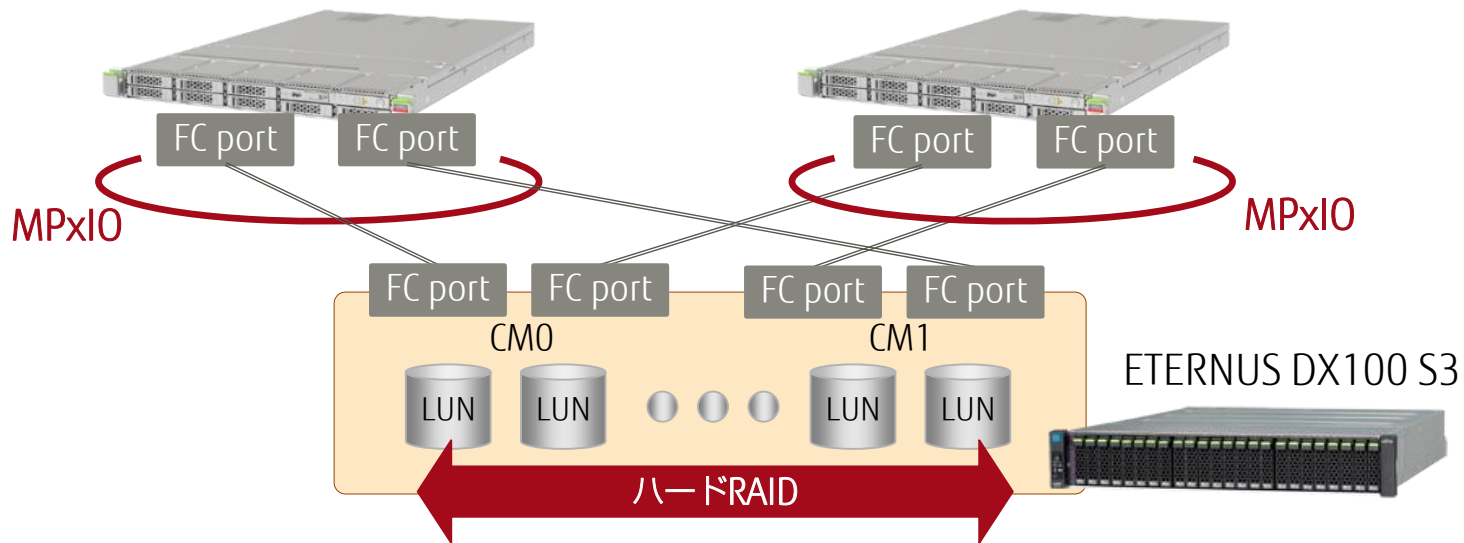
- ・本書ではZFSのミラーリング機能を使用します。
- ・ZFSの設定については、以下の資料もご参照ください。

『Oracle Solaris 11 ZFSを使ってみよう』

<http://www.fujitsu.com/jp/sparc-technical/document/solaris/index.html#zfs>

■ 共有ディスク（ETERNUS DXシリーズ）

- ・ ETERNUSのハードRAID機能（RAIDグループなど）を使用して、物理ディスクを冗長化させます。
- ・ Solaris標準のマルチパス機能「MPxIO」を使用して、FCのマルチパスを構築します。



・MPxIOによるFCのマルチパスの設定は、以下資料もご参照ください。

『SAN Boot 環境構築ガイド』

<http://updatesite.jp.fujitsu.com/unix/jp/download/driver/pfca-8info/>

3. Oracle Solaris Cluster のインストール

1. サーバの環境確認・OSの設定

2. ディスクの冗長化設定

3. Oracle Solaris Cluster のインストール

4. クラスタへのノードの登録

5. クラスタの初期設定

6. リソースの設定

7. アプリケーション／サービスのリソース登録



■ インストール形態

- ・ Solaris Cluster はSolaris 11で導入された、IPS (Image Packing System) と呼ばれるフレームワークに基づいて、管理されます。
 - パッケージはリポジトリサーバよりネットワーク経由でインストールします。
- ・ リポジトリサーバには、Solaris ClusterのリポジトリおよびSRUを格納しておきます。
 - 「OS (Solaris 11) のリポジトリ・SRU」とは別のものです。ご注意ください。

■ インストール手順


①Solaris Clusterのリポジトリ・SRUの入手

- ・ 媒体をリポジトリサーバへ格納

②リポジトリサーバの構築

- ・ リポジトリのコピー
- ・ SRUによって、リポジトリを更新

③Solaris Clusterのパッケージのインストール

- 
- 次ページ以降で、各手順について解説します。

① Solaris Clusterのリポジトリ・SRUの入手

■ Solaris Clusterのリポジトリ・SRUのダウンロード

- ・ Solaris Cluster 4.x の媒体（リポジトリ・SRU）は、SupportDesk-Webよりダウンロードできます。



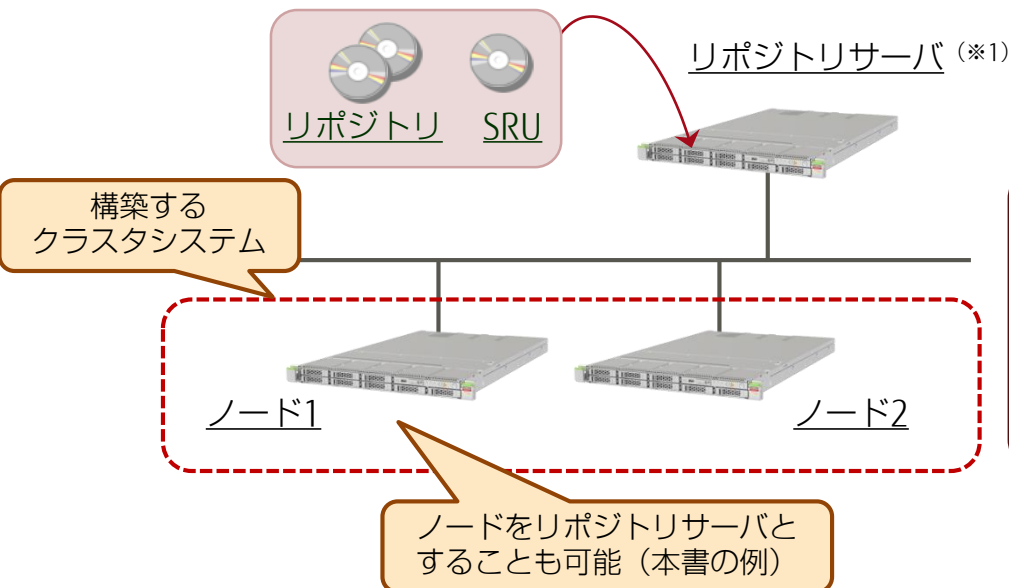
SupportDesk-Web

<http://eservice.fujitsu.com/supportdesk/>

- ・ 弊社のSolaris Clusterのサポート契約の締結が必要です。
- ・ 媒体はISOファイル形式で提供されています。

■ 媒体の格納

- ・ 媒体をパッケージの発行元となるリポジトリサーバに格納します。



※1：リポジトリサーバの構築

左の図では、構築するクラスタシステムのノードとは別のリポジトリサーバを用意して構築していますが、ノード自身をリポジトリサーバとすることもできます。本書では各ノード自身をリポジトリサーバとした、構築例を紹介します。

② リポジトリサーバの構築 1/4

■ リポジトリ用のファイルシステムを作成

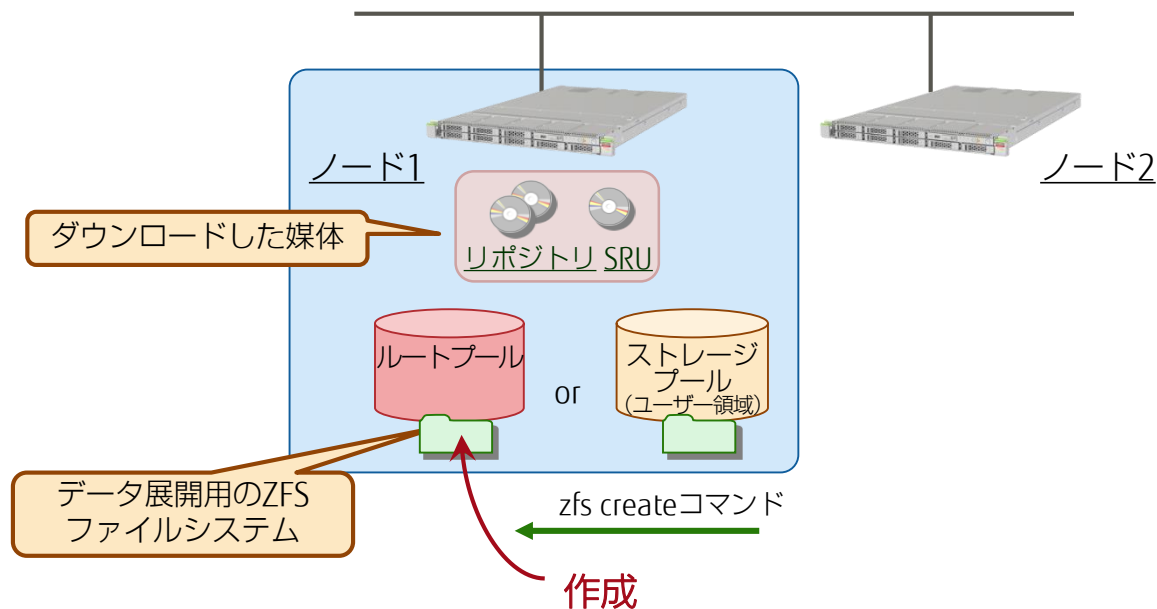
- ・ リポジトリのデータ展開用のZFSファイルシステムを作成します。

書式 : `zfs create` [オプション] ファイルシステム名

オプション : `-o compression=on` 圧縮の有効化 (必須ではありません)

```
# zfs create -o compression=on rpool/export/repo_osc4.3
```

- 上記コマンドでは、rpool (ルートプール) 配下にファイルシステムを作成していますが、rpoolとは別のディスク (ストレージプール・ファイルシステム) に作成することを推奨します。

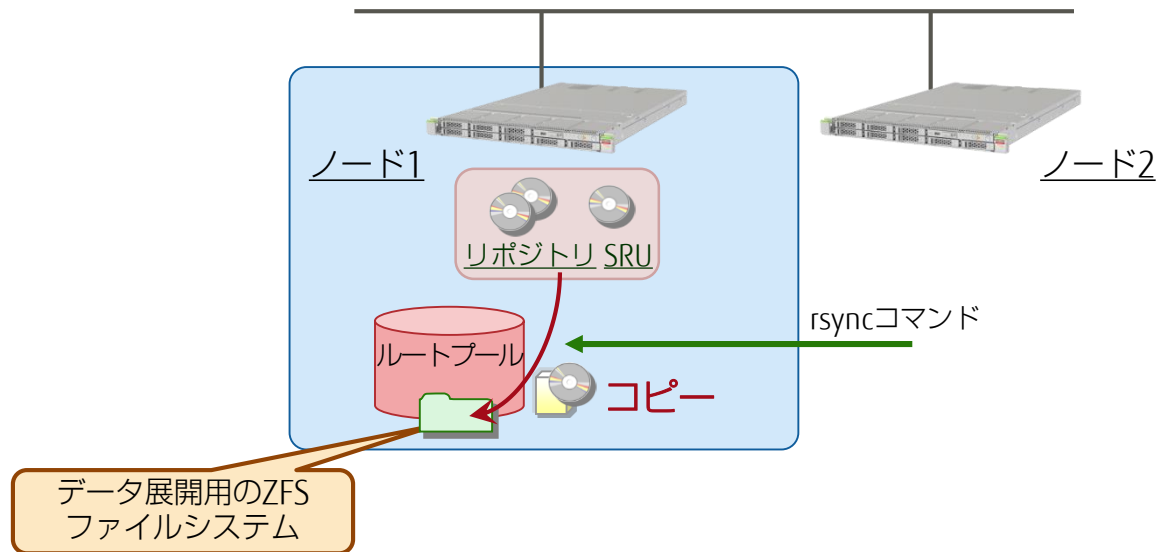


② リポジトリサーバの構築 2/4

■ リポジトリのコピー

- ・ リポジトリの媒体 (ISOファイル) をマウントします。
- ・ rsync コマンドを使用して、リポジトリのファイルをコピーします。

```
# mount -F hsfs -r /export/ISO/osc-4_3-repo-full.iso /mnt  
# rsync -a /mnt/repo /export/repo_osc4.3  
# umount /mnt
```



② リポジトリサーバの構築 3/4

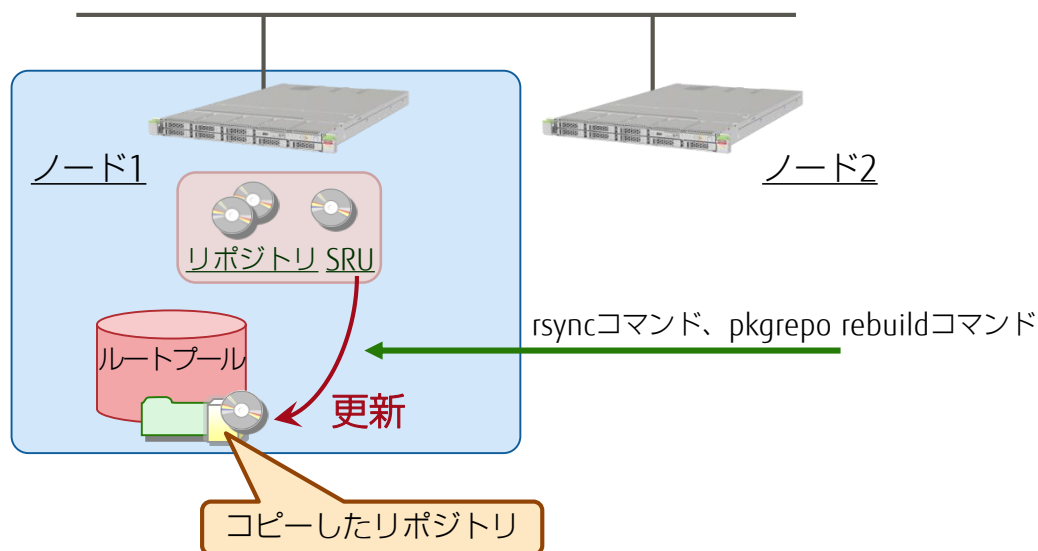
■ SRUによるリポジトリの更新

- ・SRUの媒体をマウントします。
- ・rsync コマンドを使用して、リポジトリのファイルを最新に更新します。

```
# mount -F hsfs -r /export/ISO/osc-4_3_5_3_1-repo-incr.iso /mnt  
# rsync -a /mnt/repo /export/repo_osc4.3  
# umount /mnt
```

- ・リポジトリの再構成を実施します。

```
# pkgrepo rebuild -s /export/repo_osc4.3/repo
```



② リポジトリサーバの構築 4/4

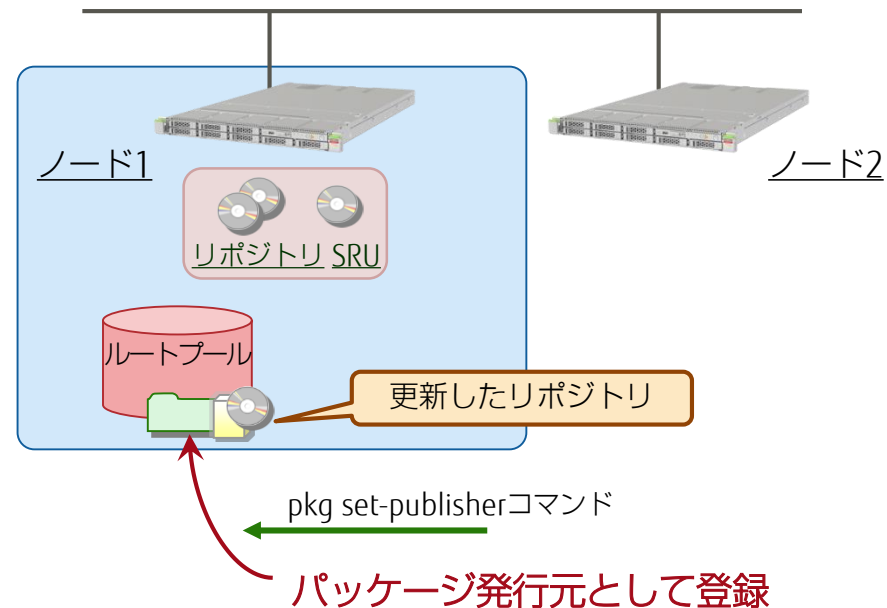
■ パッケージ発行元（publisher）の登録

- ローカルのディレクトリをパッケージ発行元として、登録します。

書式 : pkg set-publisher [オプション] publisher名

オプション : -p ディレクトリ ⇒ リポジトリの格納ディレクトリを指定

```
# pkg set-publisher -p /export/repo_osc4.3/repo ha-cluster
```

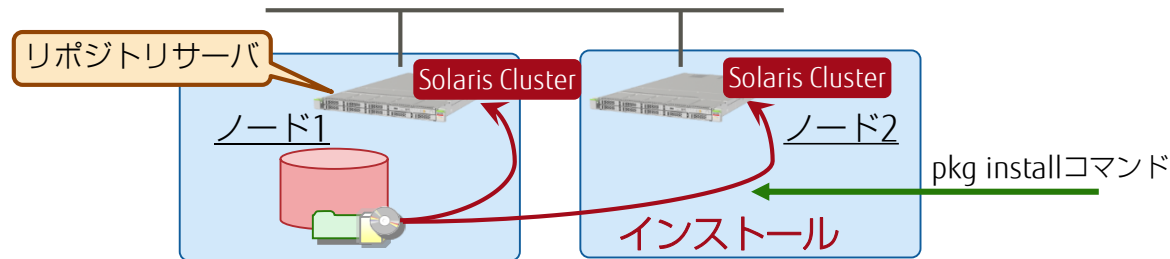


③ Solaris Clusterのパッケージのインストール

■ パッケージのインストール

- ・ pkg install コマンドを使用して、「ha-cluster-framework-full」パッケージグループをインストールします。

```
# pkg install -v --accept ha-cluster-framework-full
```



■ インストール後の環境確認

- ・ cnode コマンドを使用して、以下を確認できます。
 - Solaris Cluster のバージョン、SRUの版数
 - インストールされたパッケージ

```
# cnode show-rev -v
Oracle Solaris Cluster 4.3.5.3.1 for Solaris 11 sparc
ha-cluster/developer/agent-builder: 4.3-0.24.0
ha-cluster/developer/api: 4.3-3.1.0
ha-cluster/group-package/ha-cluster-framework-full: 4.3-0.24.0
~ (省略) ~
```



・ cnode コマンドは、主にクラスタのノード管理（ノードの追加／削除、状態表示など）に使用されます。

《参考》 Solaris Clusterのパッケージグループ 1/2

Solaris Cluster のパッケージグループは様々な種類のものが存在します。

機能	パッケージグループ (※1)			
	ha-cluster-full	ha-cluster-framework-full	ha-cluster-data-services-full	ha-cluster-minimal
フレームワーク	○	○	○	○
エージェント	○		○	
ローカリゼーション	○	○		
フレームワークのマニュアルページ	○	○		
データサービスのマニュアルページ	○		○	
エージェントビルダー	○	○		
汎用データサービス	○	○	○	
グラフィカルユーザーインターフェース	○			
Geographic Edition	○			

※1：ここでは、主要なパッケージグループのみを紹介しています。



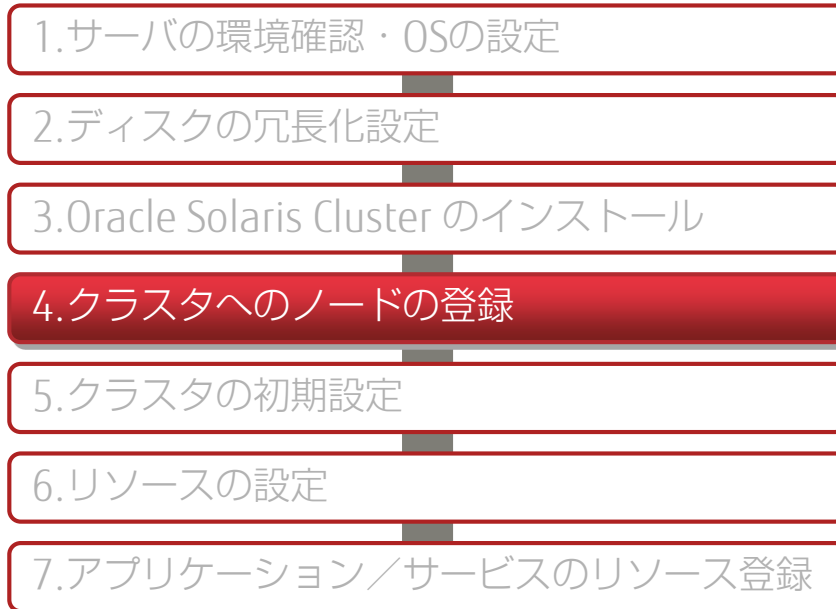
- 本書では、「ha-cluster-framework-full」を使用します。
- 各機能の説明は次ページをご参照ください。

■ パッケージグループに含まれる機能

機能	説明
フレームワーク	Solaris Cluster のベースとなる部分で、すべてのパッケージグループに含まれます。 (必須パッケージ)
エージェント	すべての業務アプリケーション／サービスのデータサービスです。 これが含まれない場合、業務アプリケーション／サービスのデータサービスを手動で 選択・インストールする必要があります。
ローカリゼーション	OSの環境に適した言語（日本語など）のパッケージを適用します。
フレームワークの マニュアルページ	manコマンドによって参照可能なフレームワークのマニュアルです。
データサービスの マニュアルページ	manコマンドによって参照可能なデータサービスのマニュアルです。
エージェントビルダー（※1）	ユーザー側でデータサービスを作成するためのツールです。
汎用データサービス（※1）	エージェントビルダーよりも簡易的なデータサービス作成ツールです。
グラフィカルユーザー インターフェース	WebブラウザによるGUI管理ツール（Oracle Solaris Cluster Manager）です。 追加でインストールすることもできます。
Geographic Edition（※1）	物理的な距離が大きく離れたサーバ間のクラスタシステム（ディザスタリカバリ運用）を構築できます。

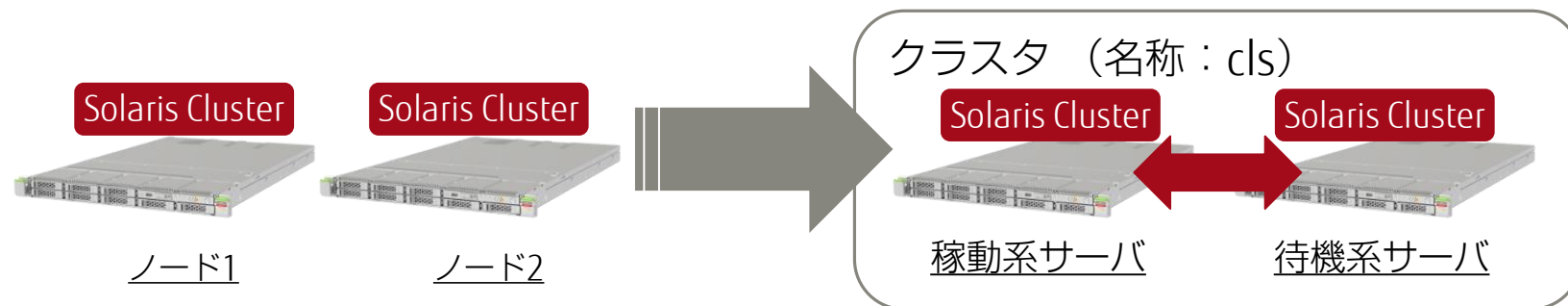
※1：本書では、エージェントビルダー、汎用データサービス、およびGeographic Editionを使用しません。

4. クラスタへのノードの登録



クラスタへのノードの登録

独立した2台のサーバ（ノード1、ノード2）を同一のクラスタとして登録します。



■ 登録方法

- ・以下の2種類があります。
 - ・構成ファイル（XMLファイル）を使用した登録
 - ① クラスタの構成ファイルの作成
クラスタの各種設定値（サーバのホスト名、プライベートネットワーク用のデバイス名など）を記述した、XMLファイルを用意しておきます。
 - ② クラスタへのノードの登録の実行
XMLファイルを指定して、cluster createコマンドを使用して、登録を実行します。
 - ・対話式メニューを使用した登録
 - scinstallコマンドを使用して、登録の対話形式メニューを起動します。
 - クラスタの各種設定はすべて、対話形式で入力できます。




・本書では、「構成ファイル（XML）による登録」を実行します。

① クラスタの構成ファイルの作成

■ 構成ファイルの作成

- ・ クラスタの設定値を記述したXMLファイルを作成します。
- ・ 記述する設定値は主に以下が含まれます。
 - cluster name : クラスタの名称
 - private_netaddr : プライベートネットワークのネットワークアドレス
 - transportAdapter name : プライベートネットワークに使用するデバイス
 - node name : 各ノードのホスト名

 構成ファイルの記述方式や各設定値の内容については、Oracle社のリファレンスマニュアルをご参照ください。
cluster(1CL) https://docs.oracle.com/cd/E62281_01/html/E62238/cluster-1cl.html
clconfiguration(5CL) https://docs.oracle.com/cd/E62281_01/html/E62238/clconfiguration-5cl.html

■ 構成ファイルの検証

- ・ xmllintコマンドを使用して、構成ファイルの書式に問題がないか確認します。

```
# xmllint --valid --noout /export/OSC/OSC-config.xml
```

- ・ 確認するのは、XMLファイルとしての書式のみです。
 - クラスタの設定をチェックすることはありません。ご注意ください。

■ 登録前準備

- ・ clauthコマンドを使用して、アクセス認証を行います。

```
node1# clauth enable -n node2
```

- ・ クラスタ登録を実行するノードからほかのノードへのアクセスを有効化します。
- ・ クラスタ登録を実行しない全ノード（本書ではノード1）で実行します。

② クラスタへのノードの登録の実行

■ 構成ファイルによる登録

- ・ cluster create コマンドを使用して、クラスタへのノードの登録を実行します。

```
node2# cluster create -i /export/OSC/OSC-config.xml
```

- ・ 待機系のノード（ノード2）で実行します。
- ・ -i オプションでクラスタの構成ファイルを指定します。

■ 登録後の環境確認

- ・ clnode, clinfo, clinterconnect などのコマンドを使用して、両方のノードがクラスタとして正しく登録されていることを確認します。
- ・ これらのクラスタ関連のコマンドは、どちらのノードでも実行可能です。

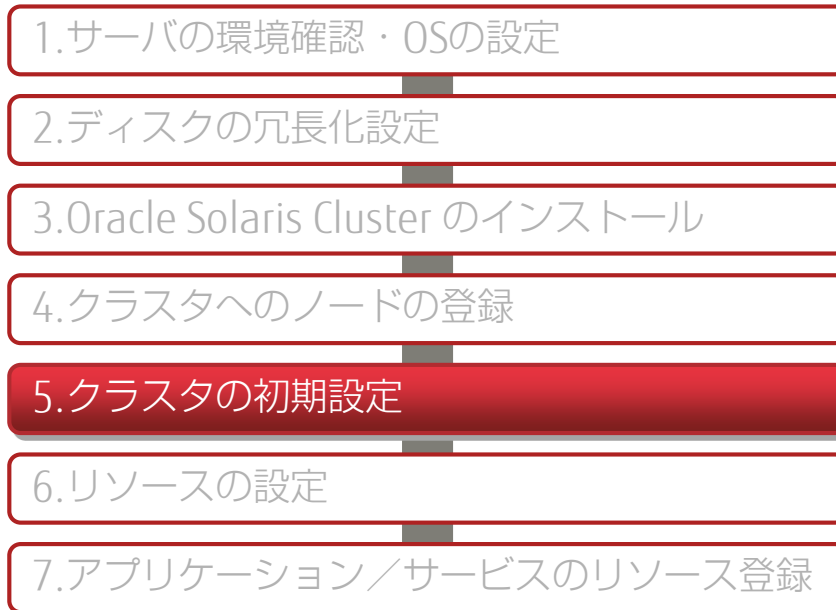
```
# clnode status
```

```
=== クラスタノード ===
```

```
--- ノードのステータス ---
```

ノード名	ステータス
-----	-----
node1	オンライン
node2	オンライン

5. クラスタの初期設定



■ クラスタの初期設定手順

①Quorum Deviceの設定

- ・ クラスタシステムの正常起動に有効な、Quorum Deviceの設定を行う
- ※共有ディスクの1つをQuorum Deviceとして設定

②NTP（Network Time Protocol）による時刻同期設定

- ・ 各ノードのサービスプロセッサ（XSCF）とOSの時刻同期を設定

③IPMPによるネットワーク冗長化

- ・ Solarisの標準機能「IPMP（IP Networking Multi Pathing）」を使用して、パブリックネットワークの冗長化構成を構築



①Quorum Deviceの設定 -Quorum（定足数）とは-

Solaris Clusterには、クラスタを正常に起動させるために、「Quorum（定足数）」、「Quorum Device（定足数デバイス）」という仕組みがあります。

■ Quorum（定足数）の概要

- ・ ノード間の共有ディスクの1つ（または複数）をQuorum Deviceとして設定します。
- ・ 各ノードとQuorum Deviceには、以下の「投票数」が定義されています。

ノード



- ・ 投票数 : 1
- ・ 投票対象 : 自ノードおよび通信可能な別のノードに投票
- ・ 投票条件 : ノードが正常に起動されている

Quorum Device

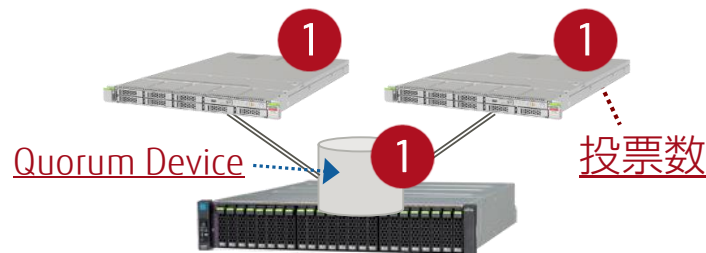


- ・ 投票数 : 接続されたノード数-1
- ・ 投票対象 : 正常な全ノードに投票
- ・ 投票条件 : 1台以上のノードから正常に接続されている

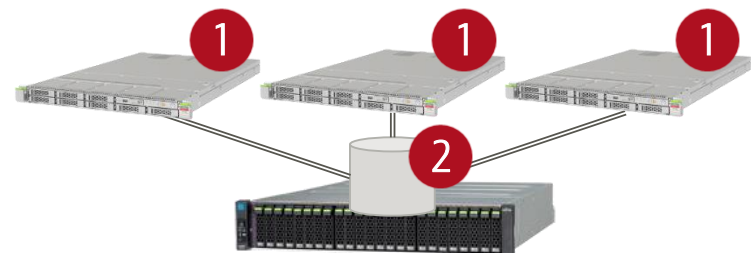
- ・ クラスタシステム内に存在するノードのうち、全投票数のうち過半数の投票を得たノードにおいて、クラスタシステムを起動できます。

■ Quorum（定足数）の構成例

2ノードの基本的な構成（本書の構成）



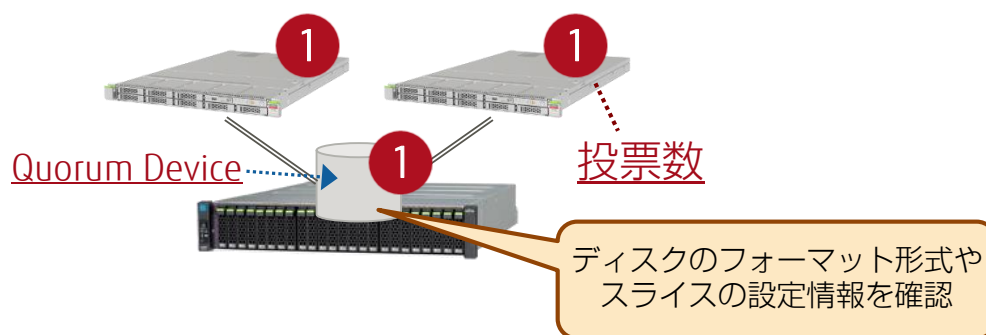
3ノードの構成



①Quorum Deviceの設定 -設定方法- 1/4

■ Step1 : Quorum Deviceに設定するディスクの確認

- ・ formatコマンドなどを使用して、Quorum Deviceに設定する共有ディスクとLUNを確認します。
 - ・ ディスクのフォーマット形式（SMIまたはEFI）やスライスの設定情報を確認します。
- ・ 2ノードによるクラスタシステムの場合、1本のみの共有ディスクをQuorum Deviceとして設定します。



Quorum Deviceのディスクのフォーマット形式

- ・ Quorum Deviceには業務システム用の共有ディスクやLUNを指定できます。
（Quorum Device専用のディスクやLUNを用意する必要はありません）
- ・ ただし、Quorum Deviceの設定後に、当該のディスクのフォーマット形式（SMI・EFI）を変更することは推奨しません（Quorum Deviceの設定が確認できなくなることがあります）。

■ Step2：共有ディスクのDIDデバイス名の確認

- ・ cldevice listコマンドを使用して、Quorum Deviceに設定するディスクの DIDデバイス名 (dX) を確認します。

```
# cldevice list -v
DID デバイス      デバイスのフルパス
-----
d1                node1:/dev/rdisk/c0t50000394281B5310d0
d2                node1:/dev/rdisk/c0t50000394281B59D4d0
d3                node1:/dev/rdisk/c0t600000E00D280000002804B800000000d0
d3                node2:/dev/rdisk/c0t600000E00D280000002804B800000000d0
~ (省略) ~
```

- ・ 確認したDIDデバイス名は、次のステップで設定するため、控えておきます。



DIDデバイスについて

- ・ Solaris Clusterでは、共有ディスクをはじめとしたすべてのストレージ装置（ノードの内蔵ディスク、テープ装置なども含め）を「DIDデバイス」という形式で管理します。
- ・ ディスク装置のDIDデバイスファイルの形式は以下のとおりです。
 - ブロックディスク： /dev/did/dsk/d[X]s[X]
 - rawディスク : /dev/did/rdisk/d[X]s[X]
- ・ ZFSストレージプールやファイルシステムなどは、すべてDIDデバイスによって管理されます。

■ Step3：共有ディスクのDIDデバイス名の設定

- ・ clsetupコマンドを使用して、クラスタの対話式設定メニューを起動します。
 - ・ このメニューではノードの追加やリソースの登録など、クラスタの一通りの設定を実行できます。
 - リソースについて詳しくは、「[リソースとリソースグループの概要](#)」参照
 - ・ 初回実行時は自動でQuorum Deviceの設定メニューが起動します。
- ・ 対話式設定メニュー内で共有ディスクのDIDデバイス名（dX）を指定します。

```
>>> 共有ディスク定数デバイスの追加 <<<
```

```
～(省略)～
```

```
どのグローバルデバイスを使用しますか (d<N>) ? d3
```

```
更新を継続しますか [[はい]] ? yes
```

```
/usr/cluster/bin/clquorum add d3
```

```
Feb 21 10:52:02 node1 cl_runtime: NOTICE: CMM: Cluster members: node1 node2.
```

```
Feb 21 10:52:02 node1 cl_runtime: NOTICE: CMM: node reconfiguration #4 completed.
```

```
コマンドが正常に完了しました。
```

■ Step4：設定内容の確認

- ・ clquorumコマンドを使用して、設定した内容を確認します。
- ・ 以下の情報が出力されます。
 - クラスタシステム内のトータルの投票数、現在の投票数
 - 各ノード、Quorum Deviceの状態（オンラインまたはオフライン）

```
# clquorum status
=== クラスタ定足数 ===
```

```
--- 定足数投票数サマリー（最新のノード再構成） ---
```

必要	現在の数	可能な数
2	3	3

```
--- ノード別の定足数投票数（現在のステータス） ---
```

ノード名	現在の数	可能な数	ステータス
node1	1	1	Online
node2	1	1	Online

```
--- デバイス別の定足数投票数（現在のステータス） ---
```

デバイス名	現在の数	可能な数	ステータス
d3	1	1	オンライン

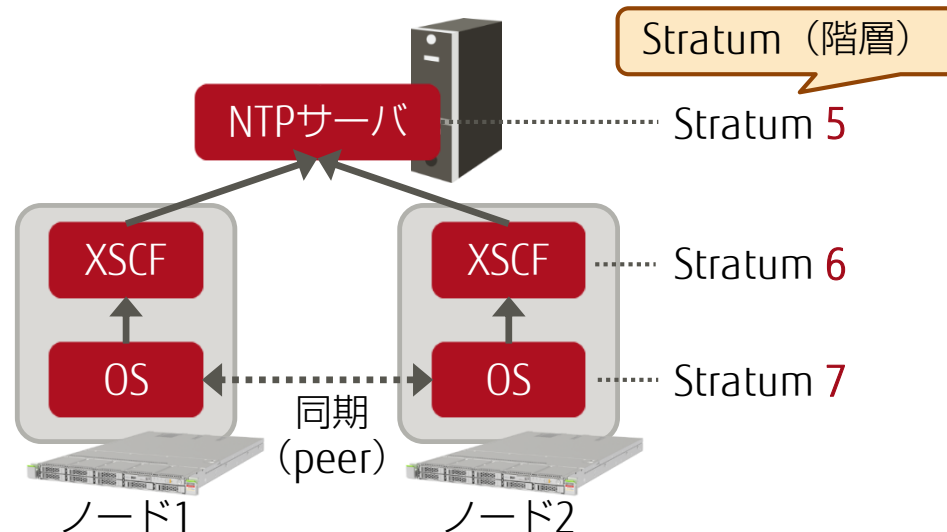
②NTPによる時刻同期設定 -概要-

■ 時刻同期

- ・ Solaris Clusterではすべてのノード（OS）間で時刻を同期させる必要があります。
 - ・ SPARC M10の場合、ノードのXSCFも含めて、時刻を同期させます。
- ・ 時刻同期にはNTP（Network Time Protocol）を使用します。
 - ・ 単一のサーバ環境（Solaris Clusterを導入していない環境）と比べて、設定方法が少し異なります。→次ページ参照

■ NTPの構成

- ・ 右図は本書の構成例です。
 - ・ XSCFは外部のNTPサーバから時刻を取得します。
 - ・ OSは同一のノードのXSCFから時刻を取得します。
- ・ さらに、同期の精度を高めるために、ノード間の同期（peer）を設定します。



本書におけるNTPの構成



・ 本書では外部のNTPサーバを使用していますが、NTPサーバは必須ではありません。ノード間でNTPの時刻同期が設定されていれば、問題ありません。

・ NTPの構成における考え方や注意事項は、以下の資料も参考にしてください。
『NTP 運用ガイド』

<http://www.fujitsu.com/jp/products/computing/servers/unix/sparc/downloads/manual/solaris/>

②NTPによる時刻同期設定 -設定方法-

■ XSCFの設定方法

- ・通常的环境（Solaris Clusterを導入していない環境）と同一の操作で設定します。
- ・設定を完了させる際には、XSCFを再起動させる必要があります。



- ・XSCFの設定方法については、以下の資料も参考にしてください。

『SPARC M10 システム システム運用・管理ガイド』

『SPARC M10 システム XSCFリファレンスマニュアル』

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

■ OSの設定方法

■ NTPサーバとNTPクライアントの設定

- ・/etc/inet/ntp.conf を編集します。
- ・通常のSolaris環境と同様です。
- ・上記設定ファイルはSolaris Cluster用に自動で再作成されています。

■ peer（同一のNTP階層による同期）の設定

- ・/etc/inet/ntp.conf.scを編集します。
- ・ファイルには、同期させるノードを指定します。
- ・ノードは、clusternode[X]-privの形式で指定します。

②NTPによる時刻同期設定 -NTPの設定ファイル-

■ Solaris ClusterのNTP設定ファイル

- ・一部の設定ファイルが自動で編集されたり、新規に作成されたりしています。
- ・格納ディレクトリはすべて、/etc/inet/ です（通常のSolaris環境と同様）。

■ 設定ファイル一覧

ファイル名	説明
ntp.conf	NTPサーバやNTPクライアントの設定を記述する基本的なファイルです。
ntp.cluster	ntp.confの初期状態のファイルです。 NTPの設定を初期化したい場合は、このファイルをntp.confの名前でコピーします。 # cp -pr /etc/inet/ntp.cluster /etc/inet/ntp.conf このファイル自体は基本的に編集する必要はありません。
ntp.conf.sc	peer（同一の階層・stratumの時刻同期サーバ）を指定するファイルです。
ntp.conf.include	ntp.conf.scのシンボリックリンクです。 このファイルは、ntp.conf内で参照されています。

③IPMPによるネットワーク冗長化 -概要-

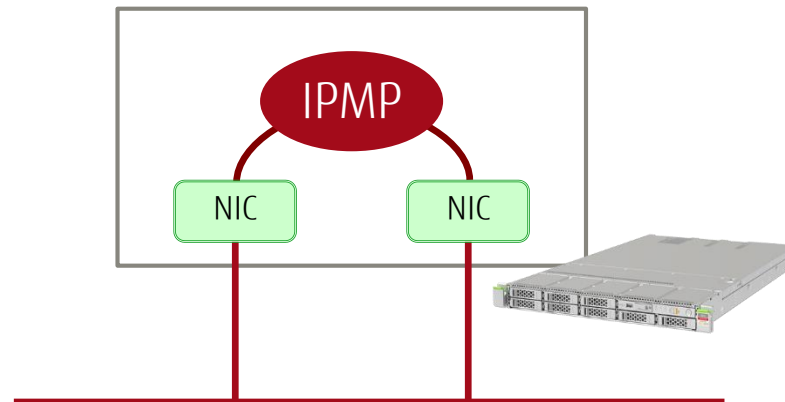
■ ネットワークの冗長化の設定方法

■ プライベートネットワークの場合

- ・ ノードのクラスタ登録時に、自動で冗長化が設定されます。

■ パブリックネットワークの場合

- ・ 手動でIPMP (IP Networking Multi Pathing) による冗長化設定をする必要があります。



- Solaris Clusterを用いたクラスタシステムの構築には、「プライベートネットワーク」と「パブリックネットワーク」が必要です。
→詳しくは、「[本書で構築するクラスタ環境 -ネットワーク-](#)」参照
- 本書では、手動で設定が必要な「パブリックネットワーク」の冗長化の設定を行います。

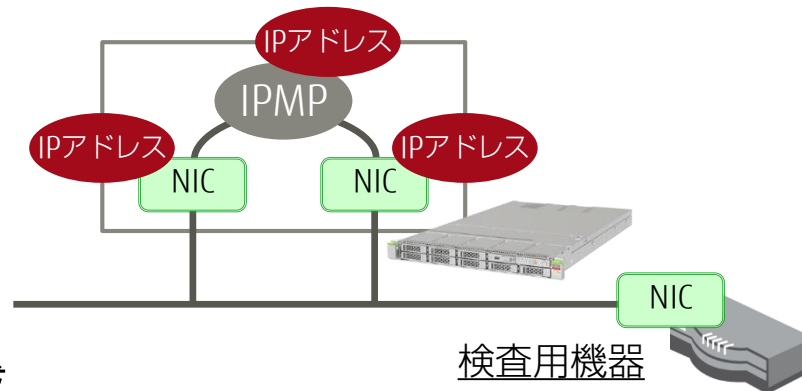
③IPMPによるネットワーク冗長化 -IPMPの動作-

IPMPの監視（障害検出）と構成には、それぞれ2種類の設定があります。

■ 障害検出

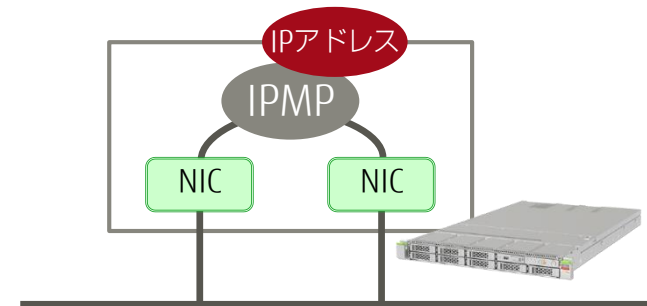
■ プローブベース（検査信号ベース）

- ・ ルータなどの同一ホスト上の監視用機器（ターゲット）を用意します。
- ・ 各NICからターゲットに対して、検査信号（ICMP）を送信して、障害検出を行います。
- ・ 引継ぎIPアドレスのほかに、各NICにIPアドレスを設定します。



■ リンクベース

- ・ 監視用機器の用意は不要です。
- ・ NICのドライバがリンク状態を監視して、障害の検出を行います。
- ・ 設定するIPアドレスは引継ぎIPアドレスのみです。



■ 構成

■ 待機型構成

- ・ 通常時は1つのNICのみを通信に使用します。
- ・ 残りのNICは、障害発生後に使用します。

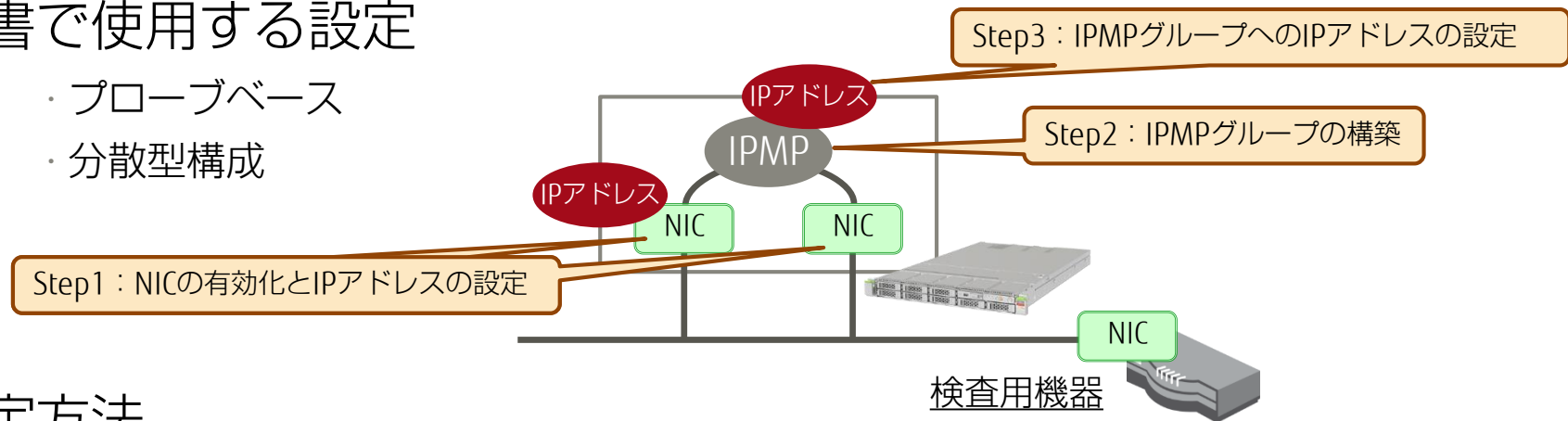
■ 分散型構成

- ・ すべてのNICを通信に使用します。
- ・ 送信/受信パケットは各NICに分散されます。

③IPMPによるネットワーク冗長化 -IPMPの設定方法-

■ 本書で使用する設定

- ・ プローブベース
- ・ 分散型構成



■ 設定方法

- ・ Step1: IPMPを構成する2つのNICを有効化して、IPアドレスを設定します。

```
# ipadm create-ip net1
# ipadm create-addr -T static -a local=192.168.5.11/24 net1/v4
# ipadm create-ip net5
# ipadm create-addr -T static -a local=192.168.5.12/24 net5/v4
```

- ・ Step2: 2つのNICからIPMPグループを構築します。

書式: `ipadm -i NICのデバイス名 (netX) -i NICのデバイス名 (netX) IPMPグループ名`

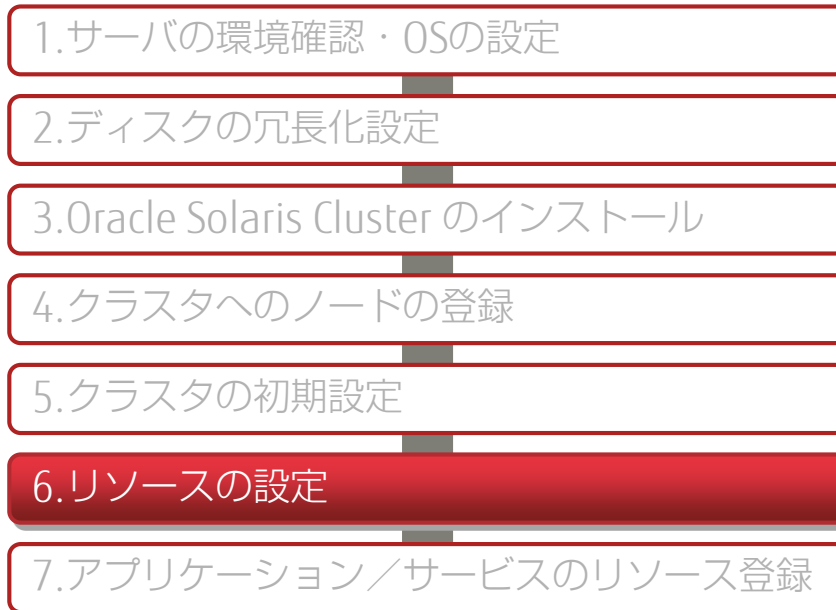
```
# ipadm create-ipmp -i net1 -i net5 sc_ipmp1
```

- ・ Step3: IPMPグループに対して、IPアドレスを設定します。

- 通常のネットワークデバイス (netX) と同様の操作です。

```
# ipadm create-addr -T static -a local=192.168.5.13/24 sc_ipmp1/v4
```

6.リソースの設定



リソースとリソースグループの概要

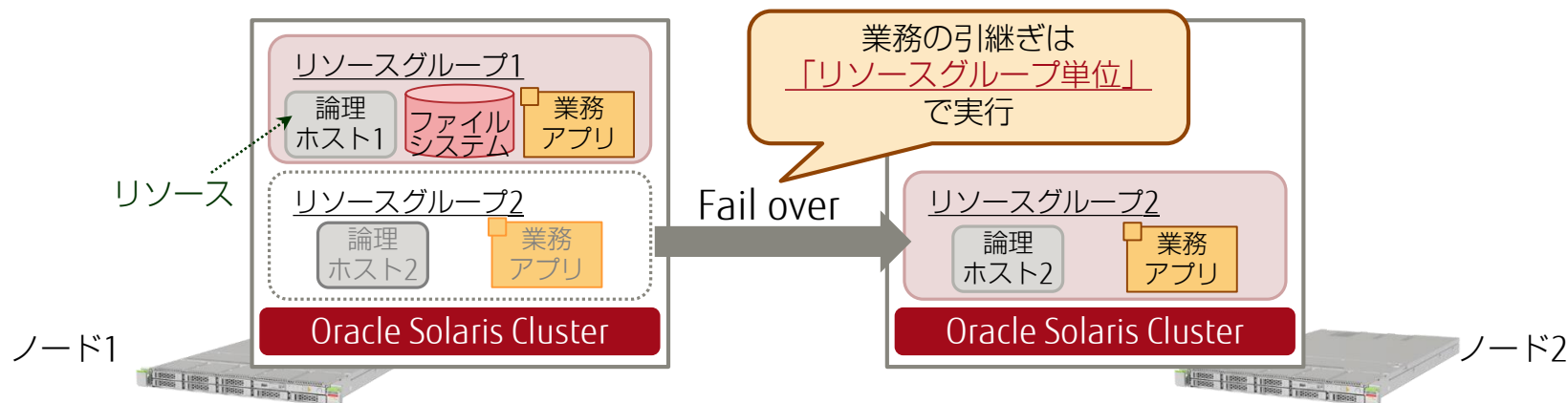
Solaris Cluster では複数のノードの論理的なホスト名や共有ファイルシステム、業務アプリケーションなどを「リソース」、「リソースグループ」という形式で管理します。

■ リソース

- ・ クラスタシステムとしてノード間で引き継いだり、共用したりする要素のことです。リソースは以下のとおりです。
 - 論理ホスト、共有ディスクによるファイルシステム、業務アプリケーション／サービス

■ リソースグループ

- ・ 単一もしくは複数のリソースによるグループ（入れ物）で、以下の2種類が存在します（※1）。
 - フェイルオーバーリソースグループ : 1つのノードのみで有効なリソースグループ（業務引継ぎ）
 - スケーラブルリソースグループ : 複数のノードで有効なリソースグループ（業務共用）
- ・ 業務の引継ぎや共用は、リソースグループ単位で実行されます。



※1：本書では、「フェイルオーバーリソースグループ」のみを扱います。

■ リソースとリソースグループの作成方法（2種類）

■ コマンドベースによる作成（本書で使用）

- ・ clresourceコマンドやclresourcegroupコマンドを使用します。
 - ・ 一部のリソースでは、OSの設定ファイル（/etc/…）を手動で編集する必要があります。

■ 対話式メニューによる作成

- ・ clsetupコマンドで起動できる、対話式メニューを使用します。
 - ・ コマンドベースよりも操作は簡単ですが、一部のリソースは対応していません。



・以降、「コマンドベースによる設定」の操作を解説します。

■ 設定手順

①リソースグループの作成

- ・ リソースを登録するリソースグループを作成

②リソースの作成

- ・ 以下のリソースを作成
論理ホスト、ファイルシステム（クラスタファイルシステム、高可用性ローカルファイルシステム）
- ・ 上記リソースをリソースグループに登録
- 業務アプリケーション／サービスのリソースグループへの登録については、[「7.業務アプリケーション／サービスのリソース登録」](#) 参照

③リソースグループの起動（有効化）

- ・ リソースグループを起動させて、リソースが有効化されていることを確認

①リソースグループの作成

■ 作成手順

- ・ clresourcegroup コマンドを使用します。
 - ・ リソースグループのほぼすべての管理（作成、削除、起動、ノード切替など）を行うコマンドです。

書式 : clresourcegroup create [オプション] リソースグループ名

・ オプション

-S : スケーラブル リソースグループを作成（指定しない場合は、フェイルオーバーリソースグループを作成）

```
# clresourcegroup create resource-group-1
```



リソースとリソースグループの作成順序

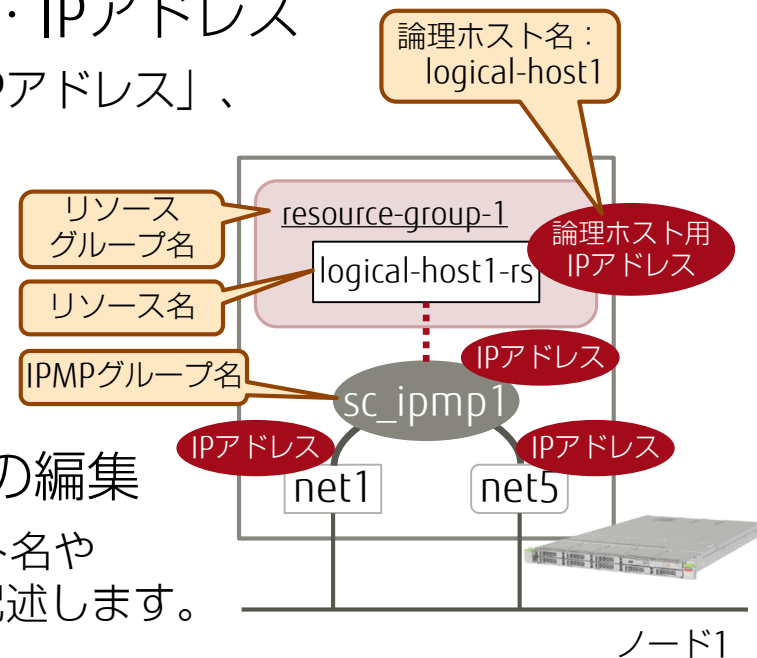
- ・ リソースグループは、リソースを作成する前に作成しておく必要があります。
リソースを作成する際、同時にリソースグループも指定する必要があるためです。

②リソースの作成 -論理ホスト-

■ 論理ホストの概要

■ クライアントからのアクセス用のホスト・IPアドレス

- ・ほかのクラスタシステムにおける、「引継ぎIPアドレス」、「仮想IPアドレス」などに相当します。
- ・IPMPのデバイスに設定したIPアドレスとは、別のIPアドレスを使用し、ノード障害時には、別のノードがそのIPアドレスを引き継ぎます。



■ 作成手順

■ Step1: OSのネットワーク設定ファイルの編集

- ・下記ファイルに、論理ホストに使用するホスト名やIPアドレス、ネットワークアドレスの情報を記述します。
 - /etc/hosts
 - /etc/netmasks

■ Step2: リソースを作成し、リソースグループに登録

- ・clreslogicalhostname コマンドを使用します。
 - オプションで、リソースグループやIPMPのグループ名などを指定します。

書式: clreslogicalhostname create -g リソースグループ -h 論理ホスト名
-N IPMPグループ名@ノードID 作成するリソース名

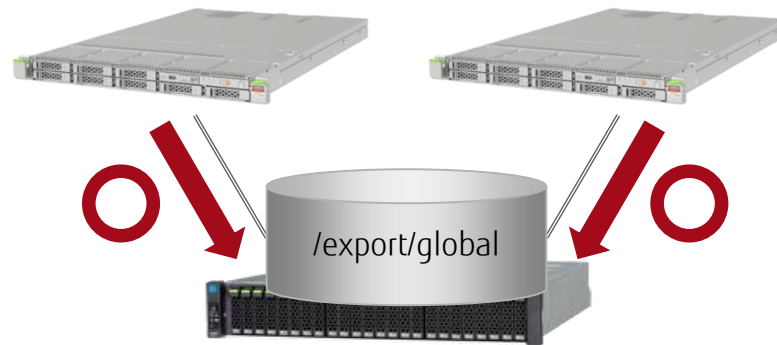
```
# clreslogicalhostname create -g resource-group-1 ¥  
> -h logical-host1 -N sc_ipmp1@1,sc_ipmp1@2 logical-host1-rs
```

②リソースの作成 -Solaris Clusterで扱うファイルシステム-

共有ディスクを管理するファイルシステムには、以下の2種類があります。

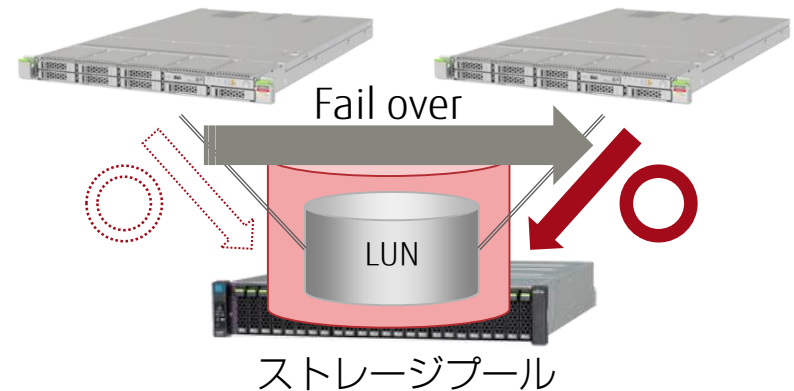
■ クラスタファイルシステム

- すべてのノードからアクセス可能です。
 - ・ 同一のパス名でアクセスします。
- **UFS**を使用して、管理・作成します。
 - ・ newfs, mountコマンドなどで管理します。



■ 高可用性ローカルファイルシステム

- 単一のノードのみアクセス可能です。
 - ・ ノード障害時にアクセス権を切り替えます。
- **ZFS**を使用して、管理・作成します。
 - ・ 共有ディスクからストレージプールを作成します。



- クラスタで管理する業務アプリケーション・業務サービスによっては、使用できるファイルシステムが制限されることがあります。
- 事前に、Oracle社のデータサービスのマニュアルより、ファイルシステムに関する要件を確認しておくことを推奨します。

■ 作成手順

■ Step1：UFSファイルシステムの作成

- ・単一のサーバ環境（Solaris Clusterを導入していない環境）と同様の手順で、共有ディスクからUFSファイルシステムを作成します。

```
# newfs /dev/rdisk/c0t600000E00D280000002804B80000000d0s0
# fsck /dev/rdisk/c0t600000E00D280000002804B80000000d0s0
```



- ・UFSファイルシステムの操作については、以下の資料もご参照ください。
『Oracle Solaris 10を使ってみよう』 <http://www.fujitsu.com/jp/sparc-technical/document/solaris/os10.html#solaris10-use>

■ Step2：ファイルシステムのマウント

- ・マウント用のディレクトリを作成します（両方のノードで実行します）。

```
# mkdir /export/global
```

- ・自動マウント設定ファイル「/etc/vfstab」を編集します。
 - ・通常のUFS設定時と同様の編集ですが、ディスクのパスをDIDデバイスのパスで記述します。
→「[① Quorum Deviceの設定-設定方法- 2/4](#)」参照
- ・ファイルシステムをマウントします。
 - ・マウントはノード1（優先ノード）のみで実行します。
 - ・マウント後、ファイルシステムはすべてのノードで認識されます。

```
node1# mount /export/global
```



クラスタファイルシステムのリソース登録

- 共有ストレージを管理するファイルシステムには、クラスタファイルシステムと高可用性ローカルファイルシステムの2種類があります。
- 前ページまでの操作によって、クラスタファイルシステムの作成は完了しています。
本書ではそれに加えて、クラスタファイルシステムを高可用性ローカルファイルシステムのリソースとして登録することをご紹介・推奨します。
- リソース登録によって、以下のようなメリットが得られます。
 - 自動でファイルシステムの定期的な動作確認を実施
 - ほかのリソース（主に業務サービス・アプリケーション）との起動/停止順序・依存関係を設定※UFSファイルシステムの環境で、高度な設定が可能になります。

■ Step3：リソースタイプの登録

- 高可用性ローカルファイルシステムのリソースタイプ「SUNW.HAStoragePlus」を登録します。
 - 初期状態で登録されているリソースタイプは、論理ホスト（SUNW.LogicalHostname）と共有IPアドレス（SUNW.SharedAddress）のみです。
 - これら以外のリソースを使用するときは、リソースタイプの登録が必要です。すでに、リソースタイプ「SUNW.HAStoragePlus」を登録済みの場合は、ここでの操作は不要です。次のStepへ進みます。

```
# clresourcetype register SUNW.HAStoragePlus
```

ファイルシステムの場合は、リソースタイプに「SUNW.HAStoragePlus」を登録

■ Step4：リソースを作成し、リソースグループに登録

- 論理ホスト以外のリソースの作成・登録には `clresource create` コマンドを使用します。
書式：`clresource create -g リソースグループ -t リソースタイプ -p オプション` 作成するリソース名

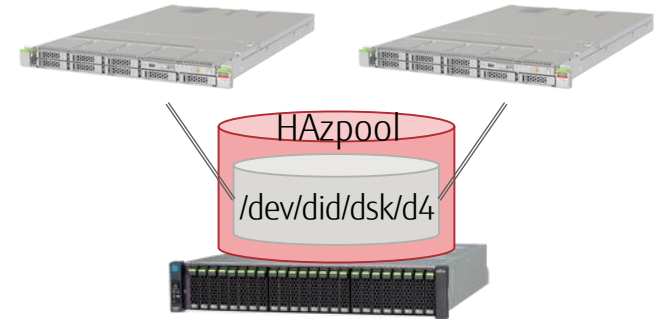
```
# clresource create -g resource-group-1 -t SUNW.HAStoragePlus ¥  
> -p FileSystemMountPoints=/export/global hasp1-rs
```

マウントしたファイルシステムを、オプションで指定

■ 作成手順

■ Step1：ZFSストレージプールの作成

- ・共有ディスクを指定して、ストレージプールを作成します。
 - ・DIDデバイスのパスでディスクを指定します。
 - ・本書では1つの共有ディスクを指定していますが、複数の共有ディスクでストレージプールを作成することもできます。



```
# zpool create HAzpool /dev/did/dsk/d4
```

■ Step2：リソースタイプの登録

- ・高可用性ローカルファイルシステムのリソースタイプ「SUNW.HAStoragePlus」を登録します。
 - ・初期状態で登録されているリソースタイプは、論理ホスト（SUNW.LogicalHostname）と共有IPアドレス（SUNW.SharedAddress）のみです。
 - ・これら以外のリソースを使用するときは、リソースタイプの登録が必要です。すでに、リソースタイプ「SUNW.HAStoragePlus」を登録済みの場合は、ここでの操作は不要です。次のStepへ進みます。

```
# clresourcetype register SUNW.HAStoragePlus
```

ファイルシステムの場合は、リソースタイプに「SUNW.HAStoragePlus」を登録

■ Step3：リソースを作成し、リソースグループに登録

- ・ `clresource create` コマンドを使用して、リソースを作成・登録します。
 - 作成したZFSストレージプールの名称をオプションで指定します。

書式：`clresource create -g` リソースグループ `-t` リソースタイプ
`-p ZpoolsSearchDir`=ストレージプールに使用するディスク（デバイスパス）の格納ディレクトリ
`-p Zpools`=ストレージプールの名称 リソース

```
# clresource create -g resource-group-1 -t SUNW.HAStoragePlus ¥  
> -p ZpoolsSearchDir=/dev/did/dsk -p Zpools=HAzpool hasp2-rs
```

作成したZFSストレージプールの名称を、オプションで指定

③リソースグループの起動（有効化）

リソースグループおよびリソースは、作成しただけではこれらを使用することはできません。
作成後、手動で起動させる必要があります。

■ 起動手順

- ・ clresourcegroup online コマンドを実行します。
 - ・ リソースグループは優先ノード（通常、ノードIDが小さいノード）側で起動します。
 - ・ リソースグループを起動すると、登録されたリソースが起動します。

```
# clresourcegroup online -M resource-group-1
```


■ リソース単位の操作

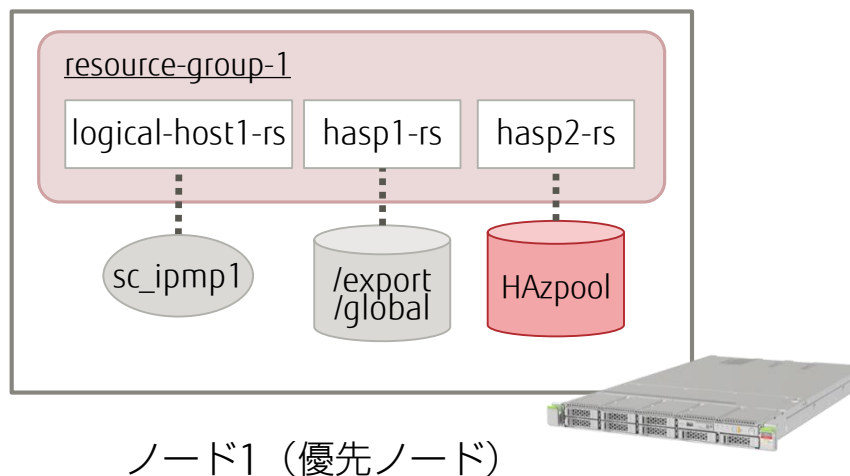
■ 有効化・無効化

- ・ 起動されているリソースのうち、一部のリソースを無効または有効にすることができます。
- ・ 無効化には、`clresource disable`コマンドを使用します。
 - 無効にするリソースを指定します。

```
# clresource disable logical-host1-rs
```

- ・ リソースの有効化（元に戻す場合）には、`clresource enable`コマンドを使用します。

```
clresource enable logical-host1-rs
```



■ ノード切替

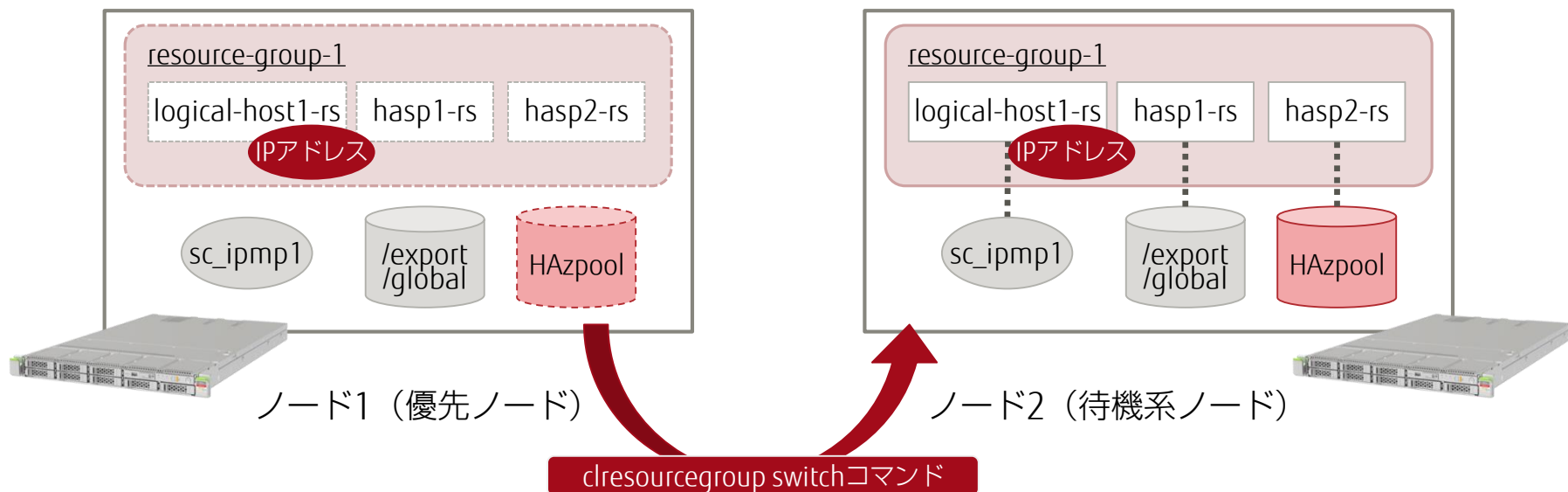
■ リソースグループ・リソースの切替

- 各リソースグループ・リソースはノードで障害が発生すると、自動で待機系ノードに切り替わります。ノード切替は手動で実行することもできます。

書式 : `clresourcegroup switch -n` リソースグループを有効にするノード リソースグループ

```
# clresourcegroup switch -n node2 resource-group-1
```

- ノード切替後、論理ホストのIPアドレスや高可用性ファイルシステムのストレージプールが待機系ノードに切り替わっていることを確認できます。
 - クラスタファイルシステム (hasp1-rs) はリソース切替が実行されても、両方のノードから使用可能です。



■ 停止

■ リソースグループ・リソースの停止

- ・ リソースグループを停止すると、登録されたリソースが停止します

```
# clresourcegroup offline resource-group-1
```

■ 状態確認

■ リソースグループ・リソースの情報確認

- ・ clresource status コマンドおよび clresourcegroup status コマンドを使用します。
- ・ 各リソース（グループ）の状態（起動 or 停止）や、ノードの情報が表示されます。

```
# clresourcegroup status
```

```
=== クラスタリソースグループ ===
```

グループ名	ノード名	中断	ステータス
-----	-----	-----	-----
resource-group-1	node1	No	Online
	node2	No	Offline

```
# clresource status
```

```
=== クラスタリソース ===
```

リソース名	ノード名	状態	ステータスメッセージ
-----	-----	-----	-----
hasp2-rs	node1	Online	Online
	node2	Offline	Offline
hasp1-rs	node1	Online	Online
	node2	Offline	Offline
logical-host1-rs	node1	Online	Online - LogicalHostname online.
	node2	Offline	Offline

7. アプリケーション／サービスの リソース登録

1. サーバの環境確認・OSの設定

2. ディスクの冗長化設定

3. Oracle Solaris Cluster のインストール

4. クラスタへのノードの登録

5. クラスタの初期設定

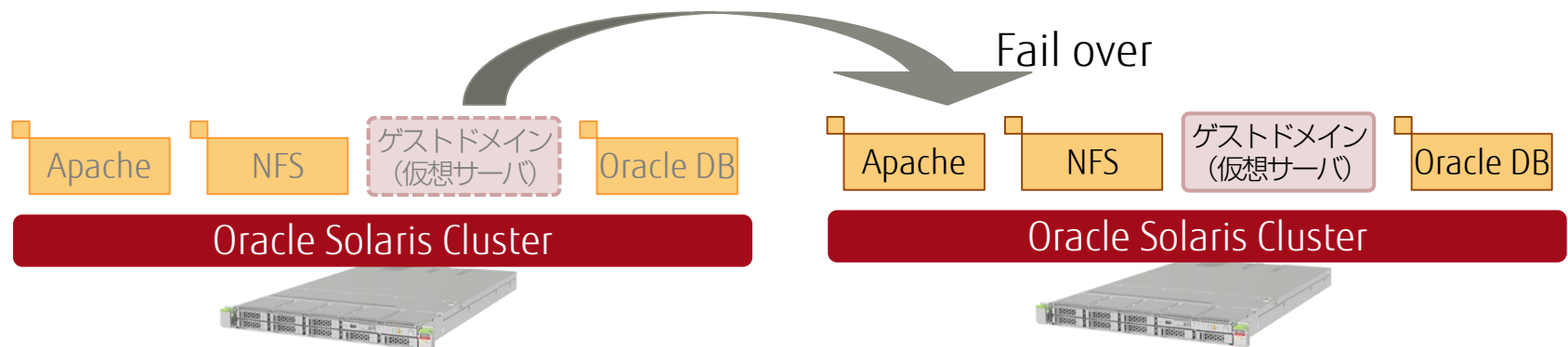
6. リソースの設定

7. アプリケーション／サービスのリソース登録



■ 業務アプリケーション／サービスのクラスタシステム

- ・ Solaris Cluster では、様々な業務アプリケーションやサービスなどをリソースとして登録し、障害を監視できます。
 - ・ これらのリソース登録をサポートするパッケージとして「データサービス」が用意されています。
- ・ 本書では、データサービスを活用し、例として以下4つの業務アプリケーション／サービスのリソース登録を実施します。
 - ① Apache (Webサービス)
 - ② NFSサービス
 - ③ Oracle VM Server for SPARC
 - 本番サーバでゲストドメイン (仮想サーバ) を起動しておき、障害時には待機系サーバでゲストドメインを起動させます。
 - ④ Oracle Database



■ リソース登録のポイント

※ ここでは、「ポイント」のみをピックアップしています。この順序どおりに実施する必要はありません。

■ データサービスのインストール

- ・登録するアプリケーション／サービスに応じたデータサービスのパッケージをインストールします（両方のノードで実行）。
 - Solaris Clusterのインストールパッケージと同様に、pkgコマンドを使用して、リポジトリサーバからパッケージをインストールします。

■ アプリケーション／サービスの設定

- ・リソース登録するアプリケーションやサービスを設定します（主に以下）。
 - 必要なパッケージのインストール（確認）
 - 環境設定ファイルの編集
 - SMF (Service Management Facility) の設定

■ リソースの登録

- ・使用するリソースタイプを登録（確認）します。
- ・clresource createコマンドでリソースを登録します。
 - 業務に必要なほかのリソースを依存リソースとして登録します。



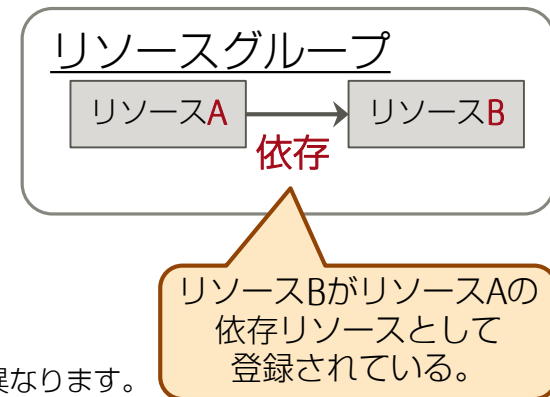
- ・上記はデータサービスを使用したリソース登録の大まかなながれを記述していますが、データサービスの種類によって、設定方法や留意事項（制限事項）は異なります。
- ・必ず、使用するデータサービスのマニュアル（Oracle社）を事前に確認してください。

業務アプリケーション／サービスを確実に正常起動させるために、リソースの依存関係を設定します。

■ 依存リソース

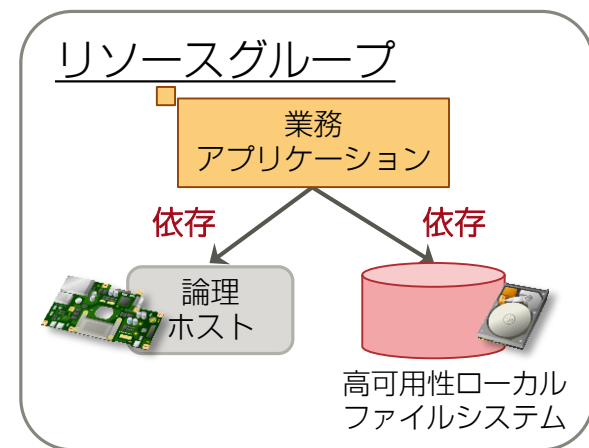
- ・ リソース間の依存関係を設定できます。
- ・ 複数のリソースで構成される業務システムに有効です。
- ・ 右図の例では、以下のように動作します。
 - 「リソースB ⇒ リソースA」の順にリソースを起動させます。
(リソースを停止させる際は、逆の順番となります)
 - リソースBが再起動または停止すると、
リソースAもそれに合わせて、再起動・停止します(※1)。

※1：リソースの依存関係設定プロパティには様々なものがあり、それぞれ動作が少し異なります。
詳細は、リソースプロパティ (r_properties) のマニュアルをご参照ください。



■ 代表的なリソースの依存関係

- ・ 業務アプリケーション（サービス）と「業務アプリケーションが使用する」I/O（ストレージ・ネットワーク）のリソースに対して、依存関係を設定します。
- ・ 主に、論理ホストや高可用性ローカルファイルシステムなどが依存リソースとなります。



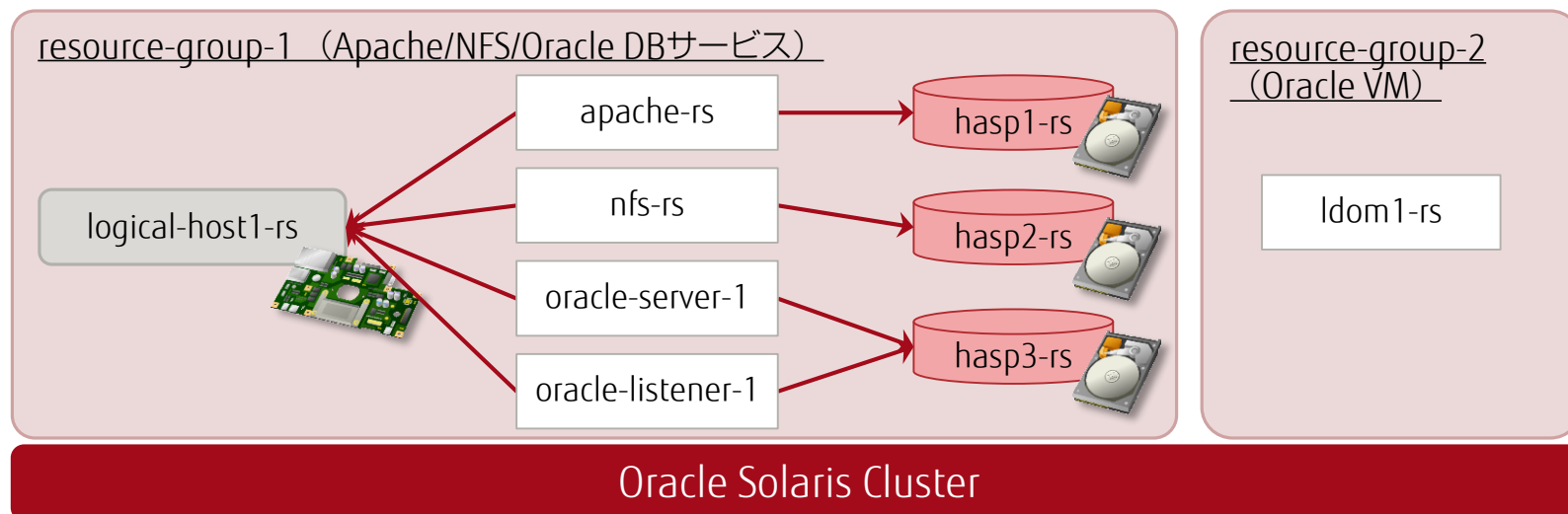
■ リソースグループ・リソース

■ Apache/NFS/Oracle Databaseサービス用のリソースグループ

- Apacheの依存リソース：クラスタファイルシステム (hasp1-rs) と論理ホスト (logical-host1-rs)
- NFSの依存リソース：高可用性ローカルファイルシステム (hasp2-rs) と論理ホスト
- Oracle Databaseの依存リソース：高可用性ローカルファイルシステム (hasp3-rs) と論理ホスト

■ Oracle VM用のリソースグループ

- 依存リソースはなし。ゲストドメインをリソースとして登録。



リソースグループと業務アプリケーション/サービス

- リソースグループは、通常、業務システムとして使用するアプリケーションやサービス（データサービス）ごとに作成します。
- 本書の場合、Apache、NFS、Oracle Databaseサービスで同一のネットワークリソース（論理ホスト）を使用するため、これらを同一のリソースグループに設定しています。

①Apacheのリソース登録 1/2

■ データサービスのインストール

```
# pkg install -v ha-cluster/data-service/apache
```

■ Apacheの設定

- Step1 : Apacheの起動・停止スクリプトファイル (apachectl) の編集
 - httpdデーモン、Apache環境設定ファイル、PIDFILEのPATHを指定
- Step2 : 環境設定ファイル (httpd.conf) の編集
 - 接続IPアドレスを論理ホスト、コンテンツ格納領域にクラスタファイルシステムを指定
- Step3 : CGI用のディレクトリをクラスタファイルシステム上にコピー
 - /export/global はクラスタファイルシステムのマウントポイント

```
# cp -pr /var/apache2/2.2/cgi-bin /export/global
```

①Apacheのリソース登録 2/2

■ リソースの登録

■ Step1 : Apacheのリソースタイプを登録

```
# clresourcetype register SUNW.apache
```

■ Step2 : リソースを登録

- クラスタファイルシステムを依存リソース (Resource_dependencies_offline_restart) として登録

※ 論理ホストは依存リソースとして自動で登録されます (指定する必要はありません) 。

```
# clresource create -g resource-group-1 ¥  
> -t SUNW.apache -p Bin dir=/usr/apache2/2.2/bin ¥  
> -p Resource_dependencies_offline_restart=hasp1-rs ¥  
> -p Port_list=80/tcp apache-rs
```

②NFSのリソース登録 1/3

■ データサービスのインストール

```
# pkg install -v ha-cluster/data-service/nfs
```

■ NFSの設定 1/2

■ Step1：ネームサービスのSMFプロパティの設定確認

```
# svccfg -s svc:/system/name-service/switch listprop config/host  
config/host astring "cluster files"  
# svccfg -s svc:/system/name-service/switch listprop config/rpc  
config/rpc astring files
```

■ Step2：NFSサービス用のファイルシステムの作成

- 高可用性ローカルファイルシステム上に、NFS用のファイルシステムを作成します。

※ 本書では、管理用と共有用の2種類のファイルシステムを作成しています。

```
# zfs create HAzpool/nfs  
# zfs create HAzpool/nfs/admin  
# zfs create HAzpool/nfs/data
```

NFS管理用ファイルシステム

NFS共有用ファイルシステム

■ NFSの設定 2/2

■ Step3：共有設定

- ・「SUNW.nfs」ディレクトリ（Solaris ClusterにおけるNFSの設定ファイル格納ディレクトリ）を作成します。
 - 管理用領域に作成します。
（NFSの設定情報が含まれるため、共有領域以外のディレクトリ配下に作成します）

```
# mkdir /HAzpool/nfs/admin/SUNW.nfs
```

- ・shareコマンドでNFSの共有設定を行います。
 - shareコマンドのオプションは、環境に合わせて適切なものを選択してください。

```
# share -F nfs -o anon=0,sec=sys,rw /HAzpool/nfs/data
```

- ・dfstabファイルによる共有設定ファイルを作成します。
 - SUNW.nfsディレクトリ配下に、「dfstab.nfs-rs」ファイルを作成します。
 - 実行したshare コマンドと同一の設定を記述します。

```
# vi /HAzpool/nfs/admin/SUNW.nfs/dfstab.nfs-rs
```

②NFSのリソース登録 3/3

■ リソースの登録

■ Step1： NFSのリソースタイプを登録

```
# clresourcetype register SUNW.nfs
```

■ Step2： リソースグループの設定

- PathPrefixプロパティをNFS管理領域に設定します。

```
# clresourcegroup set -p PathPrefix=/HAzpool/nfs/admin resource-group-1
```

■ Step3： リソースを登録

- 高可用性ローカルファイルシステムを、依存リソース (Resource_dependencies_offline_restart) として登録します。

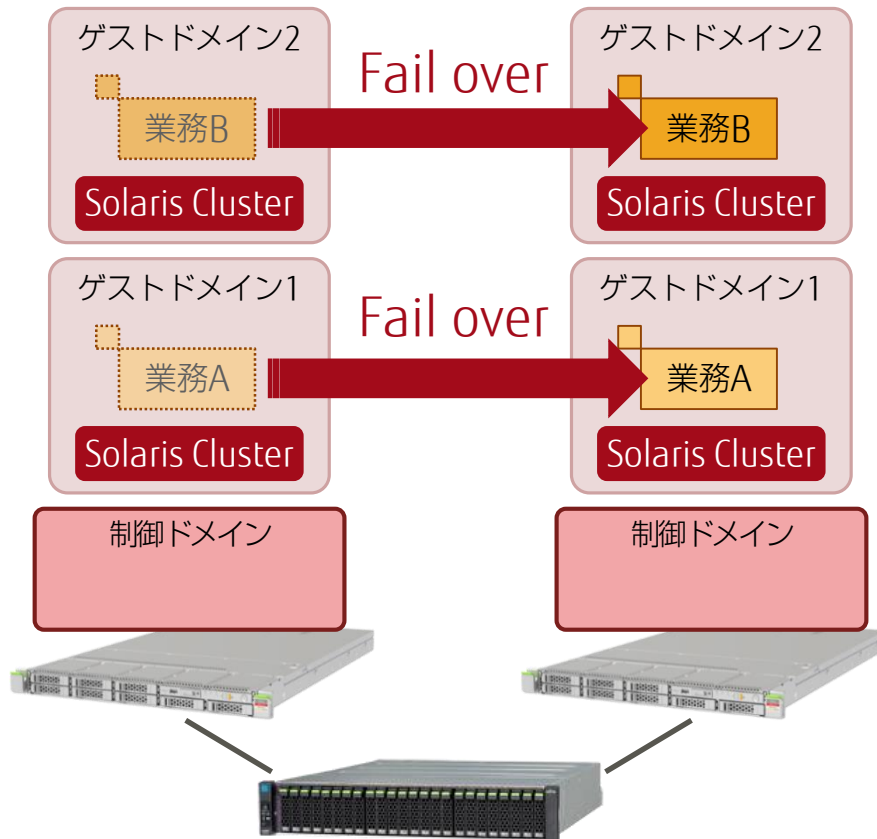
※ ここでも依存リソースに論理ホストは指定不要です。（自動で依存設定されます）

```
# clresource create -g resource-group-1 -t SUNW.nfs ¥  
> -p Resource_dependencies_offline_restart=hasp2-rs nfs-rs
```

Oracle VM環境（ゲストドメイン）で、Solaris Clusterによって構築できるクラスタシステムには、以下の2種類があります。

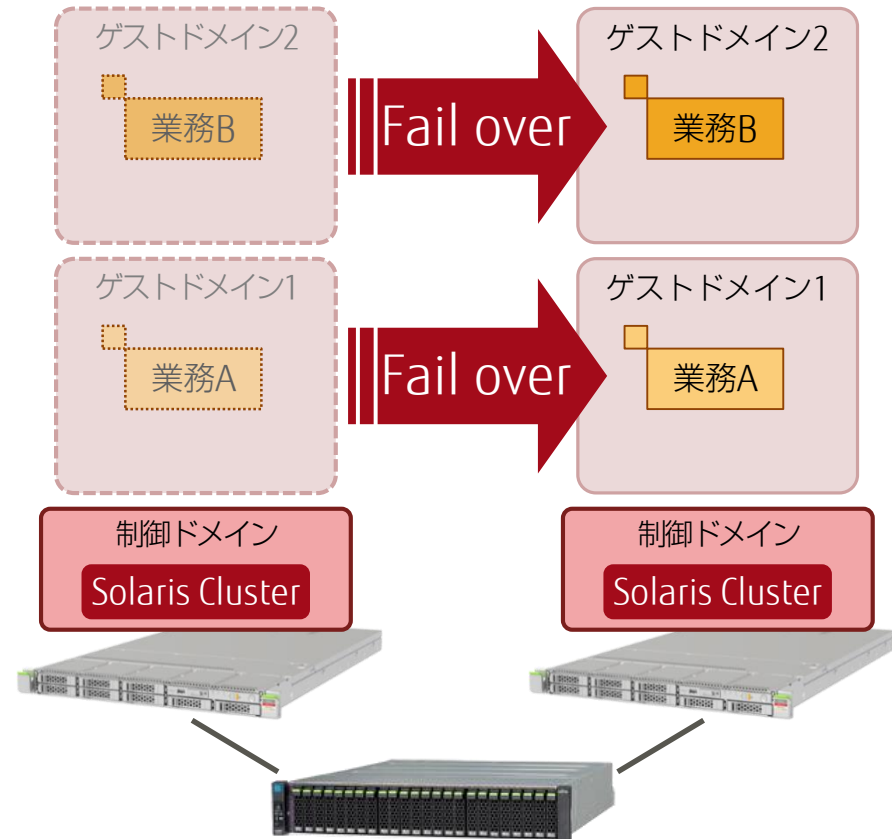
■ ゲストドメイン間クラスタ

- ・ ゲストドメインにSolaris Clusterを導入します。
- ・ 業務（アプリケーション）単位でノードを切り替えます。



■ HA for Oracle VM Server for SPARC

- ・ 制御ドメインにSolaris Clusterを導入します。
- ・ ドメイン単位でノードを切り替えます。



■ Oracle VM環境におけるクラスタシステムの機能比較

項目	ゲストドメイン間クラスタ	HA for Oracle VM Server for SPARC
構成	<ul style="list-style-type: none"> • <u>ゲストドメイン</u>にSolaris Clusterをインストール • ゲストドメイン間でクラスタを構成 	<ul style="list-style-type: none"> • <u>制御ドメイン</u>にSolaris Clusterをインストール • 制御ドメイン間でクラスタを構成
切替動作	<ul style="list-style-type: none"> • <u>ゲストドメイン上のリソースグループ（業務アプリケーションやI/Oリソース）などの異常発生時にノードを切替</u> • あらかじめ起動している待機系サーバのゲストドメインに、リソースグループを引き継ぐ 	<ul style="list-style-type: none"> • <u>ゲストドメインが停止したとき（※1）に、ノードを切替</u> • 待機系サーバで同一構成のゲストドメインを起動させて、同時に業務も引き継ぐ
Solaris Clusterのライセンス費用	<ul style="list-style-type: none"> • <u>ゲストドメインに割り当てたCPUコア数分のライセンスが必要</u> （制御ドメインの分のライセンスは不要） （※2）（※3） 	<ul style="list-style-type: none"> • <u>制御ドメインに割り当てたCPUコア数分のライセンスが必要</u> （ゲストドメインの分のライセンスは不要） （※2）
特長	<ul style="list-style-type: none"> • アプリケーション異常を含めた、<u>精度の高い障害検出環境</u>を容易に構築可能 	<ul style="list-style-type: none"> • <u>低コストのクラスタシステム</u>を構築可能

※1： 別途、簡易監視プログラム（Plugin_probe）を作成することで、アプリケーションレベルの障害検出も可能となります。

※2： Solaris Clusterをインストールするドメインが、Oracle 社が定める「Hard Partitioning」の要件を満たしていることが必要です。

※3： 制御ドメインにはクラスタ環境を構築しないことを想定しています。



• 本書では、HA for Oracle VM Server for SPARCを構築します。

■ 仮想化構成（ディスク、ネットワーク）

■ 概要

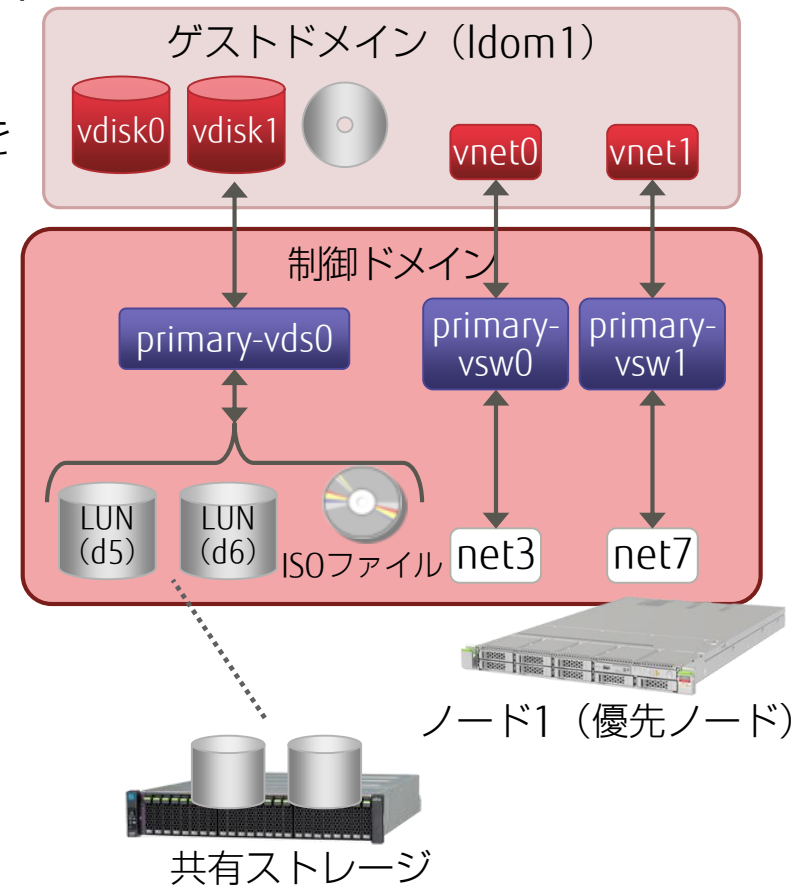
- ・ Oracle VMの仮想環境（ゲストドメイン）を1つ構築します。
- ・ 制御ドメインにはゲストドメインに必要な仮想サービスをすべて作成します。

■ 仮想ディスクサービス（vds）

- ・ 以下を仮想ディスクとして、ゲストドメインに割り当てます。
 - 2つの共有ストレージのLUN
 - Solarisのテキストインストーラ（ISOファイル）

■ 仮想スイッチサービス（vsw）

- ・ 未使用のLANポート（net3, net7）より仮想LANポート（vnet）を割り当てます。



※ 仮想コンソールなどの記載は省略しています。



HA for Oracle VM の構築のポイント

- ・ 両方のノードの制御ドメインに、同一の仮想サービスを作成します。
- ・ ゲストドメインは優先ノードでのみ構築します。

③Oracle VMのリソース登録 -登録方法- 1/4

■ データサービスのインストール

```
# pkg install -v ha-cluster/data-service/ha-lldm
```

■ Oracle VM の設定 1/2

■ Step1：制御ドメインの構築（サービスドメインの設定）

- ・ 非クラスタ環境と同様に、サービスドメインの設定を行います。



- ・ Oracle VMの基本的な操作や構築手順については、以下の資料もご参照ください。

『Oracle VM Server for SPARC を使ってみよう』

<http://www.fujitsu.com/jp/sparc-technical/document/solaris/index.html#ovm-use-11>

- ・ Oracle 社の「Hard Partitioning」に準拠させる場合、max-coresパラメータを設定します。
 - max-coresパラメータを設定する場合、設定対象のドメインを停止させる必要があります。
 - 制御ドメインの場合は、ldm start-reconfコマンドで遅延再構成状態にしてから、パラメータを設定します。

```
# ldm start-reconf primary
# ldm set-core 4 primary
# ldm set-domain max-cores=4 primary
```

- ・ failure-policyパラメータを設定します。
 - 本設定によって、制御ドメイン障害時にすべてのドメインの再起動を実行します。

```
# ldm set-domain failure-policy=reset primary
```

■ Oracle VM の設定 2/2

■ Step2：ゲストドメインの構築 ※優先ノードのみで実行

- ・ 非クラスタ環境と同様の設定（主に下記）を行います。
 - CPU・メモリリソースの割り当て
 - 仮想ネットワーク・仮想ディスク・仮想コンソールの割り当て
 - 基本的なパラメータ（auto-bootやboot-deviceなど）の設定
- ・ master パラメータを設定します。
 - 制御ドメインを依存するドメインとして登録します。

```
node1# ldm set-domain master=primary ldom1
```

- ・ ゲストドメインをバインド・起動させます。

```
node1# ldm bind ldom1
node1# ldm start ldom1
```

■ リソースの登録

- Step1 : Oracle VM (HA for Oracle VM Server for SPARC) のリソースタイプを登録

```
# clresourcetype register SUNW.ldom
```

- Step2 : HA for Oracle VM Server for SPARC 用のリソースグループを作成

```
# clresourcegroup create resource-group-2
```

- Step3 : リソースを登録

書式 : clresource create -g リソースグループ -t リソースタイプ -p オプション リソース名
・オプション
-p Domain_name=ゲストドメインの名称
-p Migration_type=(MIGRATE | NORMAL)

```
# clresource create -g resource-group-2 -t SUNW.ldom ¥  
> -p Domain_name=ldom1 -p Migration_type=NORMAL ldom1-rs
```

※ Migration_typeオプションについて
パラメータに「MIGRATE」を指定すると、高速なノード切替（障害時のゲストドメインの復旧）が可能になりますが、Oracle社の「Hard Partitioning」に準拠しないため、本書では「NORMAL」を指定しています。

③Oracle VMのリソース登録 -登録方法- 4/4

■ 動作確認

■ Step1：リソースグループの起動

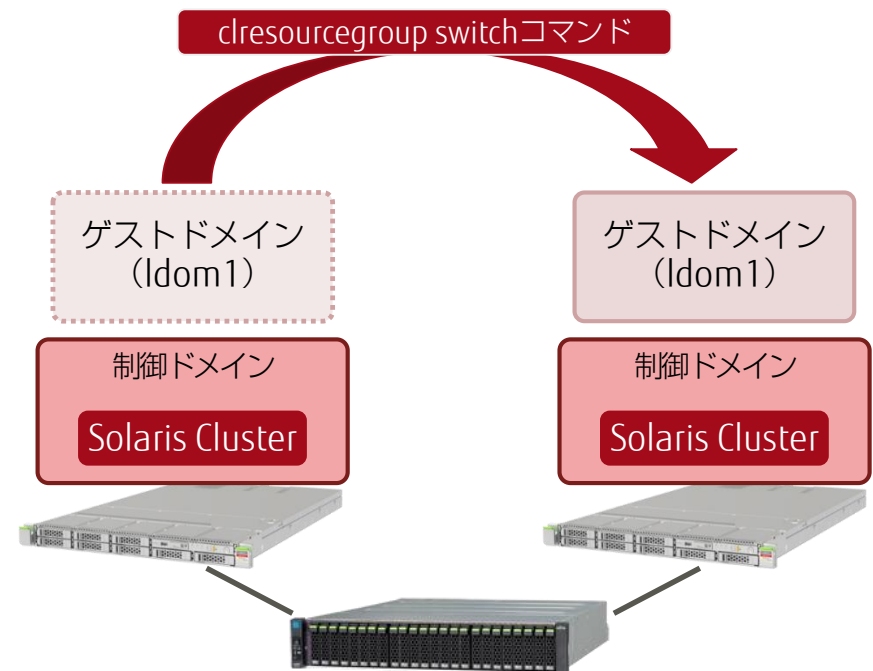
```
# clresourcegroup online -eM resource-group-2
```

■ Step2：リソースグループのノード切替

- ・ ゲストドメインを待機系サーバ側で起動させます。

```
# clresourcegroup switch -n node2 resource-group-2
```

- HA for Oracle VM では、ゲストドメインを制御ドメインのアプリケーションと同様に管理します。
(リソース・リソースグループによる管理)
- 稼動系サーバの障害によるノード切替と手動によるノード切替 (clresourcegroup switch) のどちらの場合も、ゲストドメインは一定時間停止します。
(再起動が発生します)

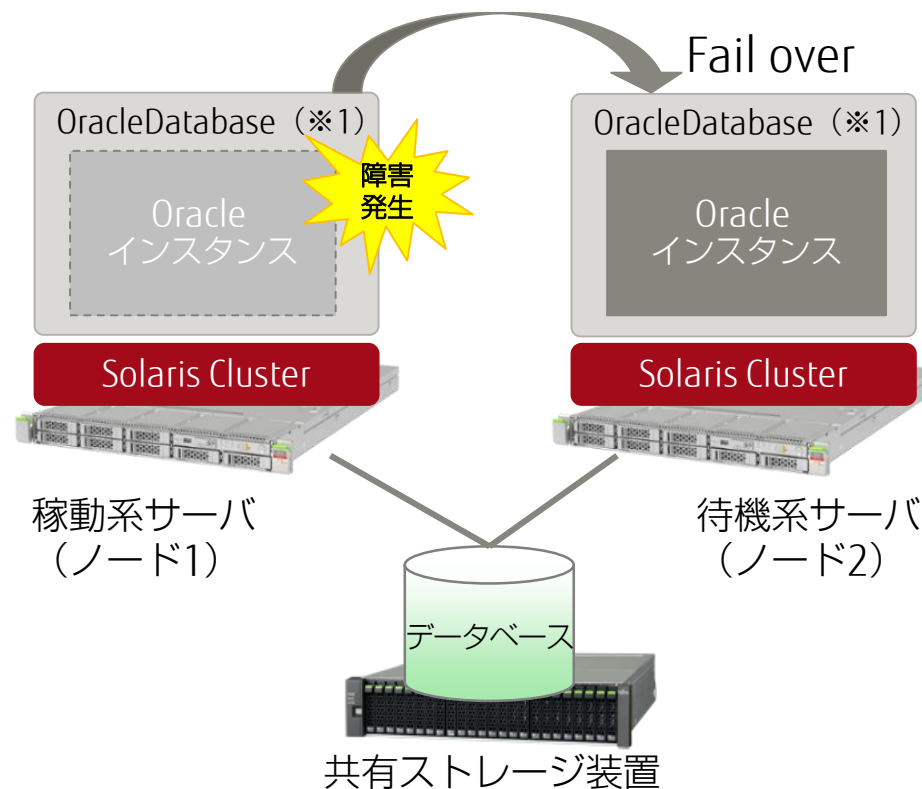


④Oracle DBのリソース登録 -Oracle DB環境のSolaris Cluster-

Solaris ClusterによるOracle DBのフェイルオーバーシステム「HA for Oracle Database」を構築します。

■ HA for Oracle Database

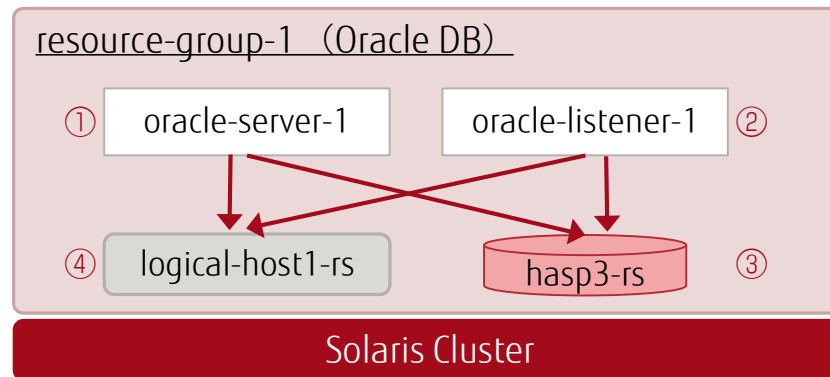
- ほかのアプリケーション同様、データサービスをインストールしてリソース登録を行います。



※1：Oracle Database 12c Release1を使用します。

■ 登録するリソース

リソース		内容
Oracle DB アプリケーション リソース	① oracle-server-1	DBサーバ側でSQLのデータ操作を行う、サーバプロセス用のリソース
	② oracle-listener-1	DBサーバ側でクライアントからの接続要求を仲介する、リスナープロセス用のリソース
関連リソース	③ hasp3-rs	高可用性ローカルファイルシステム
	④ logical-host1-rs	論理ホスト



ノード1 (優先ノード)

■ データベースの構成

構成物	内容
ディスク	共有ストレージ（高可用性ローカルファイルシステム）のリソースを使用
ネットワーク	論理ホストのリソースを使用

■ 環境構築のながれ

データサービスのインストール

Oracle DBのインストール、データベースの構築

Oracle DBのリソース登録

- ・ Oracle DBの設定
- ・ リソースの登録
- ・ リソース登録後のインスタンス接続確認



- Oracle DBの基本的な操作や構築手順はOracle社のマニュアルも参考にしてください。
 - Oracle Solaris Cluster データサービス (Oracle Database用)
https://docs.oracle.com/cd/E62281_01/html/E62264/index.html

■ データサービスのインストール

■ 以下2種類のパッケージが必要になります。

- ・ ha-cluster/data-service/oracle-database … (a)
- ・ ha-cluster/library/ucmm … (b)

```
# pkg install ha-cluster/data-service/oracle-database ha-cluster/library/ucmm
```

※ (b) は「ha-cluster-framework-full」パッケージグループの依存パッケージです。「ha-cluster-framework-full」をインストールしている場合は (b) はインストールされているため、(a) のみをインストールしてください。

■ Oracle DBのインストール、データベースの構築

■ Oracle DB用のユーザーを作成

- ・ Oracle DBのインストールや操作に使用するグループとユーザーを新規に作成します。
 - 作成グループ：oinstall (Oracleインベントリ用)、dba (DBインストール用)
 - 作成ユーザー：oracle (Oracle Databaseの管理ユーザー)

■ ディレクトリの作成

- ・ Oracle DBのインストールに必要な2つのディレクトリを作成します。
 - ベースディレクトリ (ORACLE_BASE) : /u01/app/oracle
 - インベントリ・ディレクトリ (orainventory) (※1) : /u01/app/orainventory

※1：インストールしたOracle製品の情報などを保持するためのディレクトリ

■ 高可用性ローカルファイルシステムのリソース (hasp3-rs) を作成

- ・ 共有ディスク (d7) を使用し、リソース「hasp3-rs」を作成します。
- ・ 作成したリソース「hasp3-rs」は、Oracleデータベース領域として使用します。

■ Oracle DBのインストール、データベースの構築

- ・ 事前準備が完了したらOracle DBをインストールし、データベースを構築します (本書では省略)。



・インストールについてはOracle社のマニュアルも参考にしてください。
『Oracle DBインストレーション・ガイド 12cリリース1 (12.1) for Oracle Solaris』
https://docs.oracle.com/cd/E49329_01/install.121/b71320/toc.htm

■ Oracle DBの設定 1/2

- Step1：リスナー構成ファイルとローカル・ネーミング・パラメータを更新
 - ・「HOST」に論理ホストのホスト名 (logical-host1) を設定します。

リスナー構成ファイル (listener.ora) の更新

```
# vi /u01/app/oracle/product/12.1.0/dbhome_1/network/admin/listener.ora
~ (省略) ~
LISTENER =
  (DESCRIPTION_LIST =
    (DESCRIPTION =
      (ADDRESS = (PROTOCOL = TCP) (HOST = logical-host1) (PORT = 1521))
    )
  )
```

ローカル・ネーミング・パラメータ (tnsnames.ora) の更新

```
# vi /u01/app/oracle/product/12.1.0/dbhome_1/network/admin/tnsnames.ora
~ (省略) ~
LISTENER_ORCL =
  (ADDRESS = (PROTOCOL = TCP) (HOST = logical-host1) (PORT = 1521))

ORCL =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP) (HOST = logical-host1) (PORT = 1521))
  )
~ (省略) ~
```

■ Oracle DBの設定 2/2

■ Step2 : control_filesの設定を変更

- ・ OracleユーザーでSQL*Plusを起動し、control_fileの設定を変更します。
 - node2のパラメーターがデフォルト値になっているので、SQL*Plusでnode1と同じ値に変更してください。

node1のcontrol_files確認方法

```
oracle@node1:~$ sqlplus /nolog
SQL> connect / as sysdba
SQL> show parameter control_files
NAME                                TYPE                                VALUE
-----                                -                                -
control_files                        string                             /oracledb/data/oradata/orcl/control01.ctl
```

SQL*Plusを起動


control_filesの
設定値を確認

■ Step3 : Oracle DB操作に必要なファイルをnode2へコピー

- ・ ノード1の「ORACLE_BASE」配下を圧縮し、ノード2上にコピーして展開します。

■ Step4 : /optの所有者変更

- ・ /optディレクトリの所有者を「bin」に設定します。
 - デフォルトでは「sys」になっています。必要に応じて変更してください。

 データベースを作成する際、作成モードを「デフォルトの構成」にすると、Oracle DBの動作に必要なファイルは一方のノードにのみ作成されます（本書の場合はノード1）。その場合、別途ノード1からノード2へファイルのコピーが必要になります。

■ リソースの登録 1/2

■ Step1：データサービスのリソースタイプを登録

- ・ Oracle DBでは、2種類のリソースタイプを登録します。

```
# clresourcetype register SUNW.oracle server  
# clresourcetype register SUNW.oracle listener
```

■ リソースの登録 2/2

■ Step2：リソースグループにOracle DBアプリケーションを追加

- ・高可用性ローカルファイルシステムを依存リソースとして登録します。

書式： clresource create -g リソースグループ -t リソースタイプ -p オプション リソース名

- ・オプション

-p Restart_type=障害発生時にリソースグループごと再起動させるか、リソースを再起動させるかを指定

-p Listener_name= "listener.ora" ファイルに記載したリスナーインスタンス名を設定

※ 上記のほか、HA for Oracle DB拡張プロパティを設定します。設定できるオプションは、Oracle Grid Infrastructureの有無によって異なります。ここでは、「Oracle Grid Infrastructureなし」で登録を行います。

Oracle DBサーバリソースの登録

```
# clresource create -g resource-group-1 -t SUNW.oracle_server -p ¥  
> Alert_log_file=/u01/app/oracle/diag/rdbms/orcl/orcl/trace/alert_orcl.log -p ¥  
> Connect_string=oracle/oracle00 -p Oracle_sid=orcl -p ¥  
> Oracle_home=/u01/app/oracle/product/12.1.0/dbhome_1 -p ¥  
> Restart_type=RESOURCE GROUP RESTART ¥  
> -p Resource dependencies offline restart=hasp3-rs oracle-server-1
```

Oracle DBリスナーリソースの登録

```
# clresource create -g resource-group-1 -t SUNW.oracle_listener -p ¥  
> Listener name=listener -p Oracle home=/u01/app/oracle/product/12.1.0/dbhome_1 ¥  
> -p Resource dependencies offline restart=hasp3-rs oracle-listener-1
```



・HA for Oracle DB拡張プロパティについては「[《参考》HA for Oracle DB拡張プロパティの設定](#)」参照。

■ リソース登録後のインスタンス接続確認

- ・ oracleユーザーでノード1にログインし、Oracle DBインスタンスに接続できることを確認します。

```
# su - oracle
$ sqlplus oracle/oracle00@orcl

SQL*Plus: Release 12.1.0.2.0 Production on Mon Feb 27 13:36:37 2017
Copyright (c) 1982, 2014, Oracle. All rights reserved.
Last Successful login time: Mon Feb 27 2017 13:34:48 +09:00

Connected to:
Oracle DB 12c Enterprise Edition Release 12.1.0.2.0 - 64bit Production
With the Partitioning, OLAP, Advanced Analytics and Real Application Testing options

SQL>
```

《参考》 HA for Oracle DB拡張プロパティの設定

HA for Oracle DBのリソース登録時に、拡張プロパティの設定を行います。

- ・ Oracle Grid Infrastructure (※1) の有無により、リソース登録時に設定可能な拡張プロパティが異なります。

※1： Oracle Grid Infrastructureは、Oracle Automatic Storage Management (Oracle ASM)、Oracle Clusterware、Oracle Restartなどから構成される、Oracle Databaseの可用性を高めるソフトウェアです。

拡張プロパティ	Oracle Grid Infrastructure あり	Oracle Grid Infrastructure なし
Alert_log_file (サーバメッセージログのパス)		○
Connect_string (データベースの接続方法)		○
Db_unique_name (データベース名として指定する任意の名前)	○	○
Oracle_home (Oracle DBホームディレクトリのパス)	○	○
Oracle_sid (Oracle DBシステム識別子)	○	○



・本書では、「Oracle Grid Infrastructureなし」で登録を行います。

■ Oracle Solaris Cluster 4.3 の主なデータサービス一覧

■ Web

- ・ Apache, Apache Tomcat, Oracle Web Tier, Oracle Traffic Director

■ AP(Application)

- ・ Oracle E-Business Suite, Oracle WebLogic Server, PeopleSoft Application Server

■ Database

- ・ Oracle Database, Oracle Real Application Clusters, MySQL, PostgreSQL

■ OS / サーバ仮想化

- ・ DHCP, DNS, NFS, Samba, Oracle Solaris ゾーン, Oracle VM Server for SPARC



データサービスについて

- 本ページに記述しているデータサービスは一部です（すべてではありません）。
- 各データサービスにおいて、ソフトウェアのバージョンやハードウェアの構成・設計に制限があります。詳細はOracle社のデータサービスのマニュアルをご確認ください。

参考情報

- Oracle Solaris Cluster Manager の導入

Oracle Solaris Cluster Managerの導入 1/3

Oracle Solaris Cluster Managerを導入することにより、Web-GUIベースの操作や管理が可能になります。

■ Oracle Solaris Cluster Managerの機能概要 1/2

- ・ Solaris ClusterのノードにWebブラウザでアクセス（HTTPアクセス）できます。
- ・ 以下をはじめとする、クラスタの管理・設定がGUIで実行できます。
 - ・ クラスタの基本的な設定（Quorum DeviceやIPMPなど）
 - ・ ノードやリソース（グループ）の監視・状態表示
 - ・ リソースやリソースグループの作成
 - ・ アプリケーション用データサービス、リソースの作成

リソースの状態表示

リソースの作成

名前	ステータス	タイプ
1 apache-rs	オンライン	SUNWapache:4.2
2 hasp1-rs	オンライン	SUNW.HASStoragePl...
3 hasp2-rs	オンライン	SUNW.HASStoragePl...
4 logical-host1-rs	オンライン	SUNW.LogicalHostn...
5 nfs-rs	オンライン	SUNWnfs:3.3

タスク: Oracle Solaris Cluster リソースの構成
高可用性ストレージリソースの構成

Step

1. 前提条件の検証
2. 高可用性ストレージの場所の選択
3. ノードの選択

ノードの選択
高可用性ストレージ

使用可能:

Oracle Solaris Cluster Managerの導入 2/3

■ Oracle Solaris Cluster Managerの機能概要 2/2

- ・ウィザードを使用して、リソースを作成できます。
(Oracle Solaris Cluster 4.3 新機能)

The main screenshot shows the Oracle Solaris Cluster Manager interface. In the 'タスク' (Tasks) section, 'アプリケーション用データサービスの構成' (Configure Application Data Services) is highlighted. Below it, a list of resources is shown: Oracle Database, Oracle Real Application Clusters, Oracle VM Server for SPARC, Oracle Solaris ソーン, and Oracle WebLogic Server. To the right, 'Oracle Solaris Cluster リソースの構成' (Configure Oracle Solaris Cluster Resources) is also visible, showing '論理ホスト名' (Logical Host Name) and 'ノードごとの論理ホスト名' (Logical Host Name per Node).

Three detailed wizard windows are shown, all titled 'タスク: アプリケーション用データサービスの構成 - HA for Oracle Database' (Task: Configure Application Data Services - HA for Oracle Database).

- Top Window:** Shows the 'ステップ' (Steps) section with '1. 前提条件の検証' (Verify Prerequisites) selected. The '前提条件の検証' (Verify Prerequisites) section lists conditions for the wizard. The 'ヘルプ' (Help) section provides additional information.
- Middle Window:** Shows the 'ステップ' (Steps) section with '2. Oracle Database の場所の選択' (Select Oracle Database Location) selected. The 'Oracle Database の場所の選択' (Select Oracle Database Location) section shows a list of locations. The 'ヘルプ' (Help) section provides additional information.
- Bottom Window:** Shows the 'ステップ' (Steps) section with '3. ノードの選択' (Select Nodes) selected. The 'ノードの選択' (Select Nodes) section shows a list of nodes. The 'ヘルプ' (Help) section provides additional information.

■ 導入方法

■ ha-cluster-fullパッケージグループをインストールしている場合

- ・ 上記パッケージグループには、Solaris Cluster Managerの
パッケージが含まれているため、以降のインストール操作は不要です。
- ・ クラスタのパッケージグループについては、
「[《参考》 Solaris Clusterのパッケージグループ 1/2](#)」 をご参照ください。

■ ha-cluster-full パッケージグループをインストールしていない場合

- ・ Solaris Cluster Managerのパッケージ（ha-cluster/system/manager）を
別途インストールします。

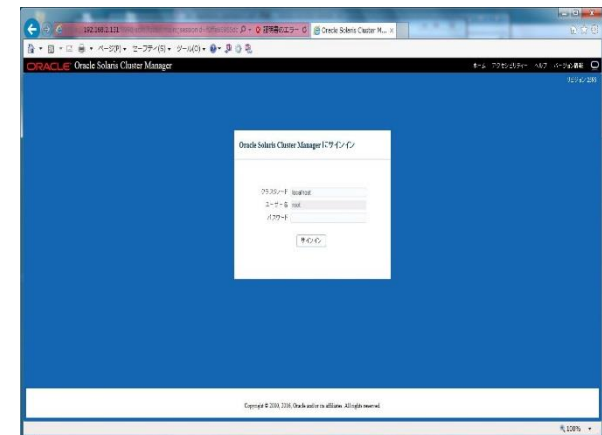
■ Oracle Solaris Cluster Managerへの接続方法

- ・ WebブラウザでノードのIPアドレスを指定してアクセスします。

`https://ノードのIPアドレス[ホスト]:8998/scm`

- ・ rootユーザーのパスワードを入力して、
ログインします。

※ 一般ユーザーではログインできません。



付録


Oracle Solaris Clusterの導入にあたっては、Oracle社の情報もご確認ください。

マニュアル : Oracle Solaris Cluster 4.3 Documentation (Oracle社)
http://docs.oracle.com/cd/E62281_01/

Oracle Solaris Cluster製品情報 (Oracle社)
<http://www.oracle.com/jp/solaris/cluster/overview/index.html>

■ SPARC/Solarisの構築に役立つドキュメントが満載

- ハイパーバイザベースの仮想化 :
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 (Zettabyte 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/>

版数	更新日時	更新内容
初版	2015年5月	新規作成
第2.0版	2017年9月	レイアウトデザインと構成を更新 Oracle Solaris 11.3、Solaris Cluster 4.3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

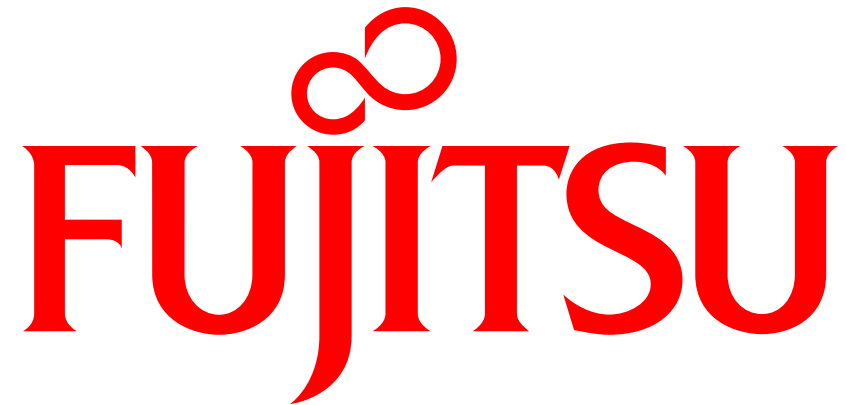
- ・本コンテンツについて、当社は、その正確性、商品性、ご利用目的への適合性等に関して保証するものではなく、そのご利用により生じた損害について、当社は法律上のいかなる責任も負いません。本コンテンツは、予告なく変更・廃止されることがあります。

■ 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。



shaping tomorrow with you