

# RAC on VirtualBox 構築ガイド

## Oracle VM VirtualBox を用いた Oracle Real Application Clusters (RAC) 12c Release 1 環境の構築

**ORACLE®**

作成日 : 2013 年 10 月 1 日

更新日 : 2014 年 3 月 5 日

バージョン : 1.2

## 目次

1.	はじめに	4
1.1	対象読者	4
1.2	関連文書	5
1.3	省略および表記規則	5
2.	概要	7
2.1	Oracle Database 12c Release 1 の新機能	7
2.2	ハードウェア	9
2.3	ソフトウェア	9
2.4	ネットワーク	9
3.	Oracle VM VirtualBox のインストールと設定	11
3.1	Oracle VM VirtualBox のインストール	11
3.2	機能拡張パッケージの追加インストール	18
3.3	インストール後の設定	22
3.4	仮想マシンの作成	24
4.	Oracle Linux 6 のインストールと再起動後における設定	29
4.1	インストールの事前準備	29
4.2	Oracle Linux 6 のインストール	35
4.3	インストール後の設定	51
4.4	DNS サーバー用の仮想マシンの作成	64
5.	インストール前の事前準備	72
5.1	仮想マシンの準備	72
5.2	oracle-validated-verify の実行	78
5.3	OS グループ、OS ユーザー、およびディレクトリの作成	78
5.4	ハードウェア要件とメモリの確認	81
5.5	ネットワーク要件の確認	83
5.6	ソフトウェア要件の確認	86
5.7	システム時刻同期の設定	86

5.8	cvuqdisk パッケージのインストール .....	89
5.9	Intelligent Platform Management Interface (IPMI) の設定 .....	89
5.10	Secure Shell (SSH) の構成 .....	89
5.11	環境変数とリソース制限の設定 .....	89
5.12	記憶域の確認と準備 .....	90
5.13	node2 用の仮想マシンの作成 .....	102
6.	Oracle Grid Infrastructure インストールおよび構成 .....	116
7.	Oracle Database のインストールと RAC データベースの作成 .....	138
7.1.	Oracle Database のインストール .....	138
7.2.	ASMCA を利用した新規 ASM ディスク・グループの作成 .....	148
7.3.	DBCA を利用した RAC データベースの作成 .....	151
8.	インストール後の確認と設定 .....	160
8.1	環境変数の設定 .....	160
8.2	SCAN を使用した SQL*Plus からの接続 .....	162
8.3	Oracle Enterprise Manager Database Express への接続 .....	164
Appendix 1.	DNS サーバーの準備 .....	166
Appendix 2.	CDB と PDB の基本操作 .....	169
Appendix 3.	Oracle ASM プロキシ・インスタンスの管理 .....	174

## 1. はじめに

本ガイドでは、Oracle VM VirtualBox を用いて Oracle Real Application Clusters (RAC) 環境を構築するための手順を説明します。RAC は共有ディスク、共有キャッシュ型のクラスタ・データベースであり、可用性と拡張性に富んだ環境を提供します。主に複数のサーバー(ノード)を使用して環境を構築しますが、本ガイドの構成では仮想化ソフトウェアである Oracle VM VirtualBox を用いて 1 台の物理マシンで環境を構築しています。

次に、環境構成についての概要図を示します。1 台のマシン (Windows OS) 上に Oracle VM VirtualBox を用いて仮想マシンを 3 台作成し、各仮想マシンに OS として Oracle Linux をインストールします。さらに 2 台の仮想マシンを使用して RAC の構成に必要なソフトウェアとして Oracle Grid Infrastructure および Oracle Database をインストールして 2 ノード RAC 環境を構築します。

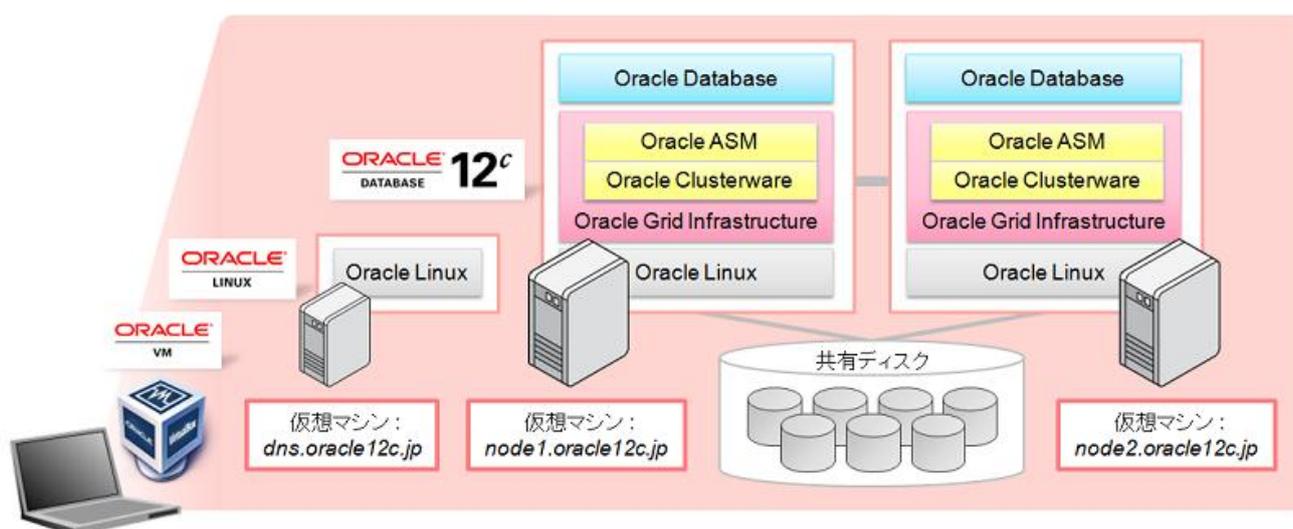


図 1: 本ガイドにおける環境構成

本ガイドで紹介する手順および構築する環境は、RAC の構築手順や動作確認を行う機能評価用の検証環境を手早く構築することを目的としています。システムおよびパッケージの開発や本番環境を構築する際には、関連ドキュメントを参照の上、インストールおよび構成を実施してください。また、本ガイドは単に情報として提供されるものであり、内容に誤りがないことの保障や弊社サポート部門へのお問い合わせはできませんのでご理解ください。サーバー仮想化ソリューションに対するサポートに関しては、以下のページからご確認いただけます。

<http://www.oracle.com/technetwork/jp/topics/ojkb120560-426058-ja.html> または

<http://www.oracle.com/technetwork/database/virtualizationmatrix-172995.html>

### 1.1 対象読者

本ガイドにおける対象読者には、主に以下の方を想定しています。

- RAC の基本的な知識を有する方
- 手持ちの環境で RAC のインストールや設定方法を確認されたい方
- 最新のバージョンにおける RAC のアーキテクチャ、あるいは機能や動作に興味のある方

## 1.2 関連文書

本ガイドでは、Oracle Database 12c Release 1 に関する記載について、以下のマニュアルを参考としています。機能および使用方法の詳細などについては、以下のマニュアルを参照してください。

- Oracle® Grid Infrastructure インストレーション・ガイド 12c リリース 1 (12.1) for Linux
- Oracle® Real Application Clusters インストレーション・ガイド 12c リリース 1 (12.1) for Linux and UNIX Systems
- Oracle® Clusterware 管理およびデプロイメント・ガイド 12c リリース 1 (12.1)
- Oracle® Real Application Clusters 管理およびデプロイメント・ガイド 12c リリース 1 (12.1)
- Oracle® Automatic Storage Management 管理者ガイド 12c リリース 1 (12.1)
- Oracle® Database プラットフォーム共通日本語 README 12c リリース 1 (12.1)
- Oracle® Database リリース・ノート 12c リリース 1 (12.1) for Linux

各マニュアルは、Oracle Technology Network の『Oracle Database オンライン・ドキュメント 12c リリース 1 (12.1) 』 (URL : <http://www.oracle.com/technetwork/jp/indexes/documentation/index.html>) より提供されます。

## 1.3 省略および表記規則

本ガイドでは、以下の省略表記および表記規則を用いています。

<省略表記>

名称	省略表記
Automatic Storage Management Configuration Assistant	ASMCA
Cluster Time Synchronization Service	CTSS
Database Configuration Assistant	DBCA
Operating System	OS
Oracle Automatic Storage Management	Oracle ASM または ASM
Oracle Database 12c Release 1	12c
Oracle Enterprise Manager Database Express	EM Express
Oracle Real Application Clusters	Oracle RAC または RAC
Oracle Universal Installer	OUI
Single Client Access Name	SCAN
Virtual IP	VIP
マルチテナント・コンテナ・データベース	CDB
プラグブル・データベース	PDB

## &lt;表記規則&gt;

規則	意味
太字	強調、あるいは操作に関連する GUI 要素を示す
イタリック体	ユーザーが特定の値を指定する変数を示す
網かけ	入力値、あるいは実行するコマンドを示す
# 記号	bash シェルの root ユーザーでの実行を示す
\$ 記号	bash シェルの Oracle Grid Infrastructure または Oracle Database インストール・ユーザーでの実行を示す

## 2. 概要

構築する環境と環境構築に使用するソフトウェアの概要について説明します。

### 2.1 Oracle Database 12c Release 1 の新機能

Oracle Database 12c Release 1 の新機能で、本ガイドの環境構築に関連する機能を説明します。

#### 1. Oracle Flex Clusters

Oracle Flex Clusters は、大規模かつ動的なクラスタ機能を提供します。Oracle Flex Clusters を用いたクラスタ構成では、非常に多くのノードで構成される Oracle RAC データベースや高可用性が求められるアプリケーションのプラットフォームとして利用することが可能です。

Oracle Flex Clusters 構成では、次の 2 タイプのノードによりハブ・アンド・スポークの構成を取ります。

- Hub ノード (ハブ・ノード)
  - 共有ストレージへの直接アクセスを行い、データベース・インスタンスといった集中的な I/O が発生するワークロードを実行するためのノードです。クラスタ内のハブ・ノードは、ネットワークおよびストレージを介して連携しています。
- Leaf ノード (リーフ・ノード)
  - 共有ストレージへの直接アクセスすることなく、プロセッシング処理を中心としたアプリケーションなどのワークロードを実行するためのノードです。

Oracle Flex Clusters を構成するすべてのハブ・ノードとリーフ・ノードに Oracle Grid Infrastructure をインストールし、1 つのクラスタとして構成する必要があります。Oracle Grid Infrastructure のインストール時に Oracle Flex Clusters 構成、あるいは 11g と同様のクラスタ構成を選択が可能で、インストール後も構成を切り替えることができます。

#### 2. Oracle Flex ASM

ストレージを管理するための Oracle Automatic Storage Management (ASM) の機能を強化し、Oracle Flex ASM を提供します。Oracle Flex ASM を用いた構成では、データベース・インスタンスの ASM インスタンスに対する依存性が取り除かれます。これにより、データベース・インスタンスとは異なるノードに ASM インスタンスを配置することも可能です。

11g までと同様の ASM 構成を用いることも可能ですが Oracle Flex ASM を用いた場合、次のようなメリットがあります。例えば、大規模なクラスタ環境を構築する場合にはすべてのノード上で ASM インスタンスを起動させる必要がなくなったため、クラスタ全体で使用する ASM のリソース消費を低く抑えることができます。また、ASM インスタンスに障害が発生した場合でもデータベース・インスタンスに影響を与えることがなくなるため、可用性を高めることができます。

#### 3. マルチテナント・アーキテクチャ

データベース統合やクラウド環境の実現といった要望に応えるために、データベースのレイヤーでマルチ

テナントを実現するためのアーキテクチャを提供します。

データベース統合やクラウド環境の構築には、これまでもデータベースに限らず様々なレイヤーにおいて実現のための手法が提供されてきました。例えば、仮想化ソフトウェアを用いたサーバー統合やスキーマを用いたインスタンス統合といった手法があります。マルチテナント・アーキテクチャは、これらの手法で課題とされていた仮想化によるオーバーヘッド、スケーラビリティに関する制限、統合にかかるコストといった面を改善する新しいアーキテクチャです。

12cでは、マルチテナント・アーキテクチャに対応したデータベースとして、マルチテナント・コンテナ・データベース (CDB) を提供します。さらに、CDB 内には 1 つ以上のプラガブル・データベース (PDB) を作成することができます。PDB はスキーマや表領域が含まれる論理的なセットであり、基本的にユーザーやアプリケーションからは通常のデータベースと同様に扱うことができます。CDB および PDB を使用できるマルチテナント・アーキテクチャを用いて、既存のスキーマやアプリケーションを変更することなく複数のデータベース環境を統合することを可能にします。従来アーキテクチャに対応したデータベースも non-CDB として提供されており、引き続き利用することができます。

CDB は、コンテナとも呼ばれる次の 3 つの要素により構成されます。

- ルート (CDB\$ROOT)
- シード (PDB\$SEED)
- プラガブル・データベース (PDB)

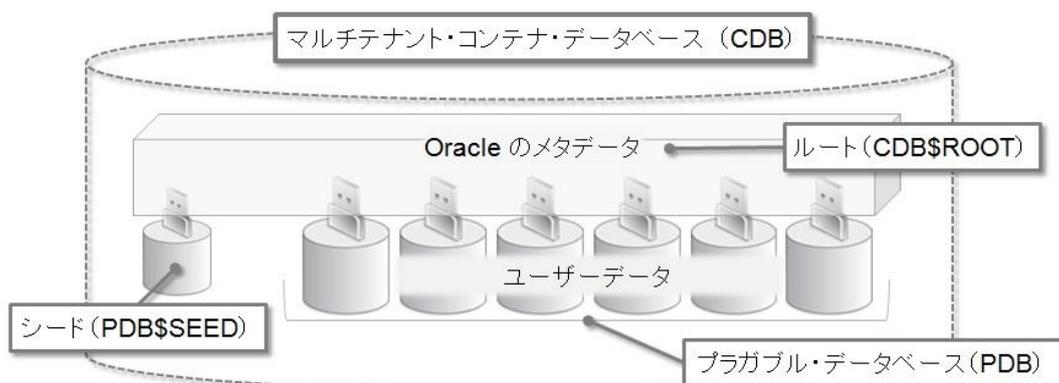


図 2: マルチテナント・アーキテクチャ概要図

#### 4. Oracle Enterprise Manager Database Express

Web ブラウザを使用してデータベースの監視や管理を実施する機能として、Oracle Enterprise Manager Database Express (EM Express) を提供します。EM Express は単一のデータベース・システムの管理を行います。複数のデータベース・システム、または複数のデータベース以外のシステムを対象に統合管理を行う場合には、Oracle Enterprise Manager Cloud Control (EMCC) を使用します。

EM Express は Oracle Database Configuration Assistant (DBCA) によるデータベースを作成時に「Enterprise Manager (EM) Database Express の構成」チェック・ボックスを選択すると自動的に構成されます。また、データベース作成後に手動で構成することも可能です。

EM Express はデータベース上に構成され、非常に軽量なため、データベースサーバーへの負荷は小さくなります。

## 2.2 ハードウェア

本ガイドの環境は、x86-64アーキテクチャの物理マシンを1台使用して構築するものとします。1台の物理マシン上に、RACの構築に必要なサーバーだけではなくSingle Client Access Name (SCAN) 用のIPアドレスについて名前解決をするためのDNSサーバーも構成します。別途利用可能なDNSサーバーがある場合には、環境構築に使用することも可能です。

参考として環境構築に使用した物理マシンのスペックを記載します。

- CPU : Intel (R) Core (TM) i5-2520M CPU @ 2.50GHz 2.50 GHz
- メモリ : 8GB
- ディスク : 280GB (最低要件としては 50GB 程度の空き容量が必要)
- OS : Windows 7 Professional Service Pack 1 (64 bit)

※ クラスタを構成するサーバーについてメモリの最小要件は、1台あたり4GBです。今回に限り、最低限の動作確認を行う目的で環境を構築するため1台の仮想マシンあたり2.5GBとしています。可能であれば最低要件である4GBを確保するようにしてください。

## 2.3 ソフトウェア

本ガイドにおいて、環境構築に使用したソフトウェアは以下です。

- Oracle VM VirtualBox 4.2.18 for Windows hosts
- Oracle VM VirtualBox 4.2.18 Oracle VM VirtualBox Extension Pack
- Oracle Linux 6 Update 4 x86-64
- Oracle Database 12c Release 1 (12.1.0.1) Grid Infrastructure
- Oracle Database 12c Release 1 (12.1.0.1)

## 2.4 ネットワーク

仮想マシンに対して複数の仮想NICを割り当てます。仮想マシンに対する仮想NICの割り当ては物理マシンのNIC搭載数には依存しません。物理マシンに搭載されているNICが1つだとしても、仮想マシンには複数の仮想NICを割り当てることができます。

本ガイドでは、クラスタを構成する各仮想マシンには次の3つの仮想NICを割り当てて使用します。

- eth0 : パブリック・ネットワークとして使用
- eth1 : プライベート・ネットワークとして使用 (インターコネクトの冗長化機能を使用)
- eth2 : プライベート・ネットワークとして使用 (インターコネクトの冗長化機能を使用)

※ インターコネクトを冗長化しない場合には eth2 は不要です。

DNSサーバーとする仮想マシンには、次の1つの仮想NICを割り当てます。

- eth0 : パブリック・ネットワークとして使用

これまでのハードウェア、ソフトウェア、ネットワークに関する説明を踏まえ、本ガイドにおける構成概要図を以下に記載します。

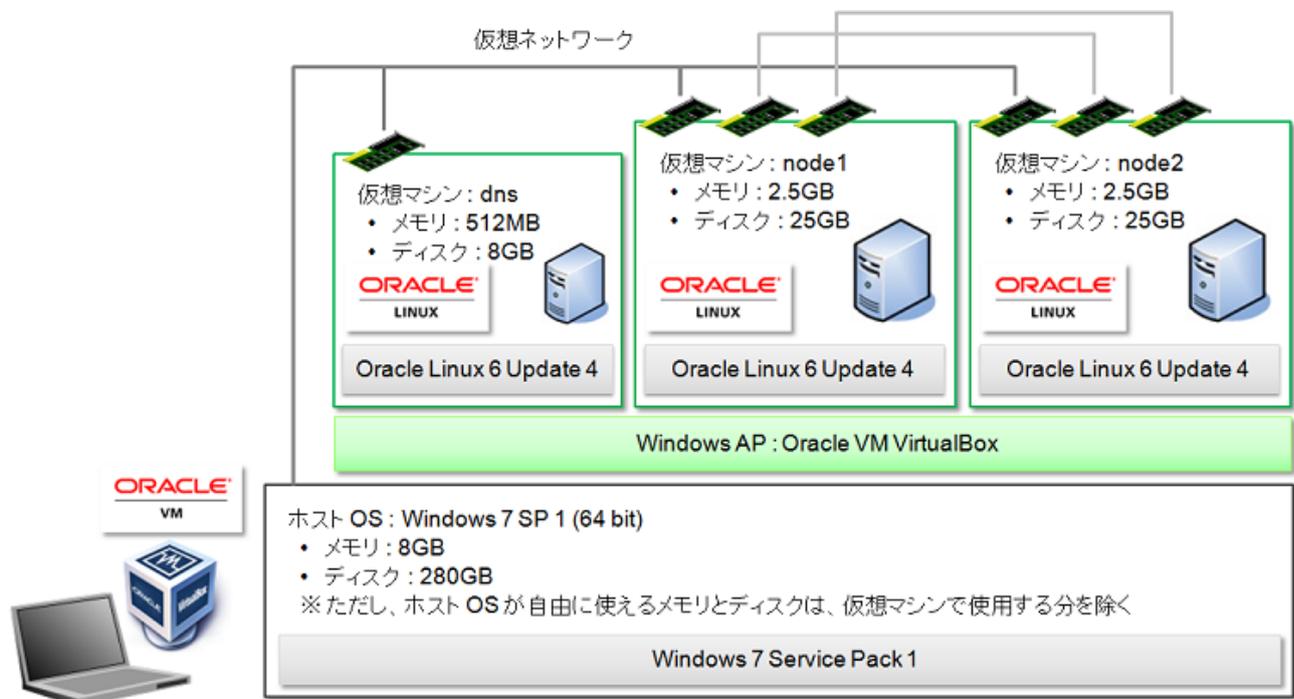


図 3 : 本ガイドにおける構成

#### <IP アドレス一覧>

ホスト名	IP アドレス	用途
node1.oracle12c.jp	192.168.56.101	node1 の eth0 (パブリック・ネットワーク)
	192.168.100.101	node1 の eth1 (プライベート・ネットワーク)
	192.168.200.101	node1 の eth2 (プライベート・ネットワーク)
node2.oracle12c.jp	192.168.56.102	node2 の eth0 (パブリック・ネットワーク)
	192.168.100.102	node2 の eth1 (プライベート・ネットワーク)
	192.168.200.102	node2 の eth2 (プライベート・ネットワーク)
scan.oracle12c.jp	192.168.56.201	SCAN 用仮想 IP (VIP)
	192.168.56.202	SCAN 用 VIP
	192.168.56.203	SCAN 用 VIP
dns.oracle12c.jp	192.168.56.254	DNS サーバーの eth0 (パブリック・ネットワーク) ※

※ DNS サーバーを node1 および node2 と同様に仮想マシンで用意する場合に必要です。既存の DNS サーバーを使用する場合には必要ありません。

### 3. Oracle VM VirtualBox のインストールと設定

ここでは、Oracle VM VirtualBox のインストールと、インストール後に実施しておく Oracle VM VirtualBox の設定について以下の順に説明します。

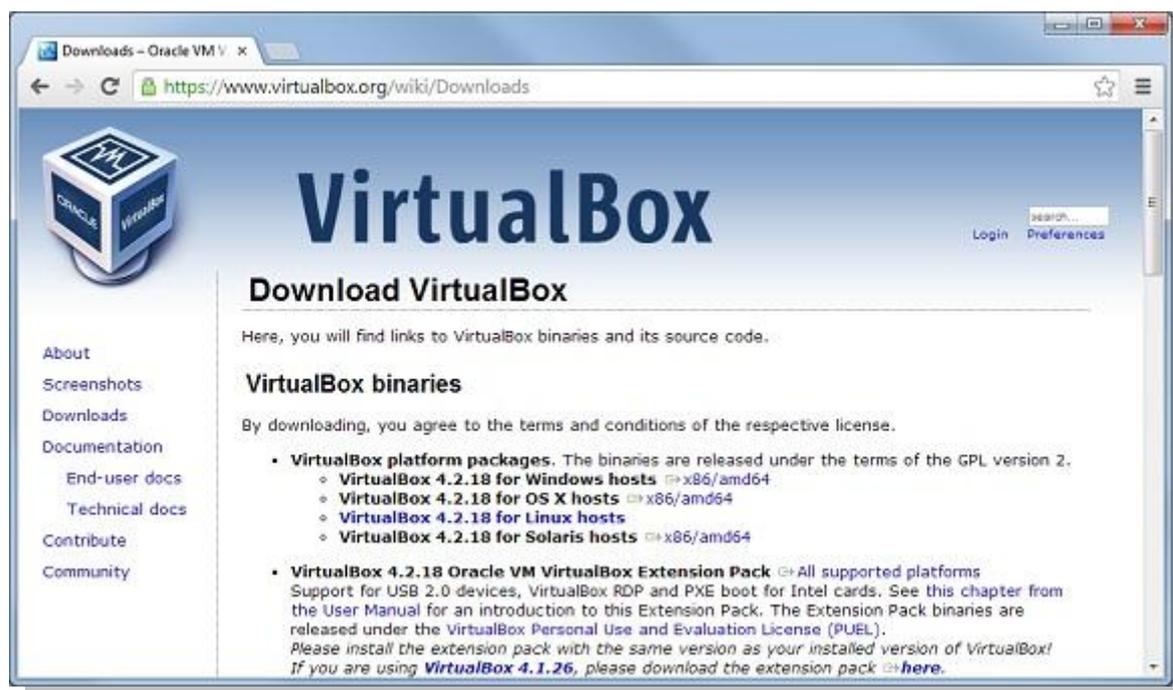
- 3.1 Oracle VM VirtualBox のインストール
- 3.2 機能拡張パッケージの追加インストール
- 3.3 インストール後の設定
- 3.4 仮想マシンの作成

#### 3.1 Oracle VM VirtualBox のインストール

##### 1. ソフトウェアのダウンロード

Oracle VM VirtualBox のダウンロード・ページより、(URL : <http://www.virtualbox.org/wiki/Downloads> ) 必要なソフトウェアをダウンロードします。ここでは以下 2 つのソフトウェアをダウンロードするものとします。

- VirtualBox 4.2.18 for Windows hosts
- VirtualBox 4.2.18 Oracle VM VirtualBox Extension Pack



ソフトウェアは Oracle Technology Network から入手が可能です。(URL : <http://www.oracle.com/technetwork/jp/server-storage/virtualbox/downloads/index.html> )

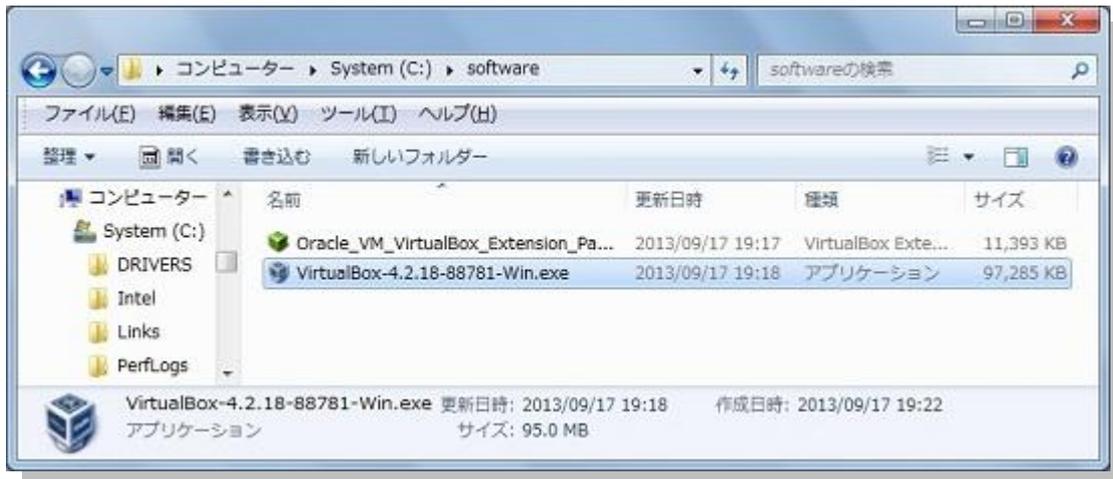
Oracle VM VirtualBox Extension Pack は、USB 2.0 のサポートやホスト OS とゲスト OS 間におけるデスクトップ上の操作をシームレスに行う機能などを提供するプラグイン(機能拡張パッケージ)です。機能拡張パッケージのインストールは任意ですが、ここでは管理者権限を持つユーザー・アカウントを使用して機能拡張パッケージのインストールを行うものとします。

ここでは VirtualBox 4.2.18 を使用した手順を紹介しますが、基本的に他の上位バージョンでも同様の手順で環境を構成することができます。

## 2. Setup Wizard の起動

ダウンロードした exe ファイルを実行して、Setup Wizard を起動します。ここでは、ダウンロードした exe ファイルを以下の場所に配置して使用します。

C:\software\VirtualBox-4.2.18-88781-Win.exe



exe ファイルの実行時に、セキュリティの警告画面が表示された場合は、確認の上、「実行」をクリックして作業を続けます。

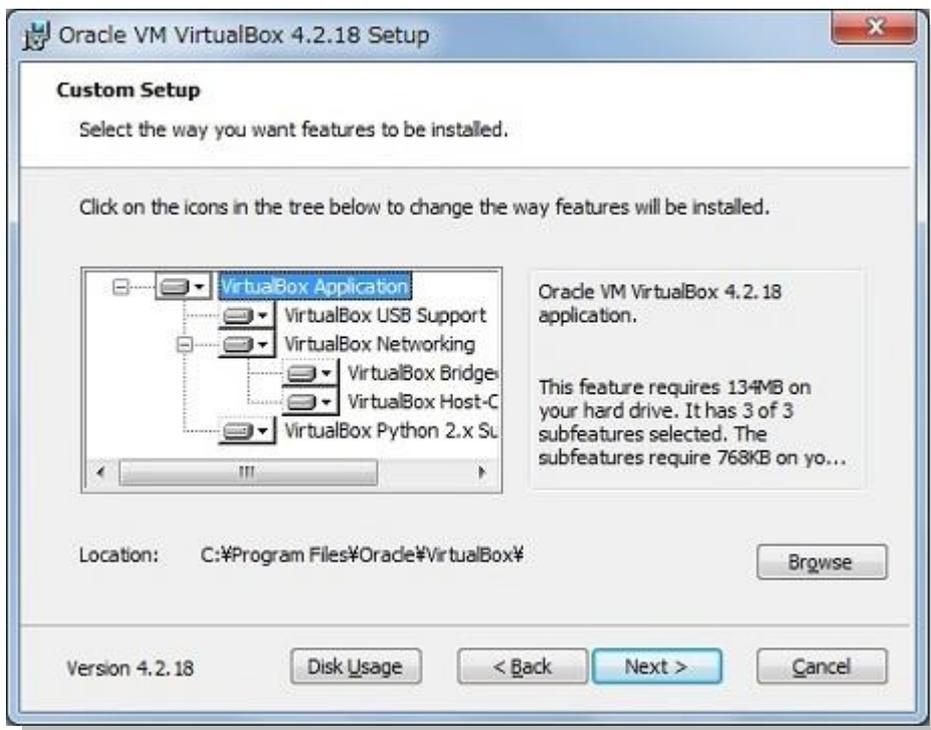


Setup Wizard の起動後は、「Next」をクリックしてインストールを開始します。

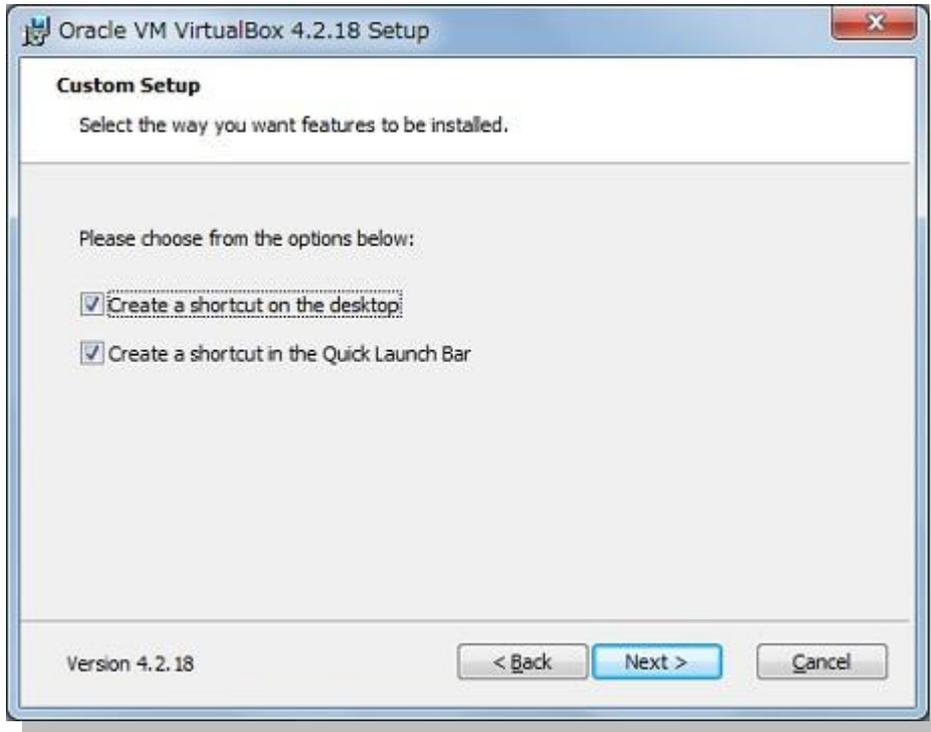


### 3. Custom Setup の設定

続いて、インストールする機能を選択します。ここでは、デフォルトの設定でインストールを行うものとするので「Next」をクリックします。



続いてショートカットの作成に関するオプションを選択します。ここでも、デフォルトの設定でインストールを継続するものとしますので「Next」をクリックします。



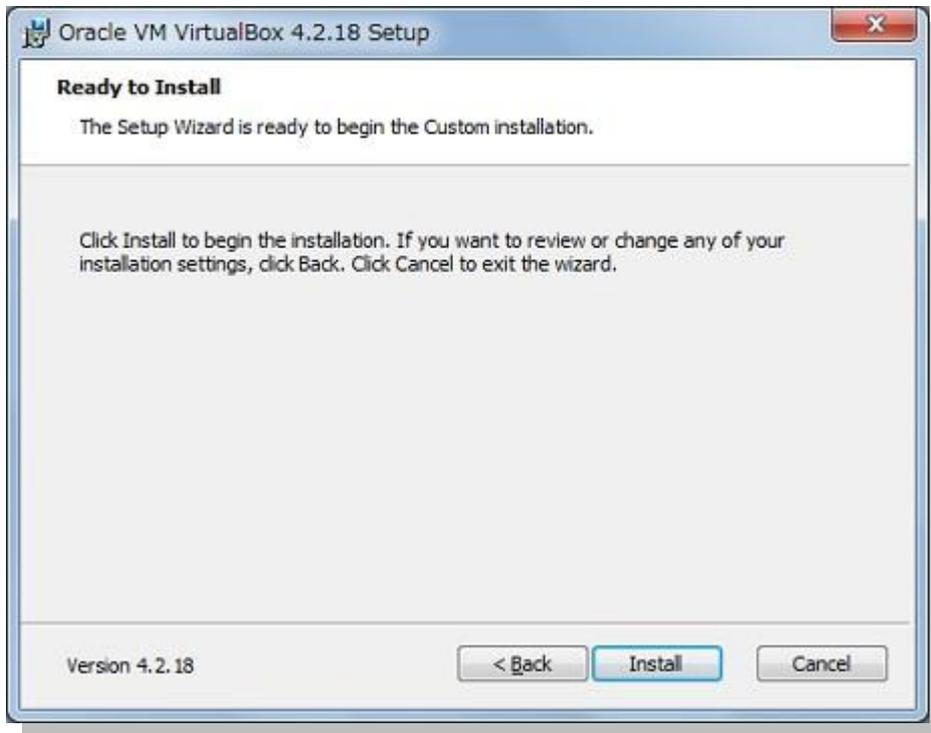
#### 4. Network Interface 警告の確認

ネットワークに関する機能のインストールに関して、一時的にネットワークが中断される旨の警告メッセージが表示されます。警告メッセージを確認の上、「Yes」をクリックしてインストールを継続します。



## 5. インストールの開始

ここまでで、インストールの準備は完了です。「Install」をクリックして、インストールを開始します。



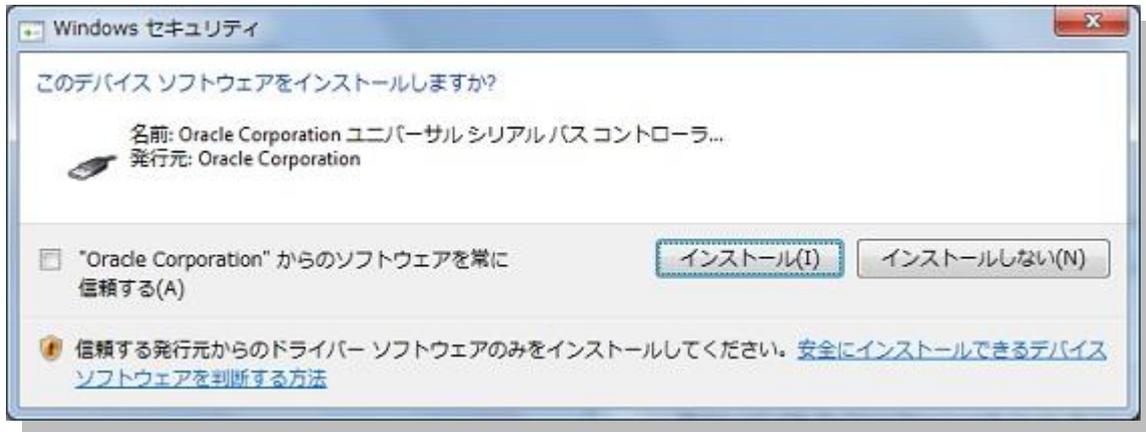
## 6. インストール中の確認

インストール中にアカウント制御により許可を求められた場合には、適宜確認の上、「はい」をクリックして、インストールを継続してください。



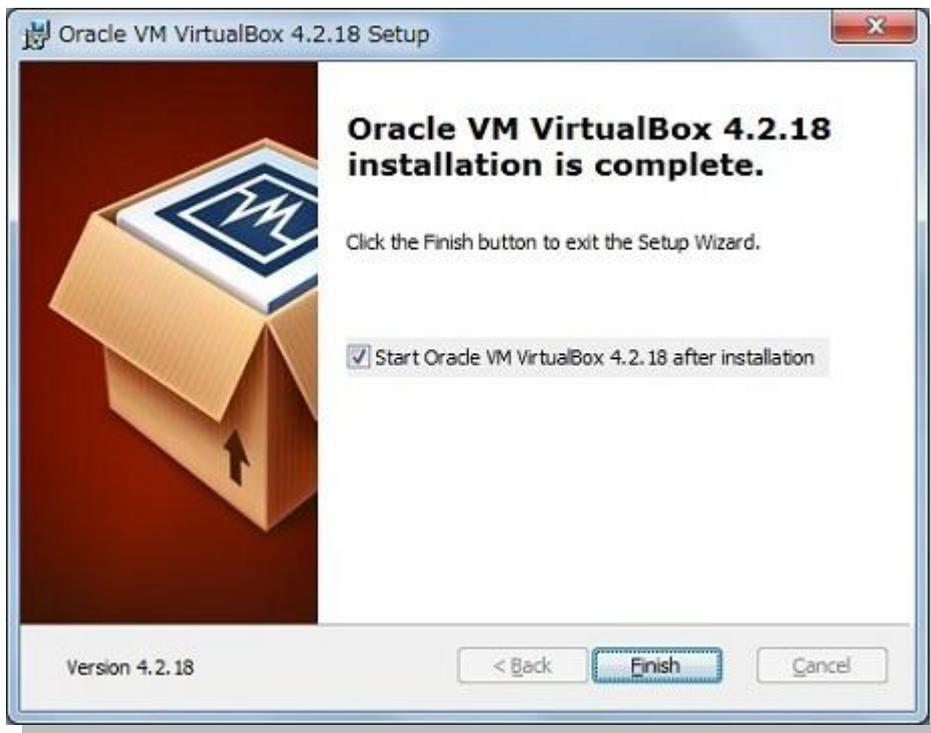
また、インストール中に以下のソフトウェアに関して、インストール可否の確認を求められた場合には、すべてのソフトウェアについて、「インストール」をクリックして、インストールを行うものとします。

- Oracle Corporation ユニバーサルシリアルバスコントローラー
- Oracle Corporation Network Service
- Oracle Corporation ネットワークアダプター など

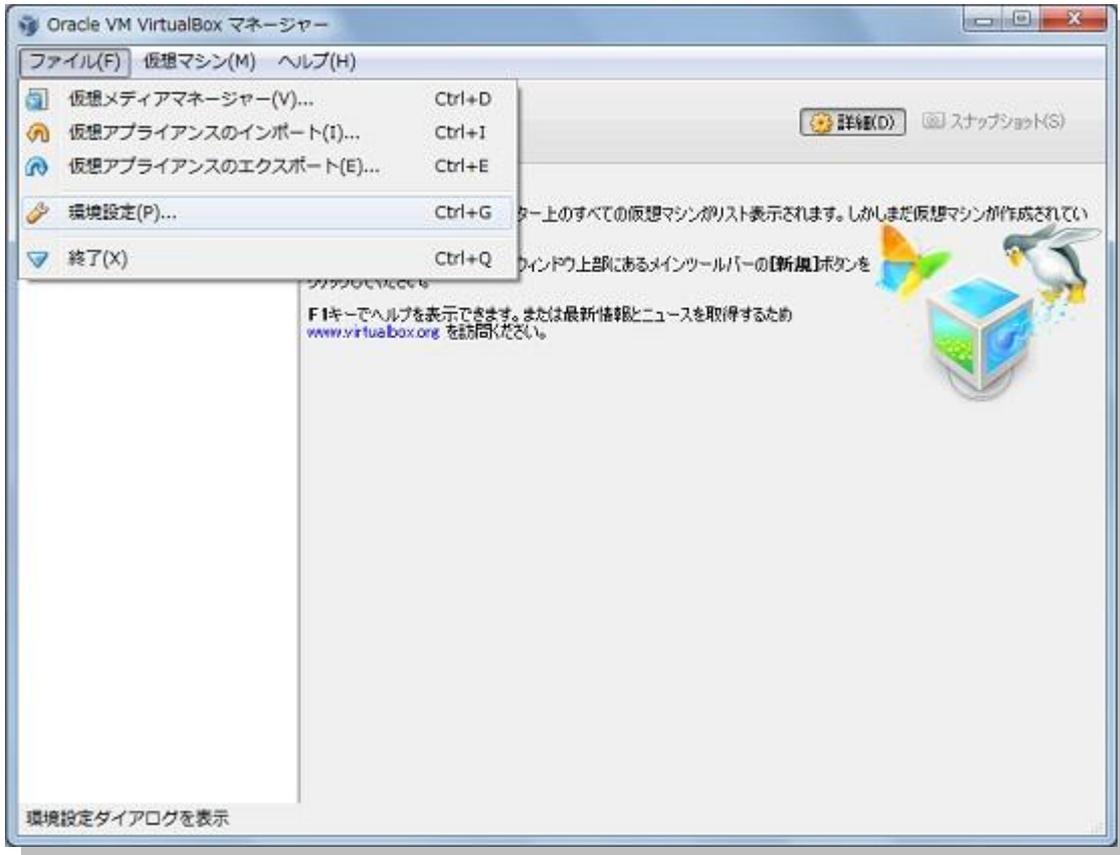


## 7. インストールの完了

インストールが完了すると、以下の画面が表示されます。「**Start Oracle VM VirtualBox 4.2.18 after installation**」にチェック(☑)をすると、Setup Wizard 終了後に Oracle VM VirtualBox マネージャーが起動されます。ここでは、デフォルトの(チェックをつけている)状態で「**Finish**」をクリックして Setup Wizard を終了します。



インストール作業は以上です。ここでは Oracle VM VirtualBox マネージャーが起動されたことを確認して、画面右上の「×」をクリックして画面を閉じます

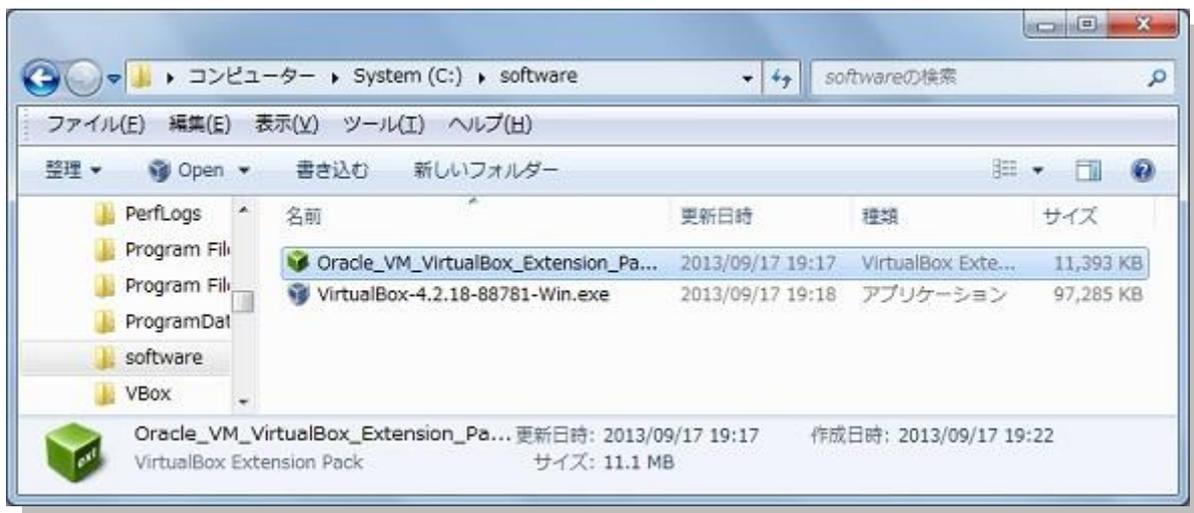


## 3.2 機能拡張パッケージの追加インストール

### 1. 機能拡張パッケージのインストールの実行

ダウンロードした機能拡張パッケージを追加インストールします。ここでは、以下に配置したダウンロード済みの Oracle VM VirtualBox Extension Pack のファイルを実行して追加インストールを開始します。

C:\software\Oracle\_VM\_VirtualBox\_Extension\_Pack-4.2.18-88780.vbox-extpack

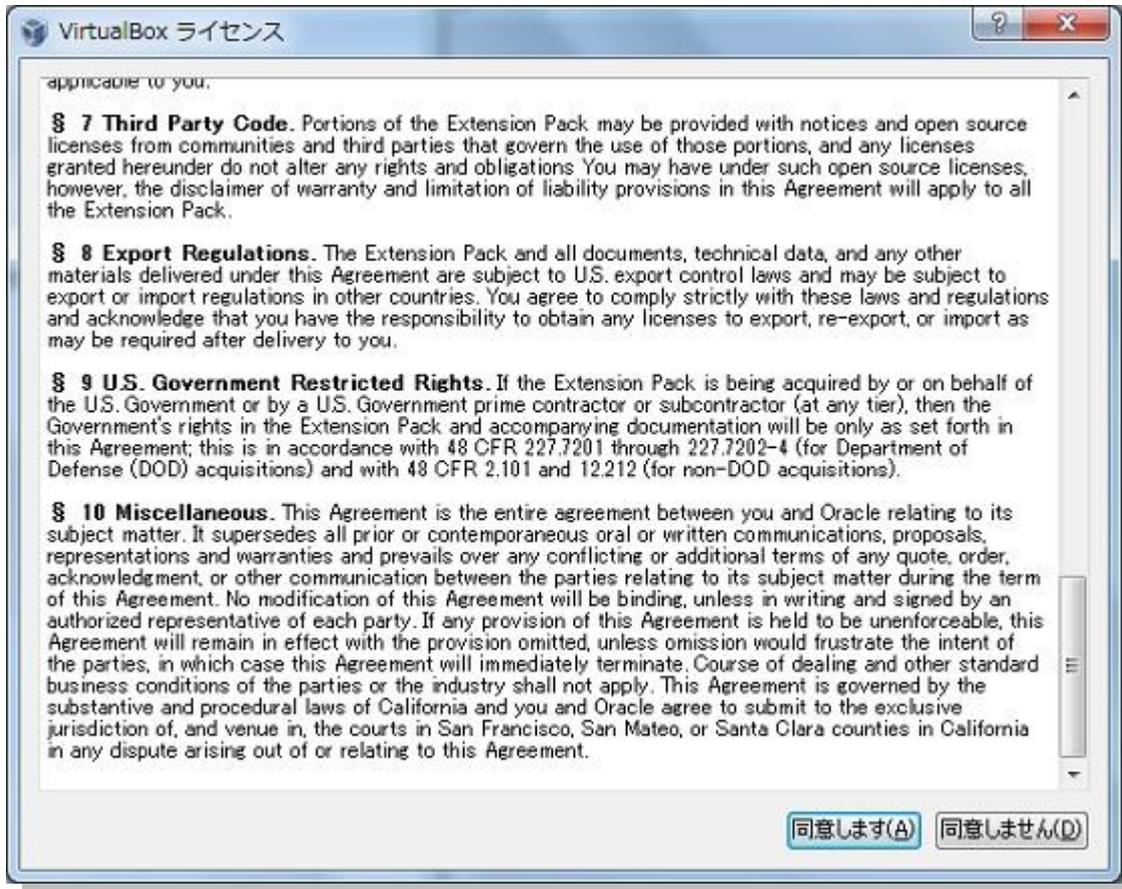


ファイルを実行すると、Oracle VM VirtualBox マネージャーの画面が表示されます。続いて、処理の実行について確認画面が表示されたら、「インストール」をクリックして続けます。



## 2. ライセンスとアカウント制御による確認

ライセンスに関する情報が表示されますので、確認の上、「同意します」をクリックして続きます。(記述を最後までスクロールするとボタンのクリックが可能になります。)



ユーザーのアカウント制御により確認画面が表示された場合は、「はい」をクリックして続きます。



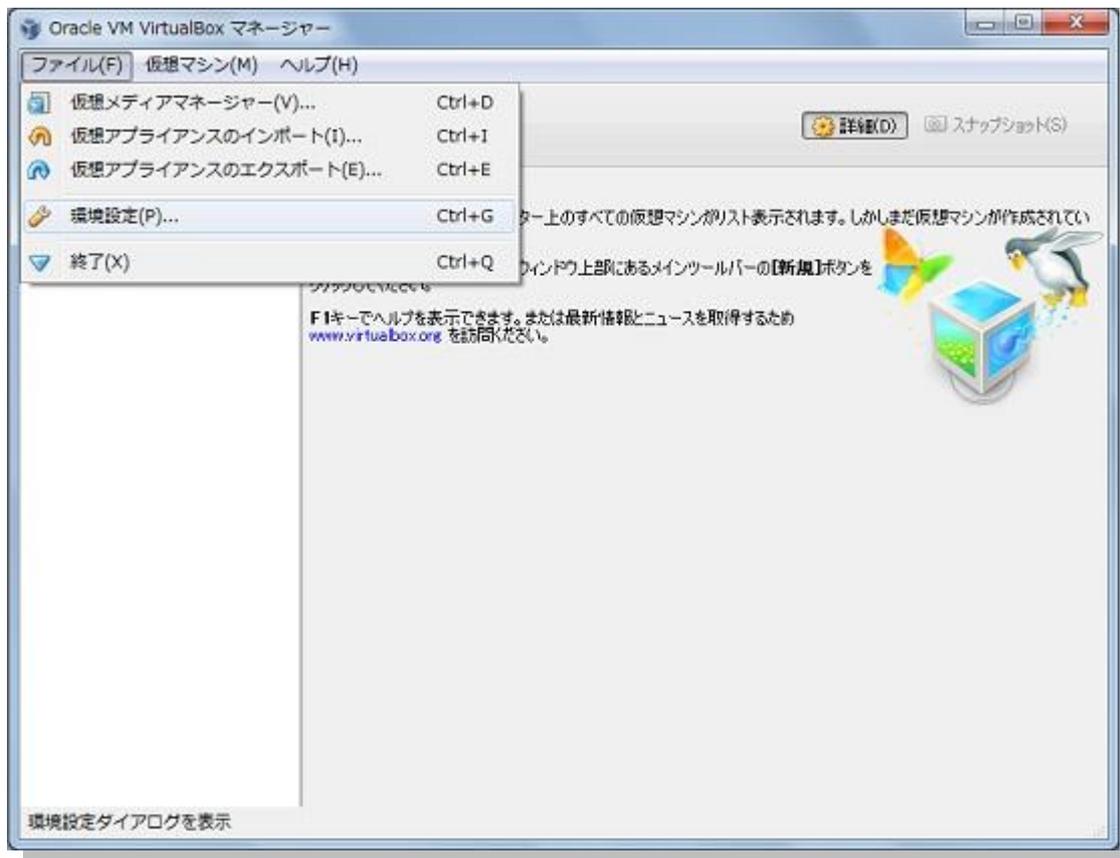
### 3. 機能拡張パッケージのインストールの完了

機能拡張パッケージのインストール完了後に表示されるメッセージを確認して、作業は完了です。「OK」をクリックしてください。

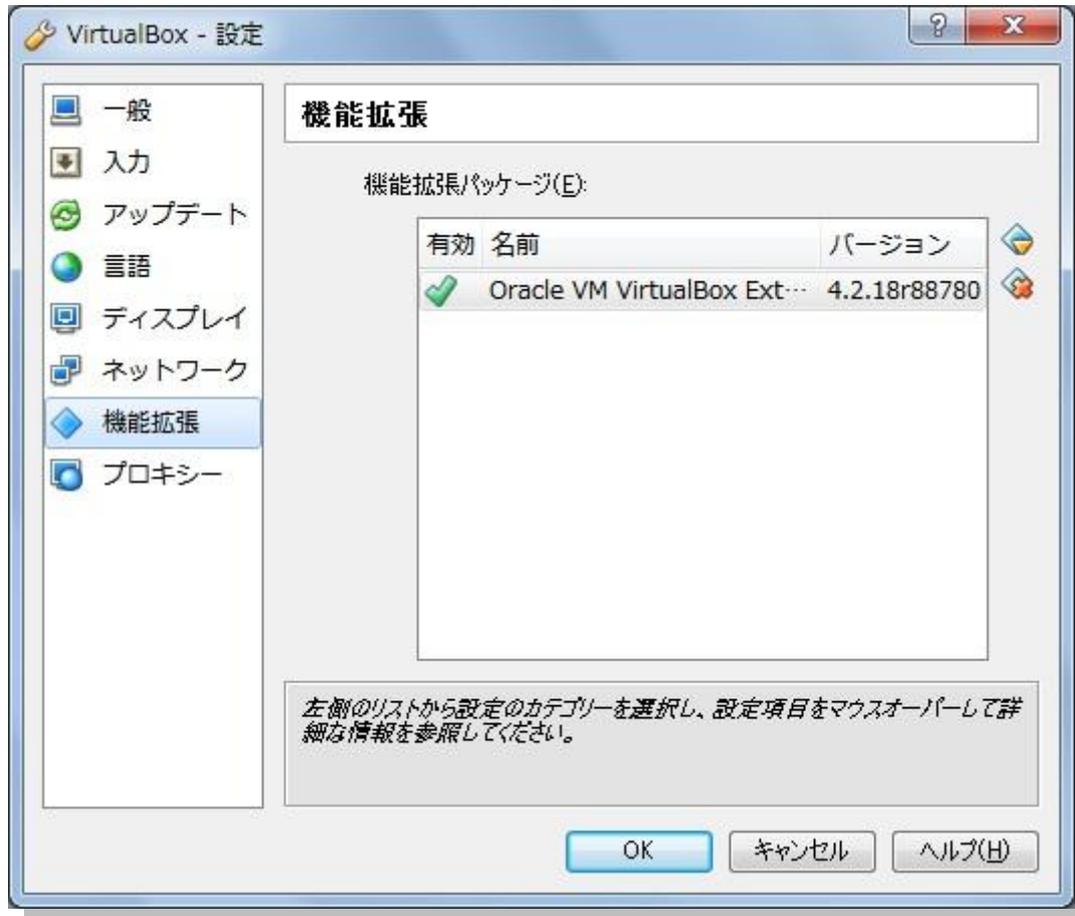


### 4. インストール後の確認

機能拡張パッケージのインストール完了後は、Oracle VM VirtualBox マネージャーから確認が可能です。「ファイル」タブの「環境設定」をクリックして設定画面を表示します。



画面左側の「機能拡張」を選択します。画面右側に表示された機能拡張パッケージのバージョンを確認して「OK」をクリックします。



### 3.3 インストール後の設定

#### 1. Oracle VM VirtualBox の設定

前述の設定画面より、引き続き Oracle VM VirtualBox で使用するフォルダーの設定を実施します。

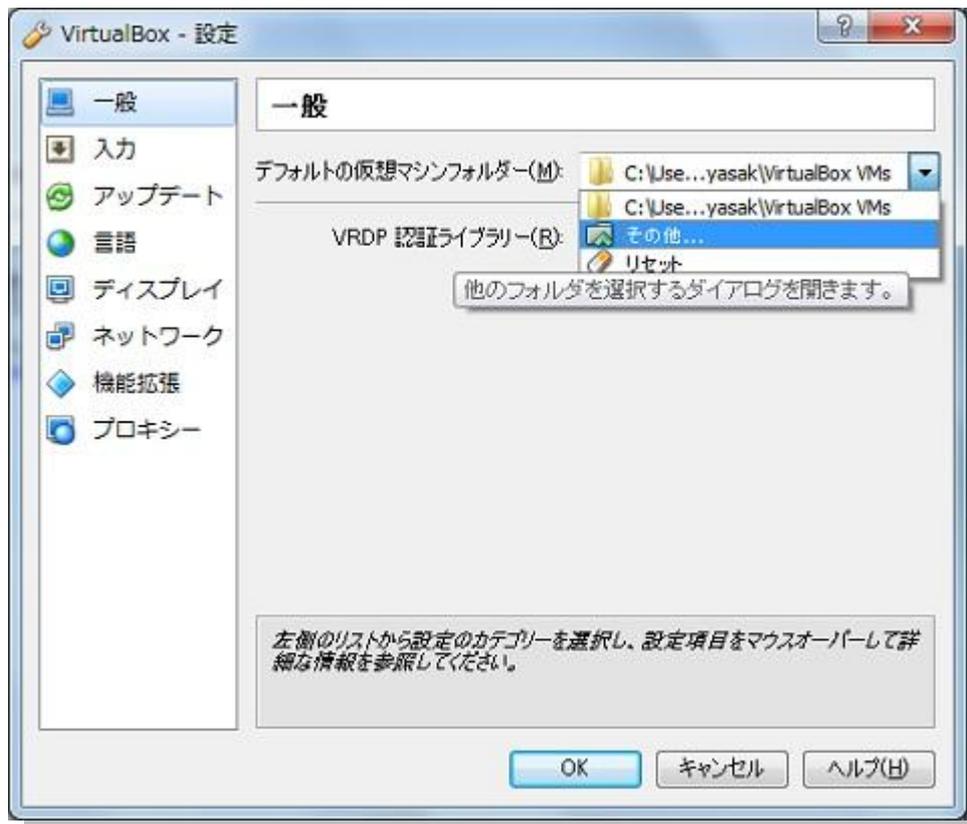
画面左側の「一般」を選択して、デフォルト仮想マシンフォルダーおよび VRDP 認証ライブラリに任意の場所を設定します。

デフォルト仮想マシンフォルダーには、ゲスト OS の情報が記載された xml ファイルや、仮想ディスクが配置されます。また、VRDP 認証ライブラリには、リモートディスプレイの認証ライブラリとして使用されます。

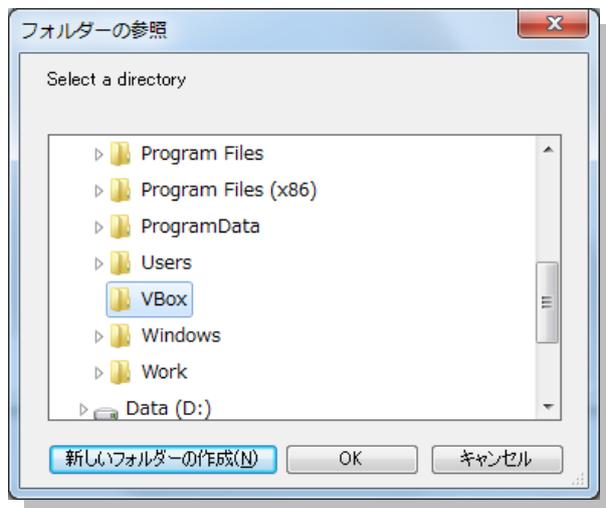
ここでは以下のように設定するものとして、引き続き手順を説明します。

デフォルト仮想マシンフォルダー	C:\¥VBox
VRDP 認証ライブラリ	VBoxAuth (今回はデフォルトのまま変更なし)

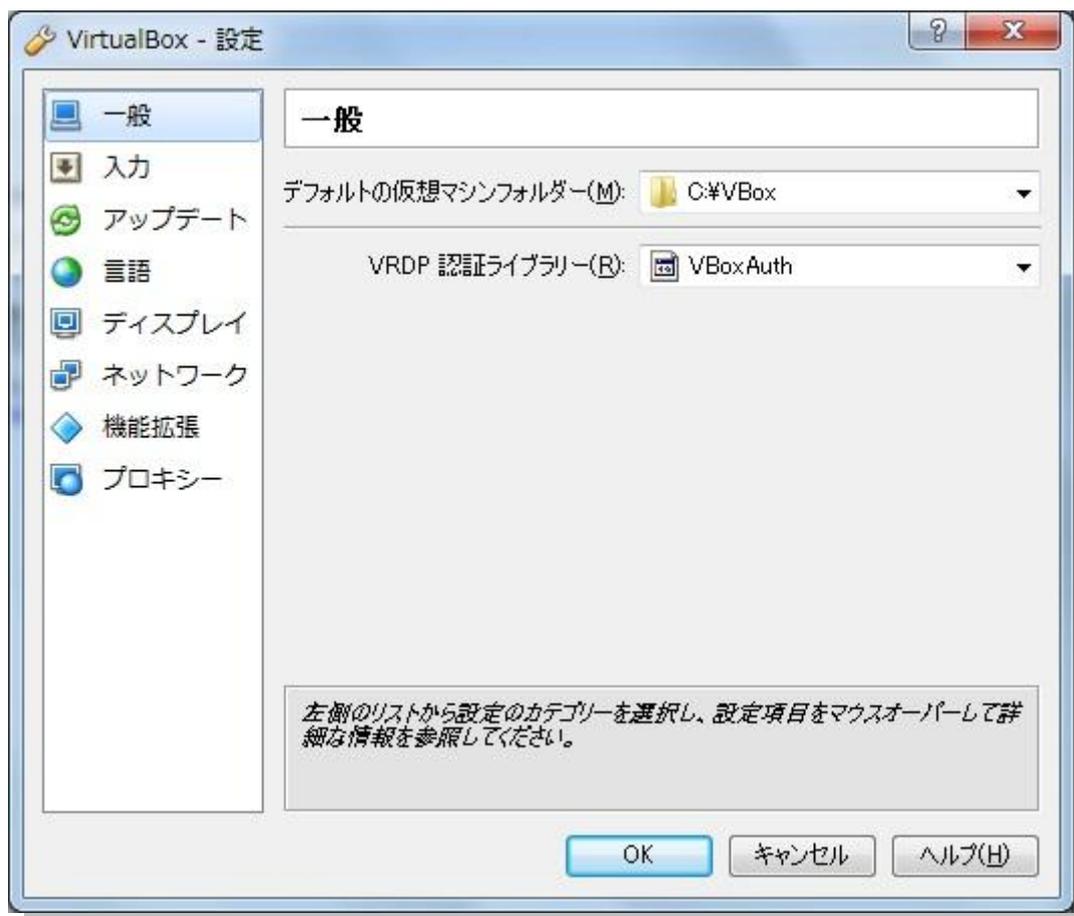
デフォルト仮想マシンフォルダーのプルダウンより「その他」を選択します。



表示された画面で「**System (C:)**」を選択して、「**新しいフォルダーの作成**」をクリックします。クリック後、今回は「**VBox**」を作成して選択し「**OK**」をクリックしてください。



設定画面に表示された、デフォルト仮想マシンフォルダーおよび VRDP 認証ライブラリを確認して、「**OK**」をクリックします。



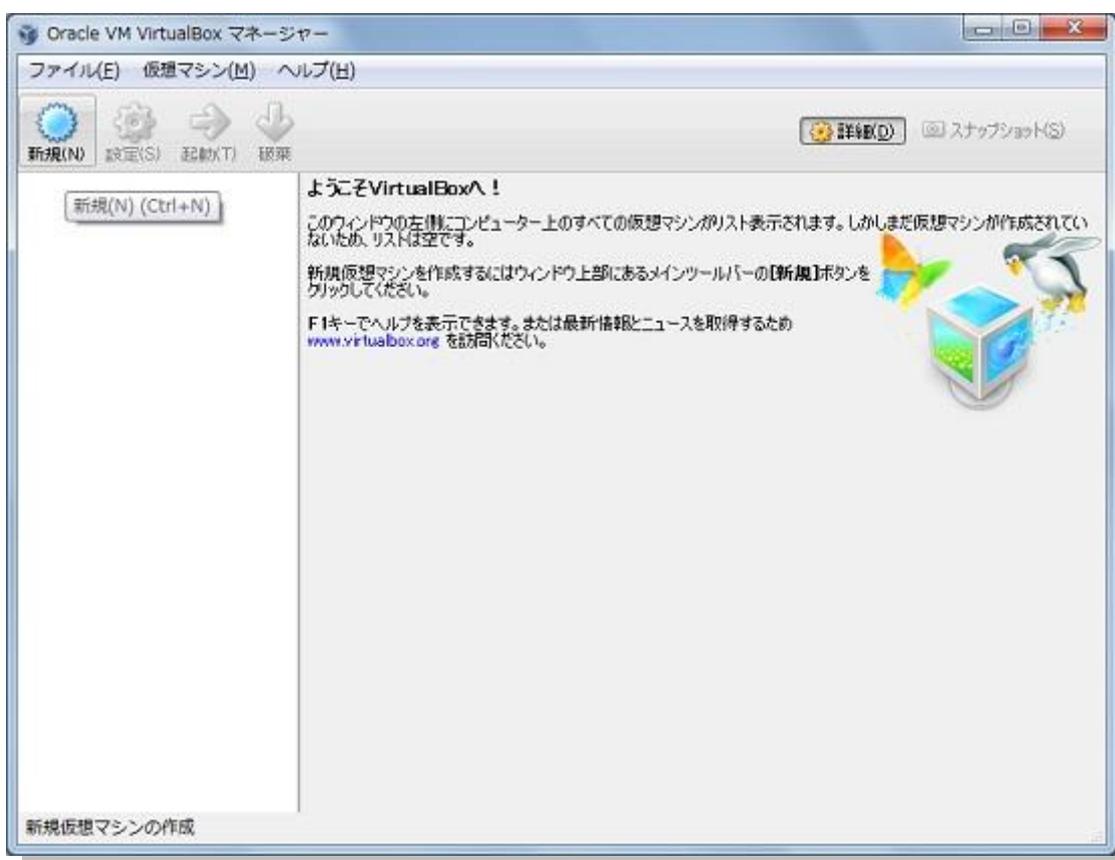
### 3.4 仮想マシンの作成

続いて、ゲスト OS として仮想マシンの作成を実施します。ここでは Oracle VM VirtualBox マネージャーを使用して仮想マシンを新規に作成していきます。

ここではまず、node1 の作成のみ実施します。node1 を作成後、OS インストールや設定を行った後に node1 をクローンして node2 や DNS サーバー用の仮想マシンを作成するものとします。node1 をクローンせずに、同様の手順で node2 や DNS サーバー用の仮想マシンを作成しても構いません。

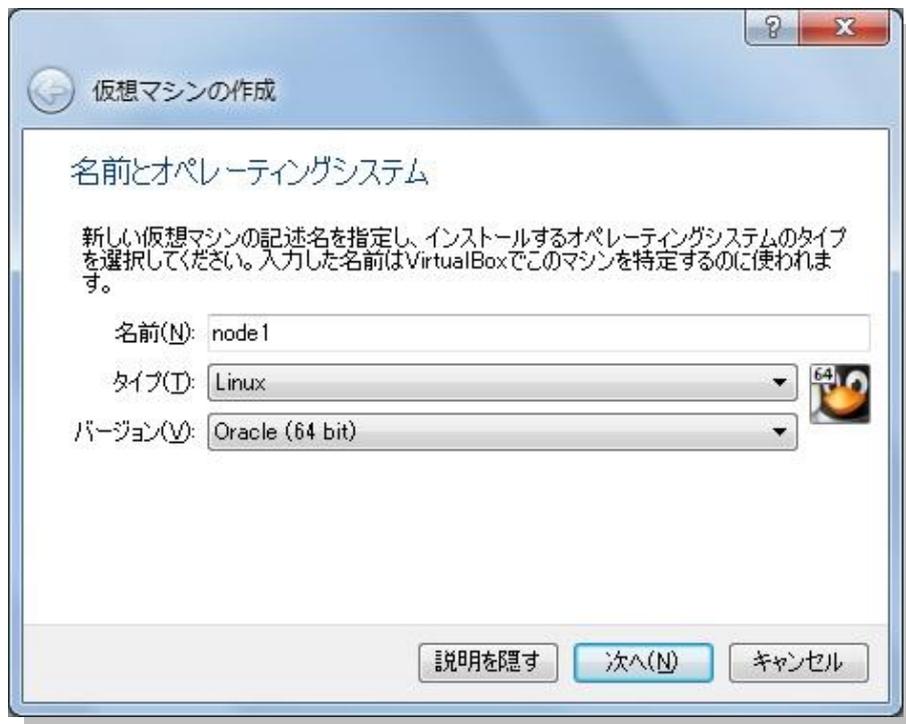
#### 1. 仮想マシンの新規作成

はじめに、Oracle VM VirtualBox マネージャーから、「**新規(N)**」をクリックします。または、「**仮想マシン**」タブより「**新規(N)**」をクリックします。



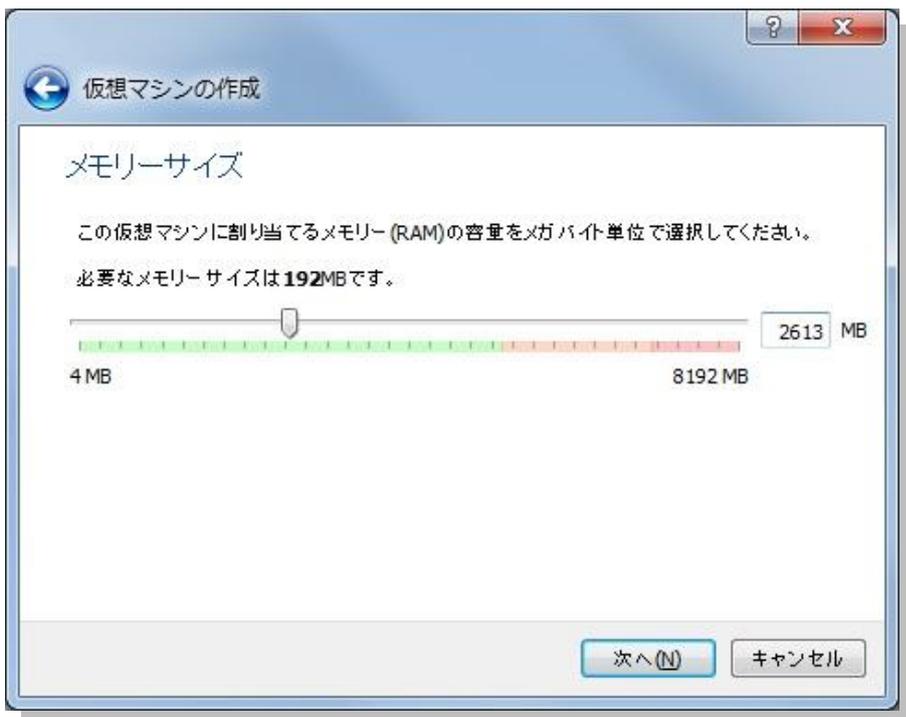
## 2. 仮想マシン名と OS タイプの入力

仮想マシンの名前として「**node1**」を入力します。また OS のタイプに「**Linux**」を、バージョンに「**Oracle (64bit)**」を選択します。入力後、「**次へ**」をクリックします。



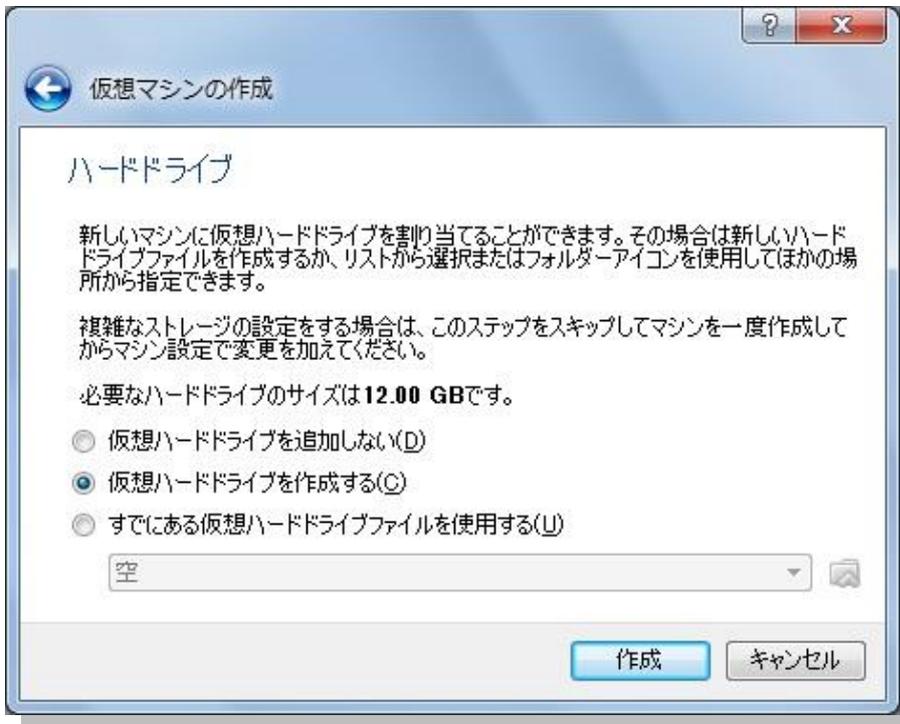
## 3. メモリの設定

仮想マシンに割り当てるメモリを設定します。ここでは「**2613**」MB (2.5GB)を設定するものとします。入力後は「**次へ**」をクリックします。(推奨としては 4GB ですが、今回は 2.5GB で作成します。)



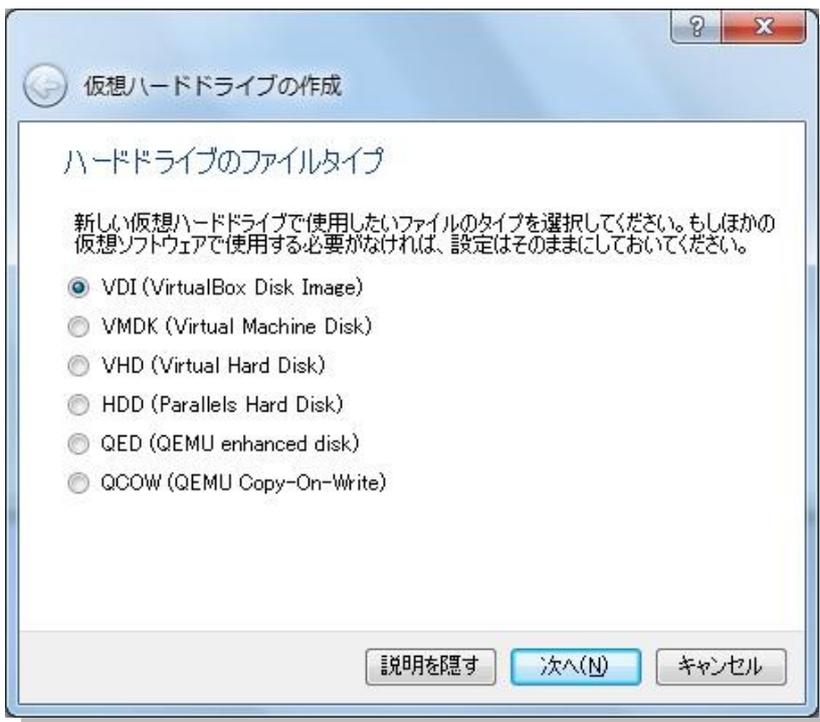
#### 4. 仮想マシンの作成

仮想マシンで使用する仮想ハードドライブを設定します。ここではまず、仮想ハードディスクを新規作成しますので「**仮想ハードドライブを作成する**」を選択して、「**作成**」をクリックします。

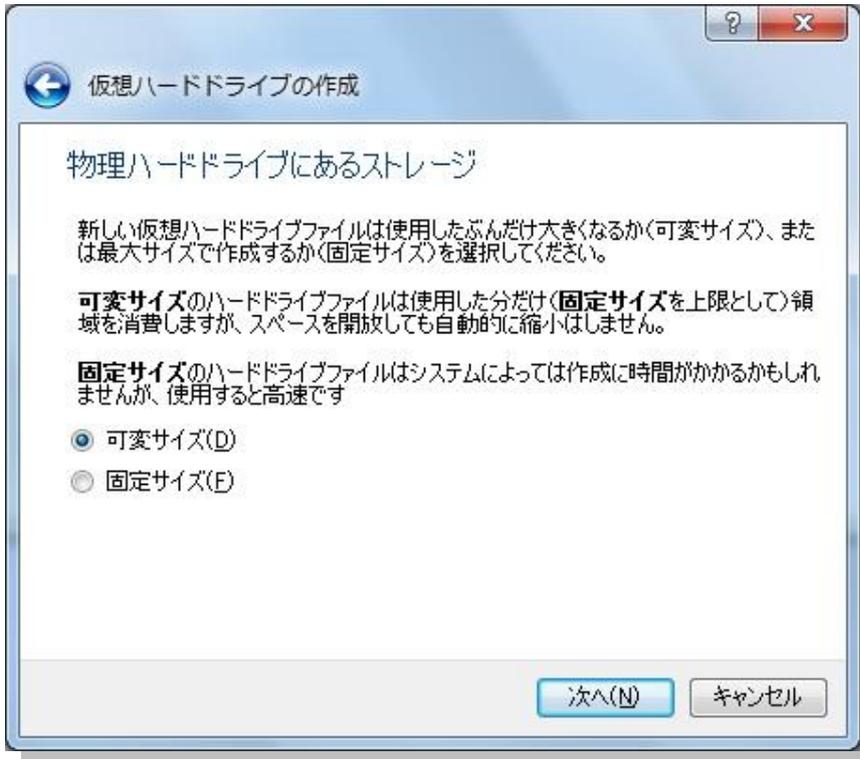


#### 5. 仮想ハードドライブの作成

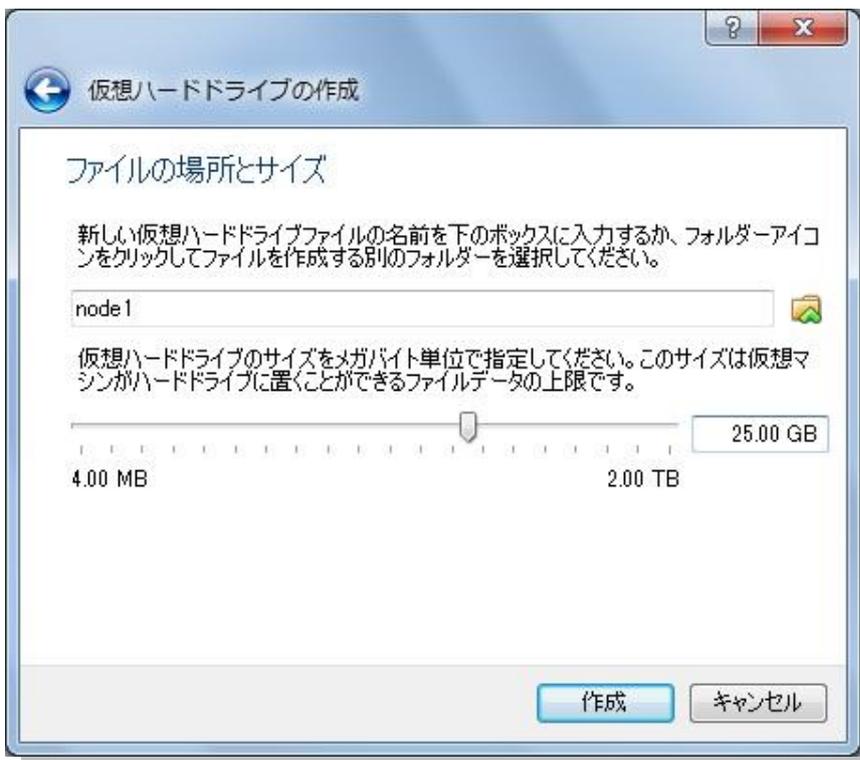
仮想ハードドライブのファイルタイプに「**VDI (VirtualBox Disk Image)**」を選択して、「**次へ**」をクリックします。



続いて領域の割り当て方法を選択します。今回は、動的に割り当てを行う「**可変サイズ**」を選択して、「**次へ**」をクリックします。

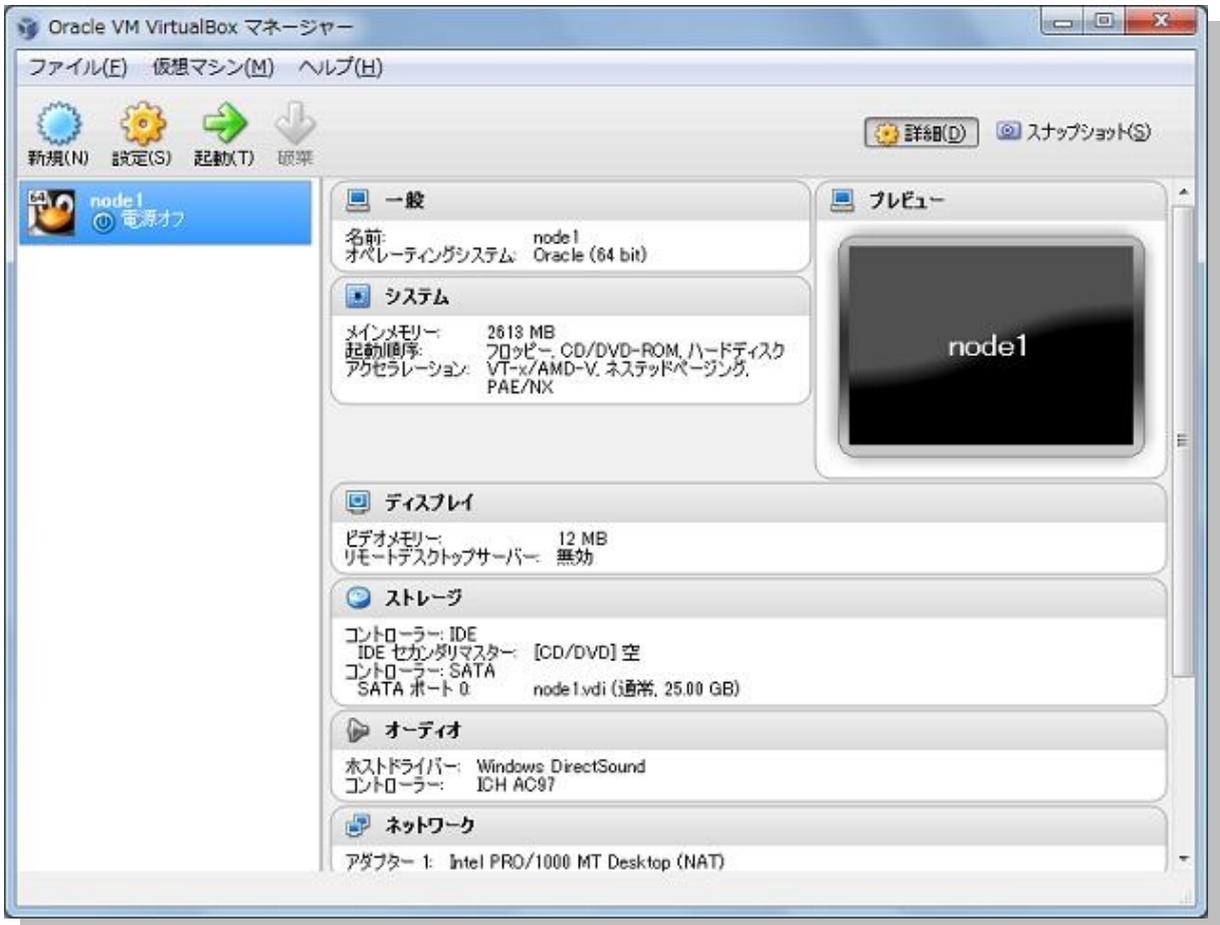


続いて、ファイルの配置場所とサイズを設定します。場所には「**node1**」を入力します。入力すると、今回はデフォルト仮想マシンフォルダーとして C:\%VBox を設定しているため C:\%VBox\node1.vdi が仮想ハードディスクとして作成されます。サイズには「**25.00GB**」を入力して、「**作成**」をクリックします。



## 6. 作成した仮想マシンの確認

仮想マシンの作成が完了すると、Oracle VM VirtualBox マネージャーに仮想マシンが表示されます。以下の画面では、作成した node1 が確認できます。ここまでで、仮想マシンの作成は完了です。



## 4. Oracle Linux 6 のインストールと再起動後における設定

続いて、作成した仮想マシンに Oracle Linux 6 Update 4 のインストールを行います。ここでは、インストールの事前準備から、インストールと再起動後に実施する設定についても以下の順で説明します。

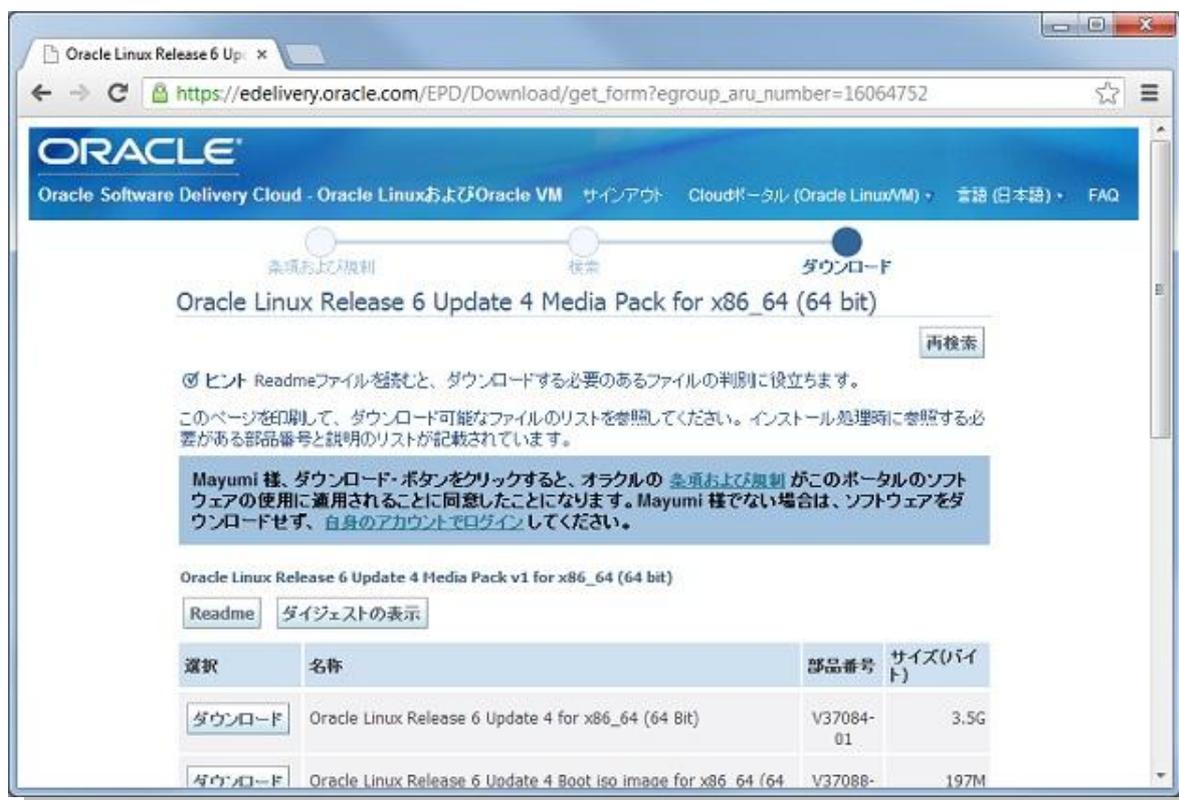
- 4.1 インストールの事前準備
- 4.2 Oracle Linux 6 のインストール
- 4.3 インストール後の設定
- 4.4 DNS サーバー用の仮想マシンの作成

### 4.1 インストールの事前準備

#### 1. ソフトウェアの準備

まず、仮想マシンへの Oracle Linux インストールに必要なソフトウェア・イメージをホスト OS 上に準備します。ここではダウンロード・ページ (URL : <https://edelivery.oracle.com/linux> ) より、有効なアカウントでログインをします。ログイン後はライセンス規定に合意して、ここでは次の製品をダウンロードします。

- Oracle Linux Release 6 Update 4 for x86\_64 (64 bit)

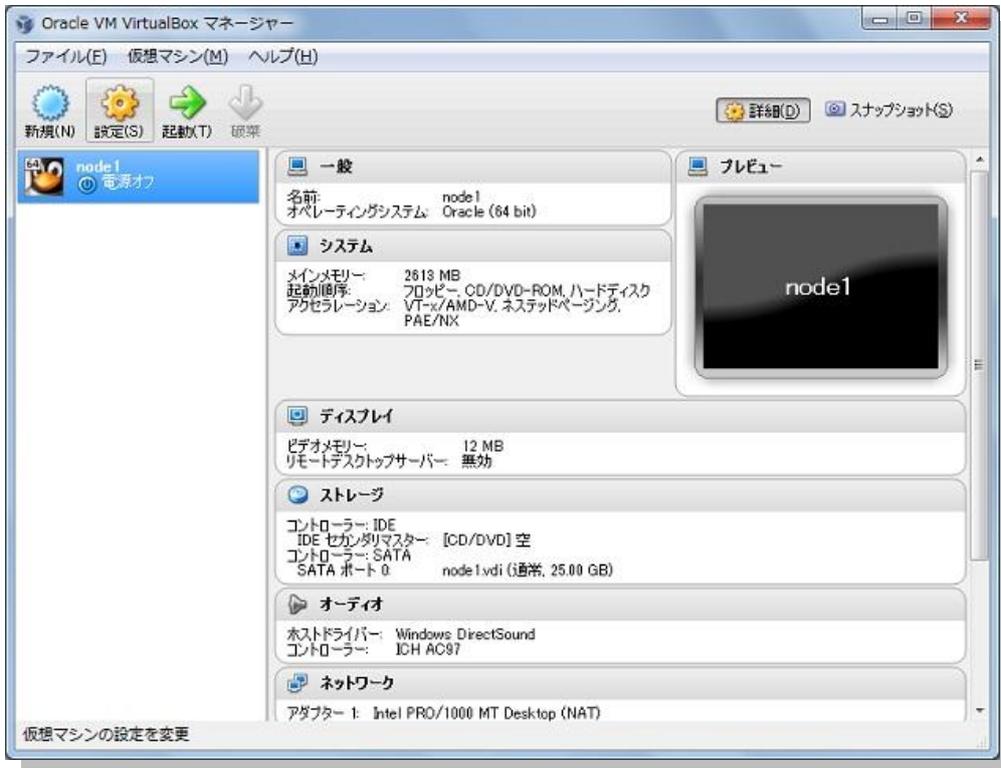


ここでは、ダウンロードしたファイルを以下の場所に配置して使用します。

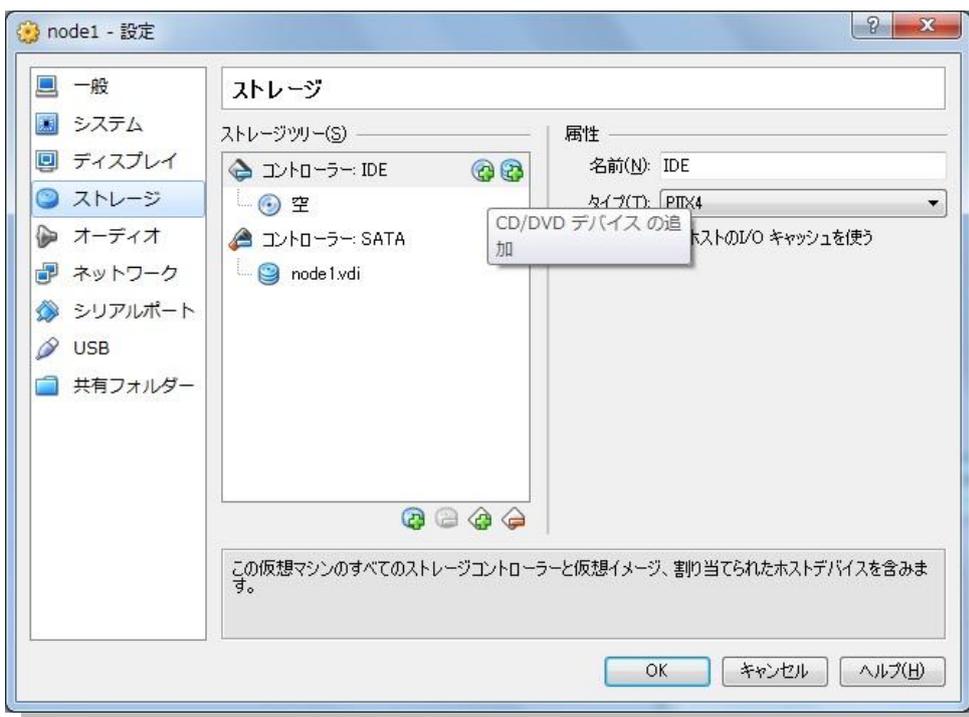
```
C:\software\V37084-01.iso
```

## 2. 仮想マシンのストレージ設定

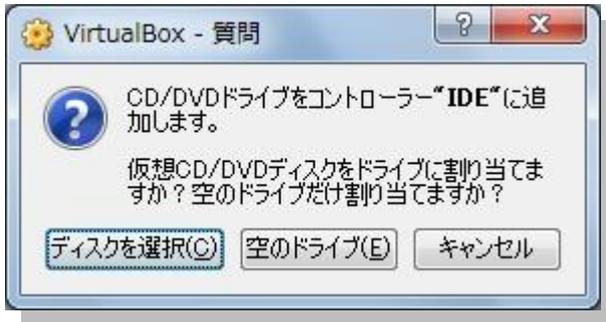
ダウンロードした OS のソフトウェア・イメージを仮想マシンから使用できるように、ストレージの設定を実施します。Oracle VM VirtualBox マネージャー画面から「設定」をクリックして設定画面を表示します。



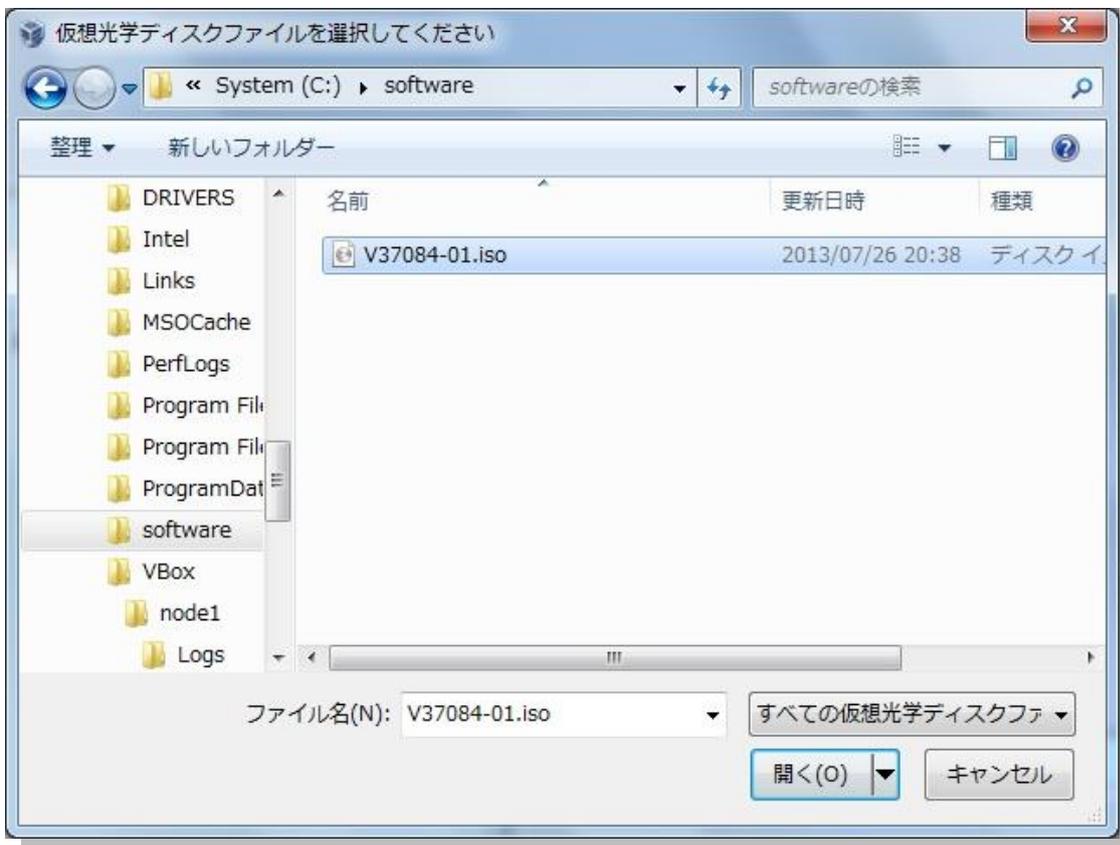
「ストレージ」の設定で IDE コントローラーの「CD / DVD デバイスの追加」アイコンをクリックして、CD / DVD ドライブを追加します。



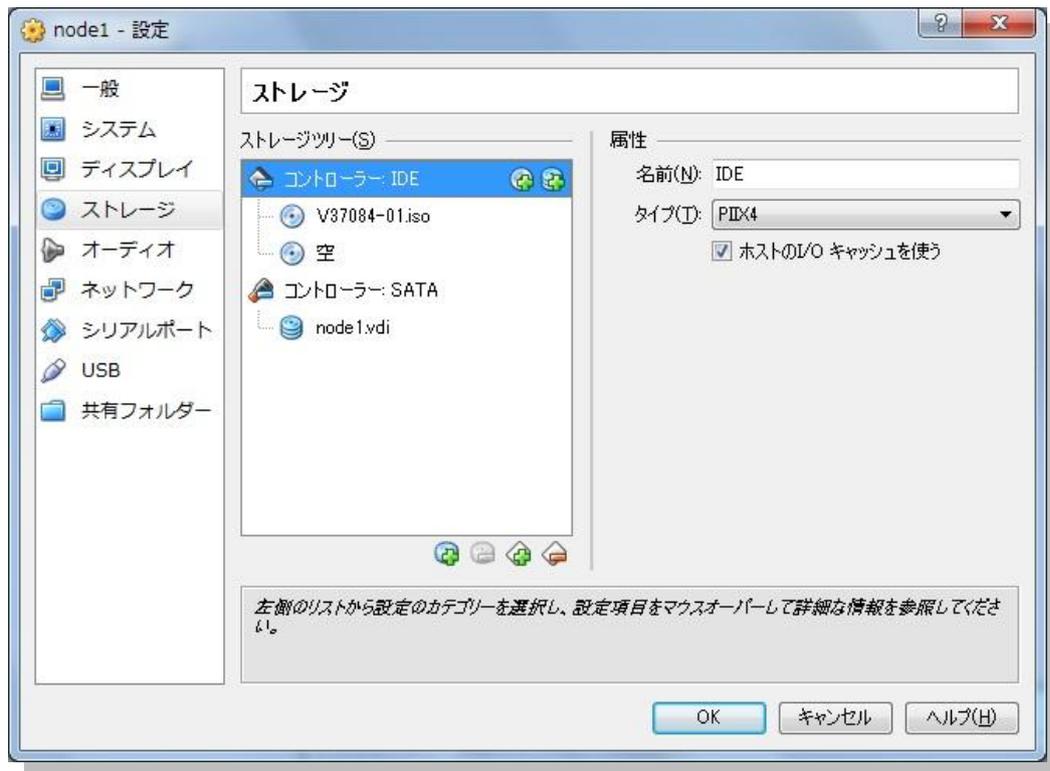
OS のソフトウェア・イメージを割り当てるために「ディスクを選択」をクリックして、仮想 CD / DVD ディスクを空のドライブに割り当てます。



準備したソフトウェア・イメージを選択します。ここでは、以下に配置しているソフトウェア・イメージを使用します。ファイルをダブルクリックするか、選択して「開く」をクリックします。



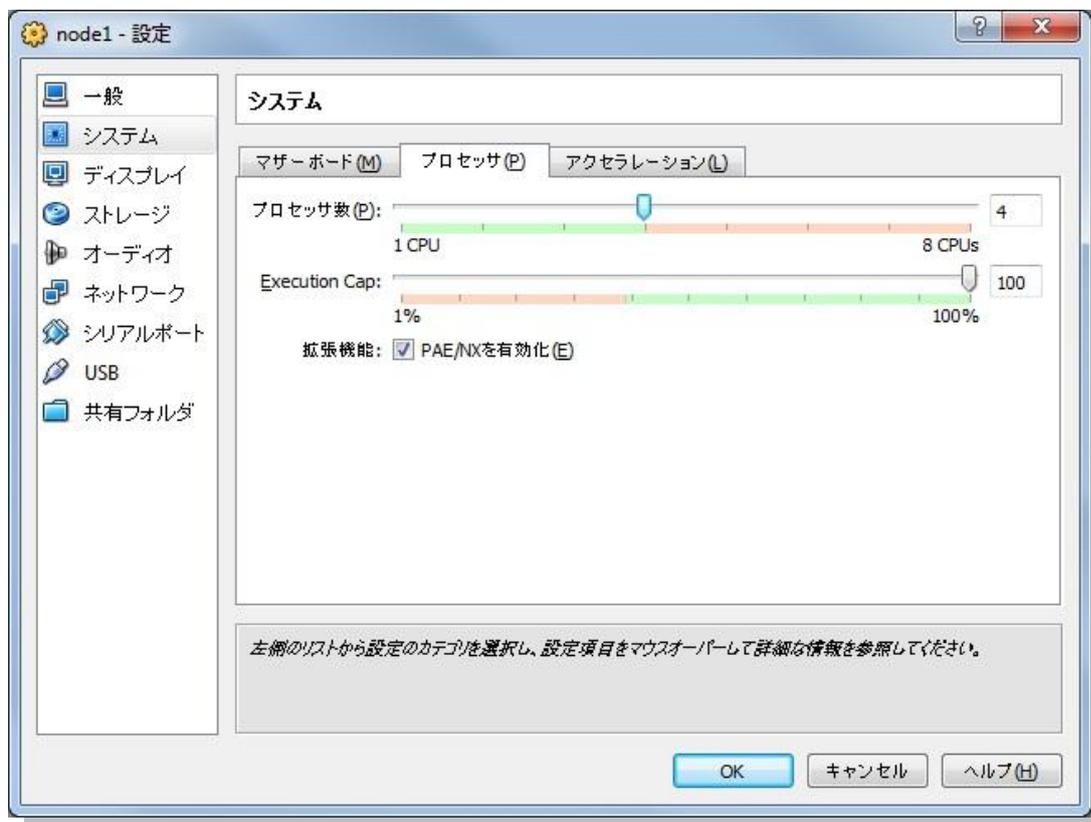
IDE コントローラーに追加したデバイス (V37084-01.iso) が表示されていることを確認します。



### 3. 仮想マシンのプロセッサ設定

続いて、仮想マシンのプロセッサ数の設定を変更しておきます。操作には、引き続き Oracle VM VirtualBox マネージャーの設定画面を使用します。設定画面の左側にある「システム」をクリックして、システムに関する設定画面を表示した後、「プロセッサ」タブをクリックして、ここではプロセッサ数を「4」に変更します。

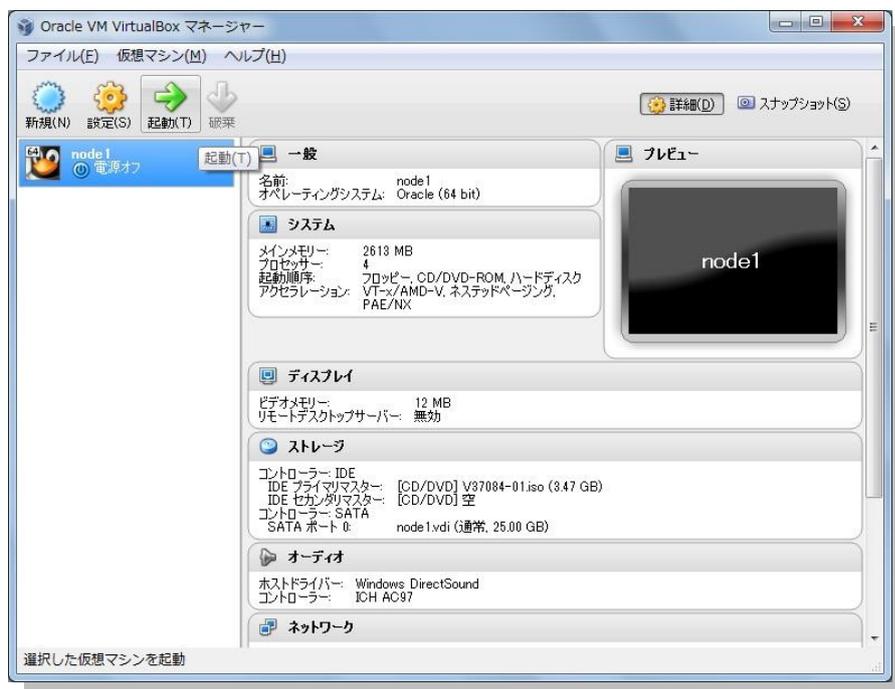
値は使用するマシンのスペックによって適宜変更してください。変更後、「OK」をクリックします。



変更後の確認として、Oracle VM VirtualBox マネージャー画面の右側に表示されている、システムのプロセッサを確認しておきます。

#### 4. 仮想マシンの起動

確認後、仮想マシンを起動します。node1 を選択して、「起動」をクリックします。



使用している物理マシンの設定によっては、仮想マシンの起動時に以下のエラーで起動できない場合があります。



この場合は Virtualization Technology の設定を確認して、有効化されていない場合には設定を変更します。以下に、本ガイドで使用している環境での対処方法を例として記載します。

1. 上記のエラー画面は「OK」をクリックして閉じます。
2. 使用しているノート PC 上で起動しているプログラム (Oracle VM VirtualBox を含む) をすべて終了して、マシンを正常終了 (シャットダウン) します。
3. マシンを起動して、BIOS の設定を以下のように変更します。

Step 1 : BIOS の設定画面を起動

Step 2 : Security のセクションより Virtualization を選択

Step 3 : Intel ® Virtualization Technology を有効に設定

Step 4 : 設定変更を保存して終了し、再起動の完了を待つ

## 4.2 Oracle Linux 6 のインストール

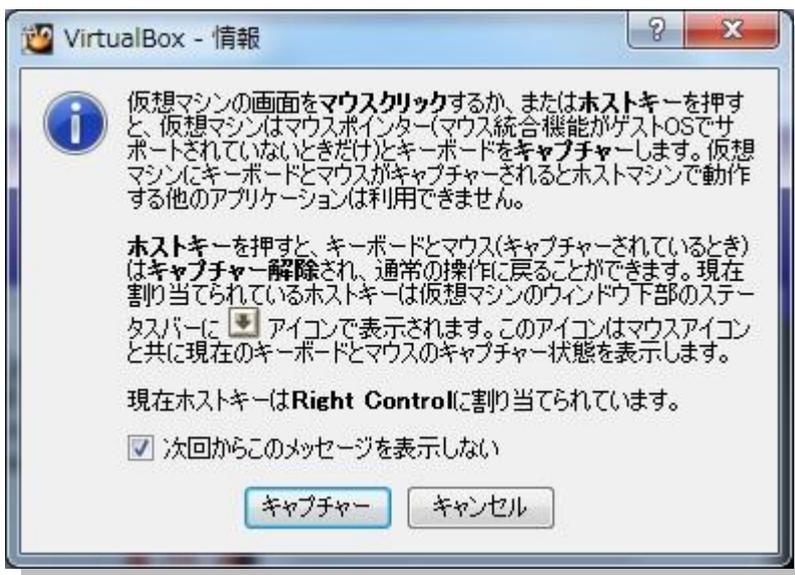
起動した仮想マシン (node1) に Oracle Linux 6 Update 4 をインストールします。以下に、インストール手順を記載します。

### 1. 情報の確認

仮想マシンが起動されると、以下の画面が表示されます。キーボードの自動キャプチャー機能が有効化されているという情報が表示された場合は、ホスト OS と仮想マシンのウィンドウの切り替えに使用するホストキーの設定を確認します。デフォルトでは、キーボードの右下にある Ctrl キーがホストキーとして割り当てられています。確認後、ここでは「次回からこのメッセージを表示しない」にチェック (☑) をして「OK」をクリックします。



また、以下の画面も確認を行い、ここでは「次回からこのメッセージを表示しない」にチェック (☑) をして「キャプチャー」をクリックします。



## 2. インストールの開始

仮想マシンが起動され、以下の画面が表示されたら「**Install or upgrade an existing system**」を選択してインストールを開始します。

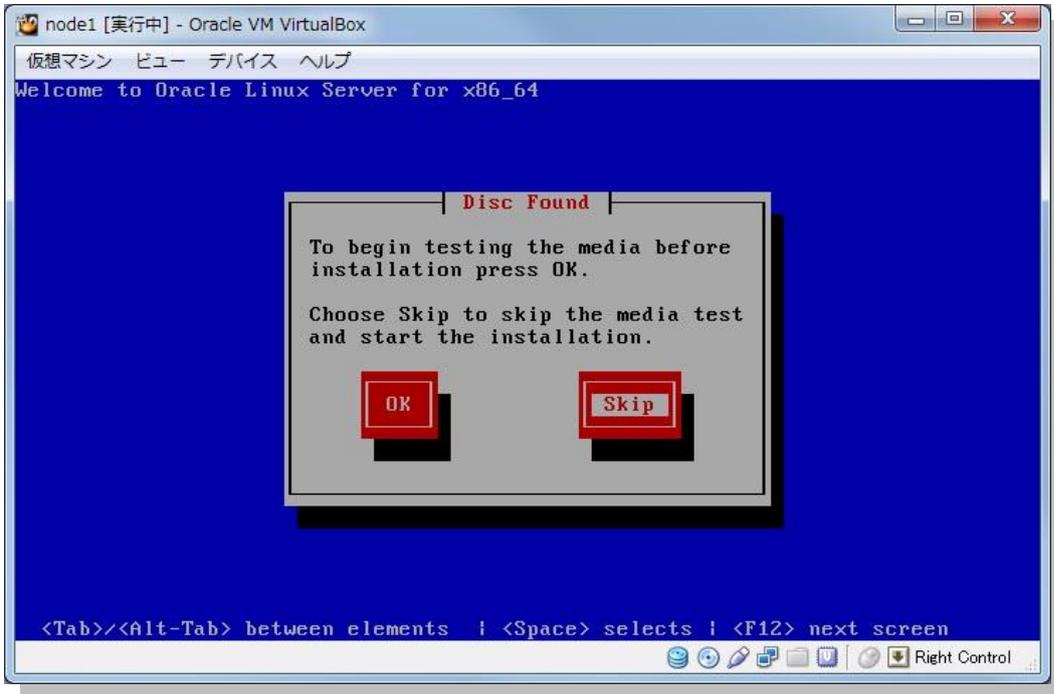


以下の画面が表示されたら、確認の上、ここでは「**次回からこのメッセージを表示しない**」にチェック (☑) をして「**OK**」をクリックします。



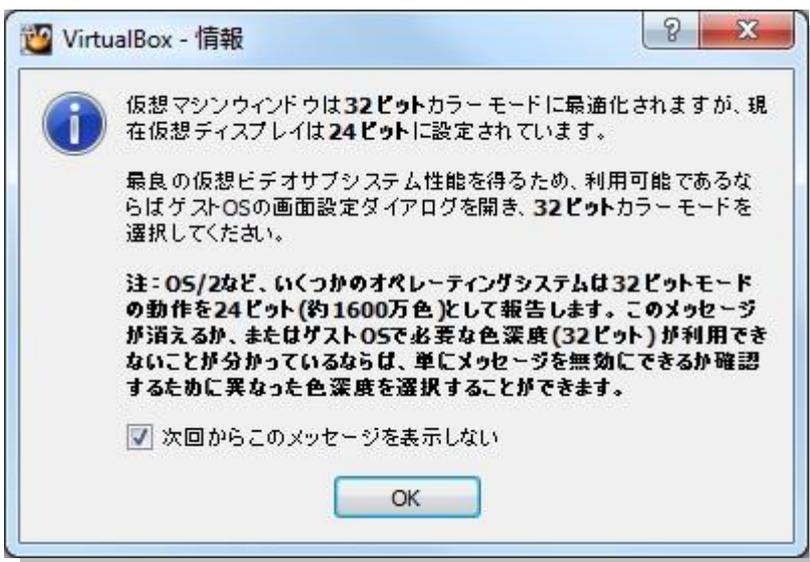
### 3. CD メディアの検証

インストールに使用するメディアの検証を選択します。ここでは、「Skip」を選択してメディアの検証をスキップするものとします。

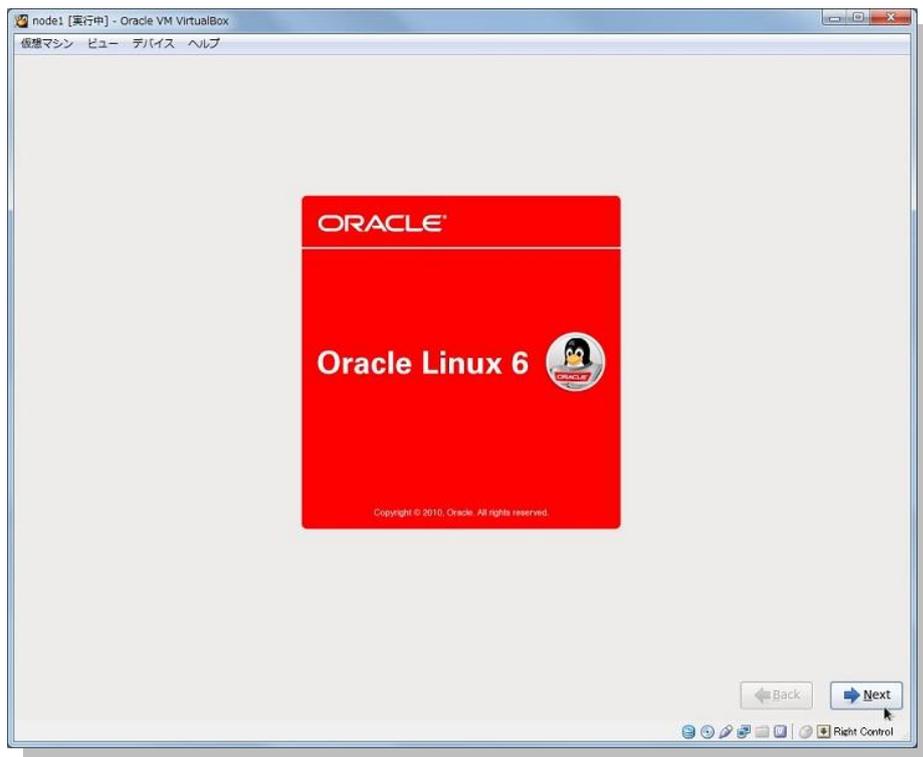


### 4. インストール画面の表示

以下の画面が表示されたら、仮想マシンのディスプレイについて確認します。ここでは「次回からこのメッセージを表示しない」をチェック (☑) して「OK」をクリックします。

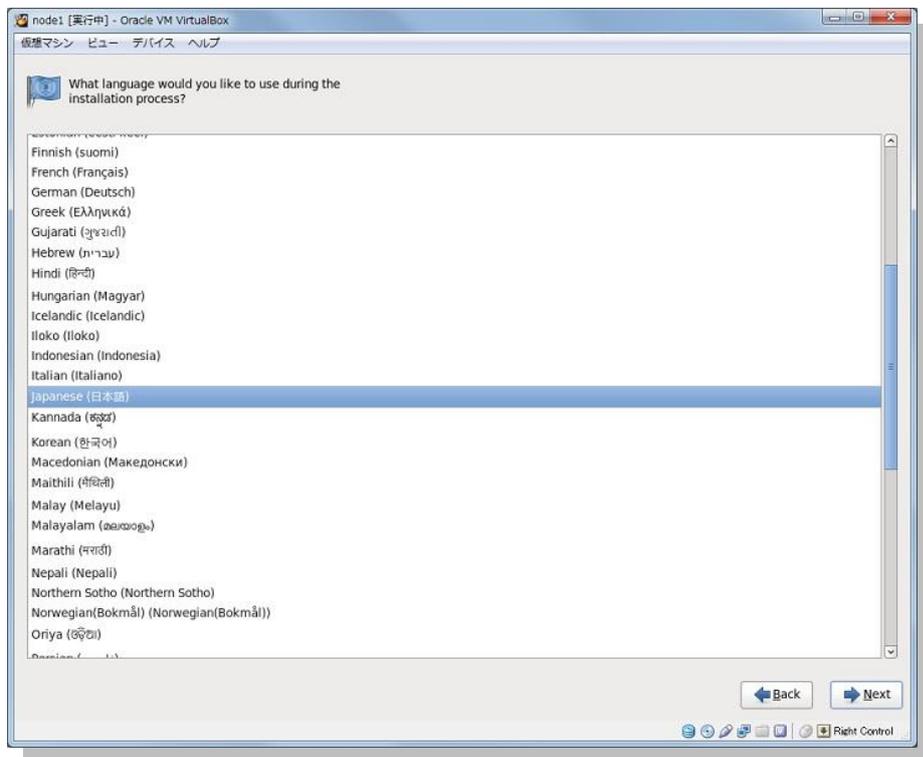


インストール画面が表示されたら「Next」をクリックします。



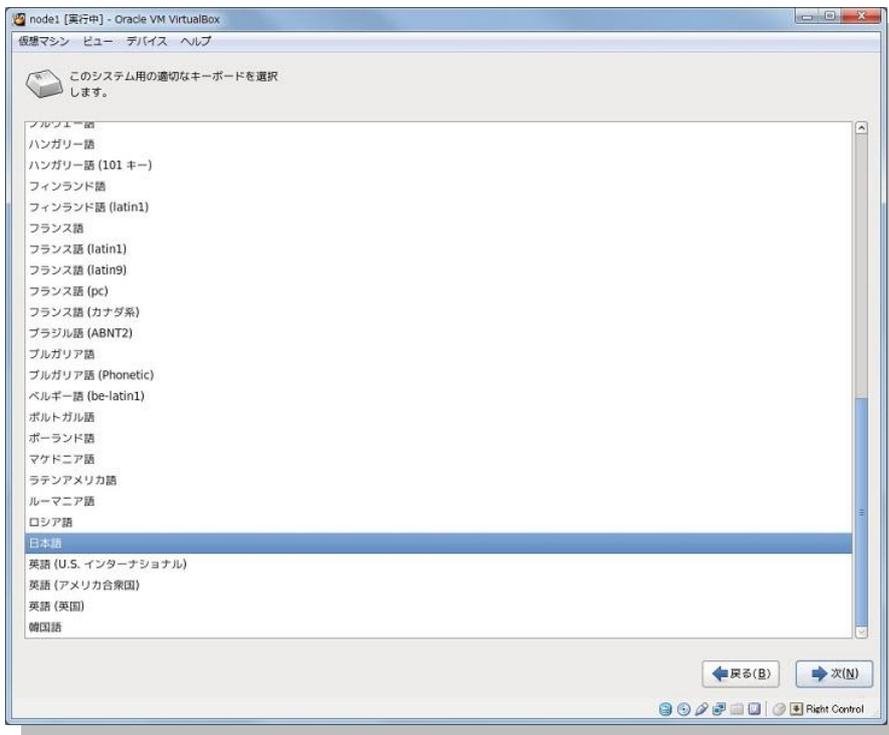
## 5. インストール言語の選択

続いて、インストール作業に使用する言語を選択します。ここでは「Japanese (日本語)」を選択して「Next」をクリックします。



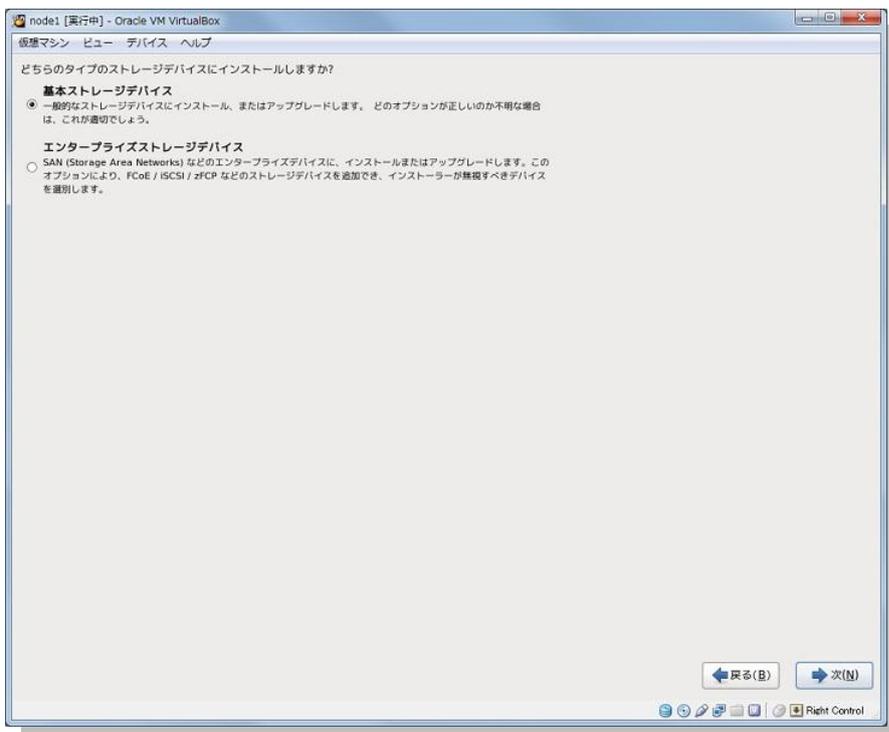
## 6. キーボードの選択

仮想マシンで使用するキーボードを設定します。ここでは「日本語」を選択して「次」をクリックします。



## 7. ストレージデバイスの選択

ストレージデバイスのタイプを設定します。ここでは、「基本ストレージデバイス」を選択の上、「次」をクリックします。

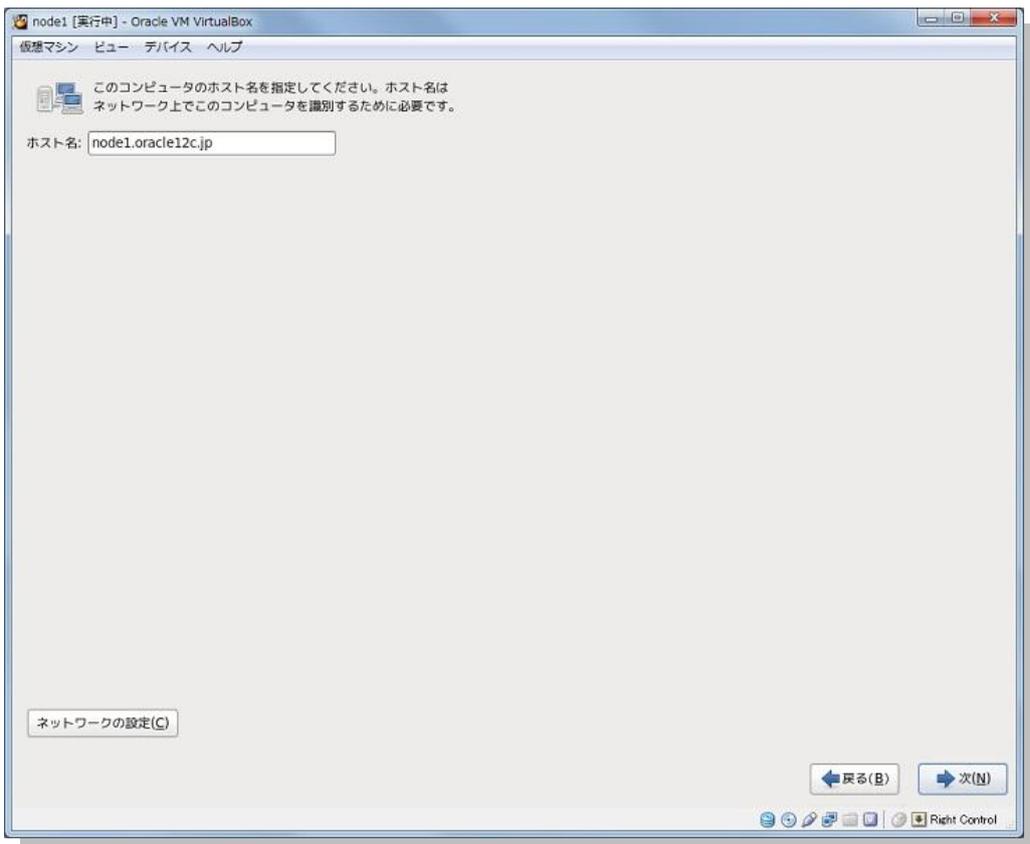


次の確認画面が表示されたら、「はい。含まれていません。どのようなデータであっても破棄してください。」をクリックして続けます。



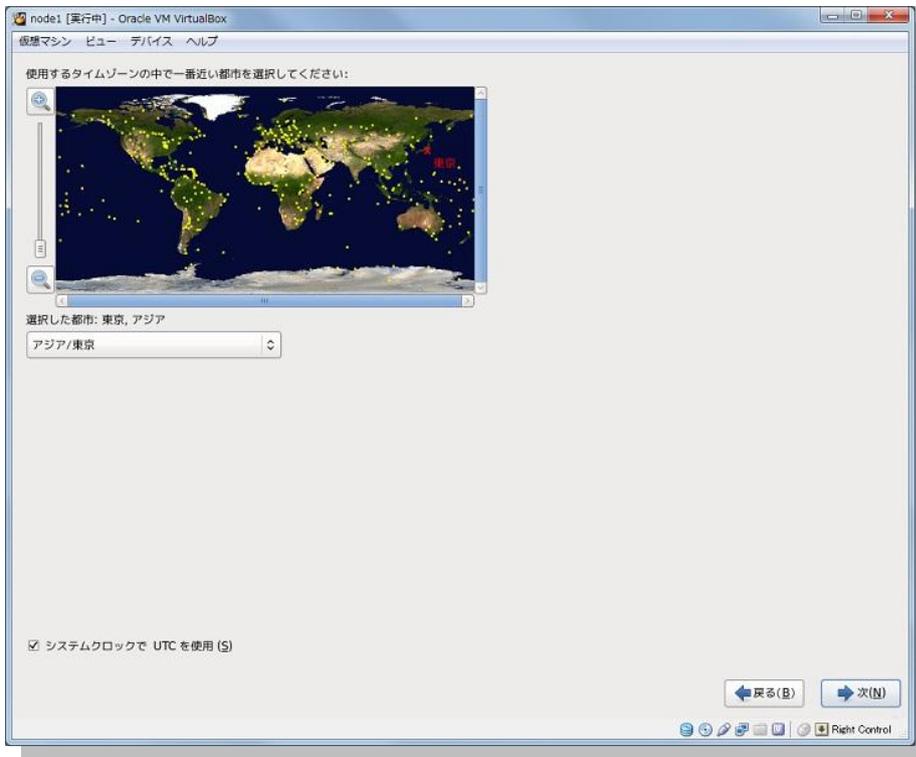
## 8. ホスト名の設定

ホスト名を設定します。ここでは「node1.oracle12c.jp」と設定して「次」をクリックします。



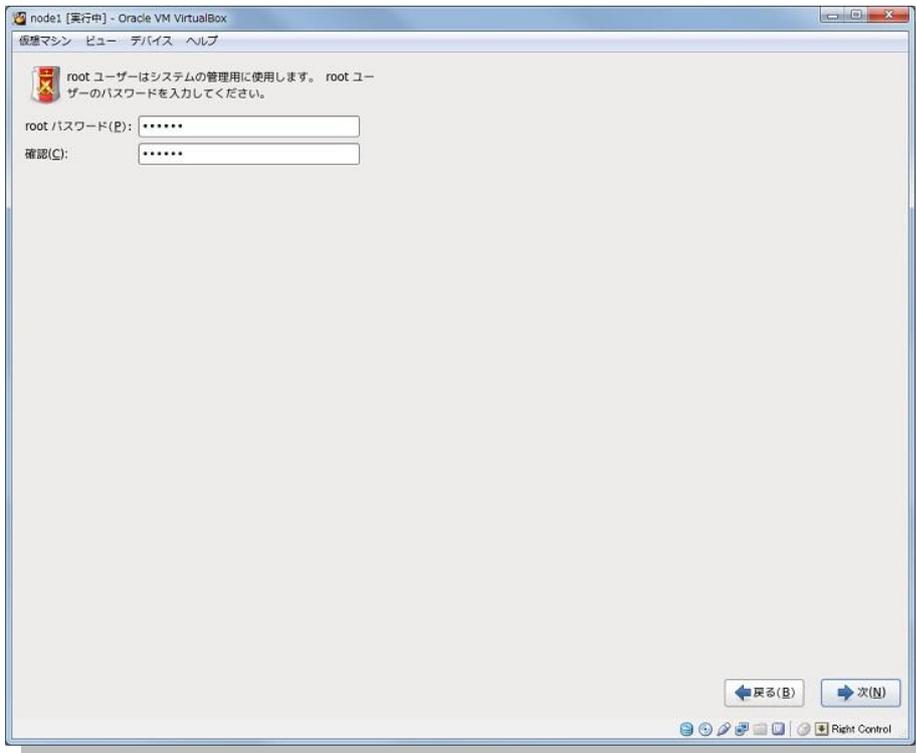
## 9. 地域の設定

地域とシステムクロックを設定します。ここでは、表示されている設定のまま「次」をクリックします



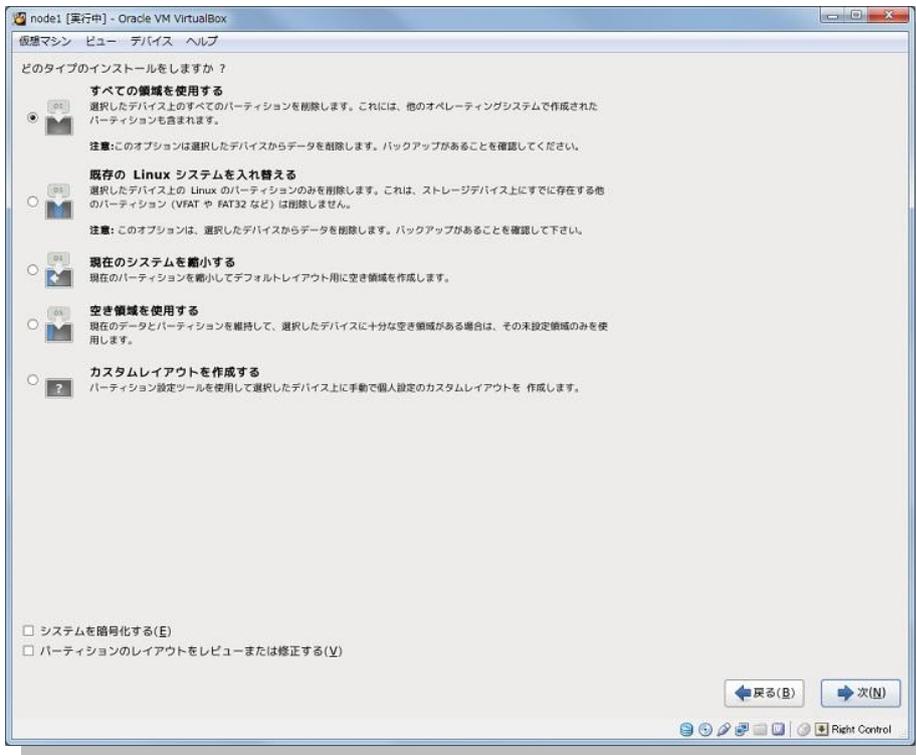
## 10. root アカウントの設定

root ユーザーのパスワードを設定します。任意のパスワードを入力して「次」をクリックします。

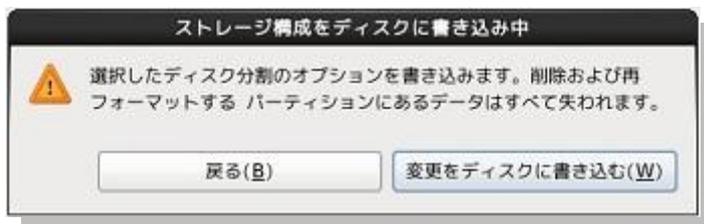


## 11. インストール・タイプの選択

実行するインストールのタイプを選択します。ここでは「すべての領域を使用する」を選択して「次」をクリックします。

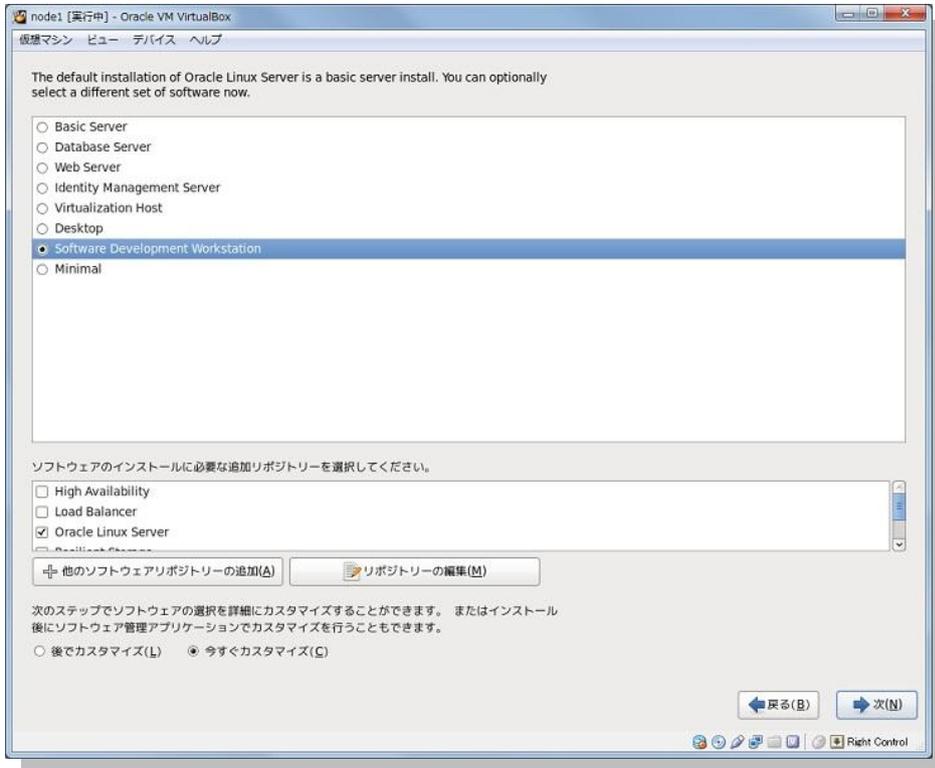


書き込みの確認が表示されたら、「変更をディスクに書き込む」をクリックして続行します。



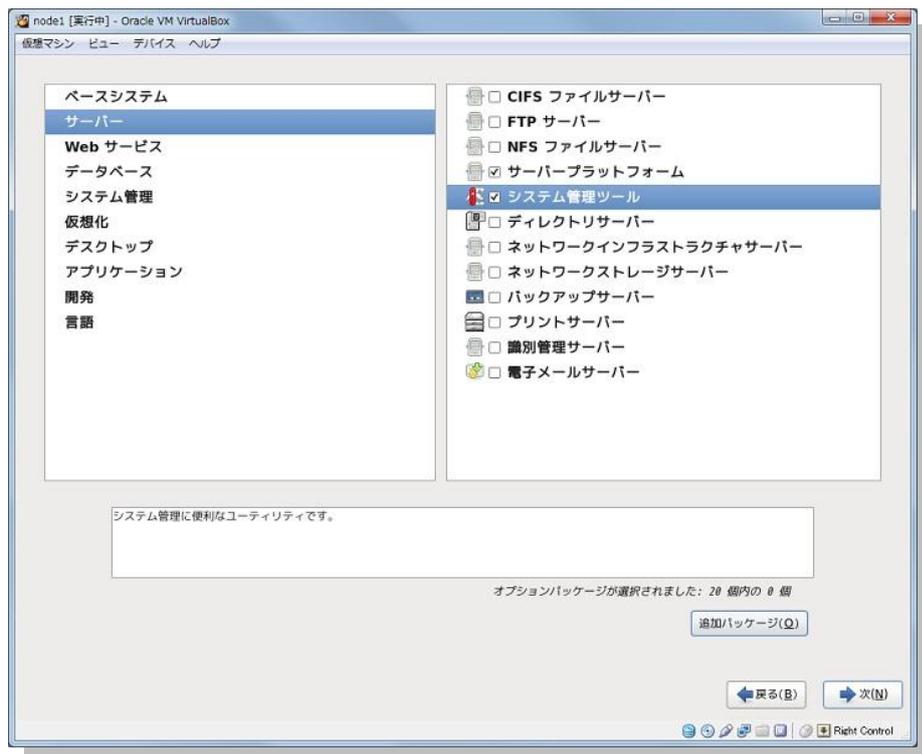
## 12. ソフトウェアの設定

インストールするソフトウェアを選択します。ここでは、「**Software Development Workstation**」を選択します。また、「**今すぐカスタマイズする**」を選択して、追加インストールするソフトウェアのより詳細な設定を実施します。選択後は「次」をクリックします。



## 13. ソフトウェアのカスタマイズ

追加インストールするソフトウェアのより詳細な設定を実施します。ここでは、「サーバー」の「システム管理ツール」をチェック (☑) して、「追加パッケージ」をクリックします。



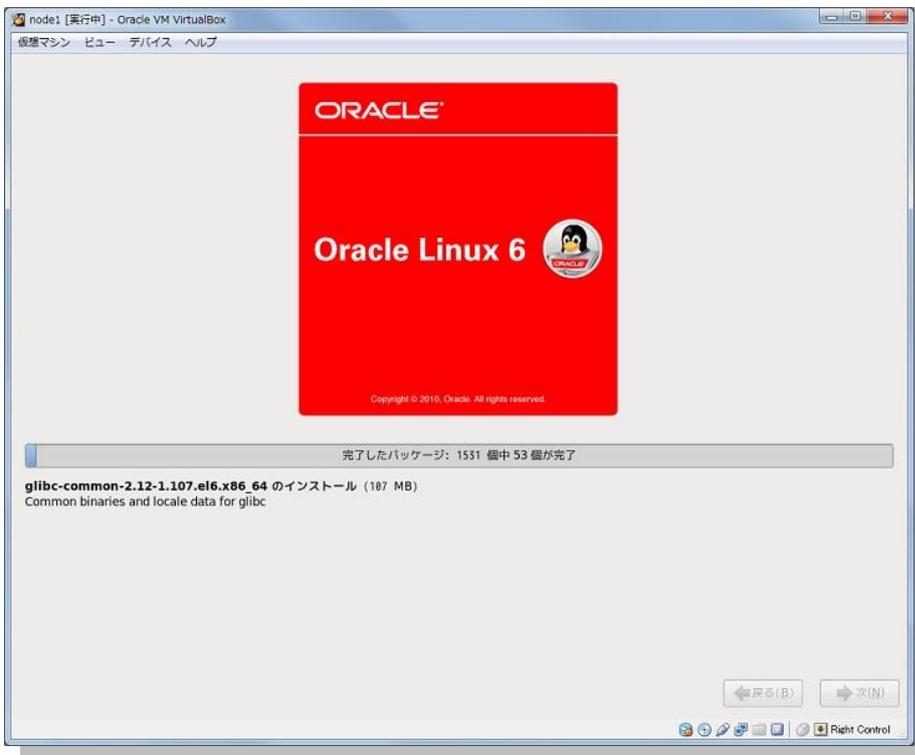
ここでは Oracle Validated RPM パッケージをインストールして Oracle Database のインストールに必要な構成の一部 (oracle ユーザーおよび OS グループの作成、追加パッケージのインストール、sysctl.conf の設定など) を実施するものとします。

「**oracle-rdbms-server-11gR2-preinstall-1.0.7.el6.x86\_64**」をチェック (☑) して「閉じる」をクリックします。12c 用の Oracle Pre-Install RPM パッケージ (oracle-rdbms-server-12cR1-preinstall) は別途 Oracle public yum リポジトリなどから入手できます。ここでは同梱されている 11g 用を使用します。



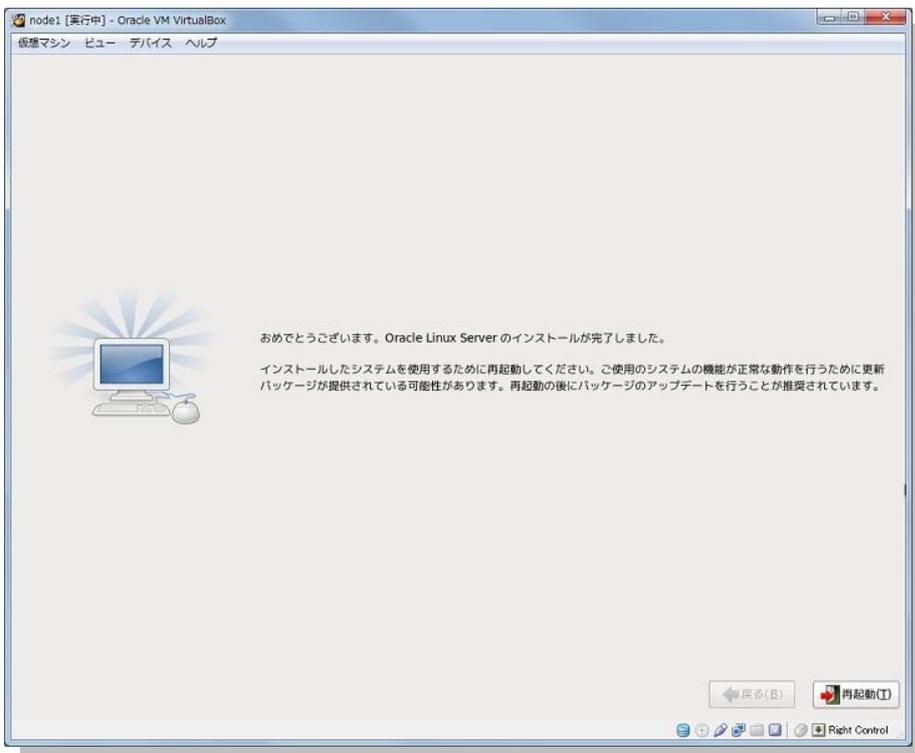
## 14. インストールの開始

「次」をクリックして、インストールを開始します。



## 15. インストールの完了

以下の画面が表示されたらインストールは完了です。「再起動」をクリックして、システムを再起動します。



## 16. ようこそ

インストール後のシステム設定を実施します。「進む」をクリックします。



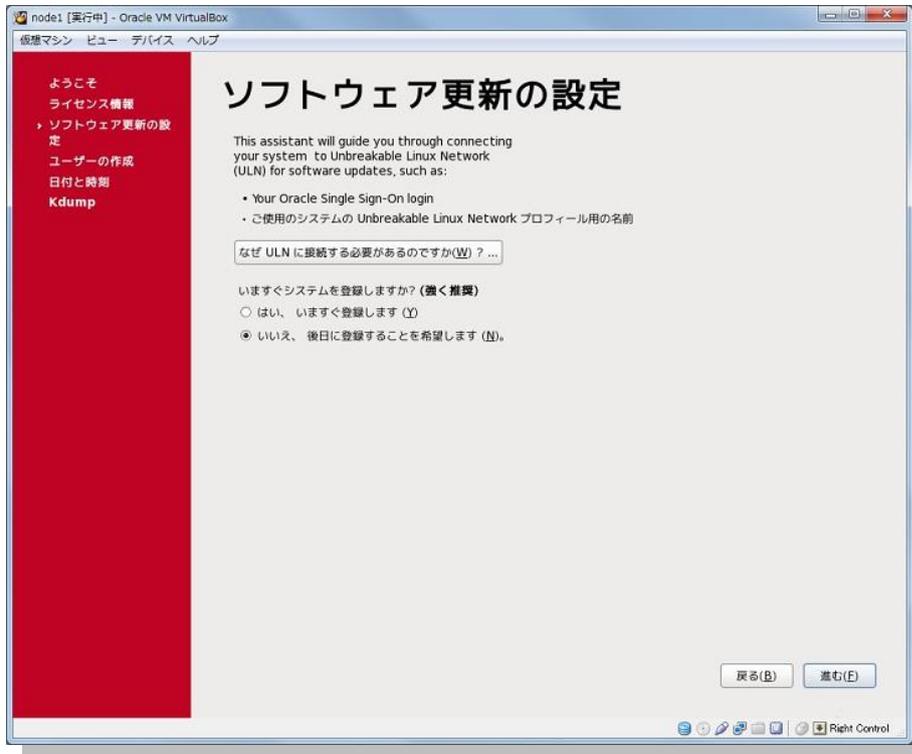
## 17. ライセンス同意書

ライセンス同意書の内容を確認して、「進む」をクリックします。



## 18. ソフトウェアの更新

ソフトウェア更新の設定を実施します。ここでは「いいえ、後日に登録することを希望します」を選択して「進む」をクリックします。



確認のためメッセージが表示されますので、確認の上「いいえ、後で接続します」をクリックします。



続いて、完了画面で「進む」をクリックします。

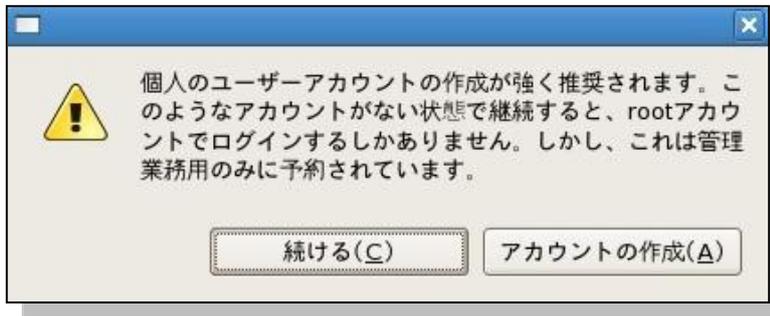


## 19. ユーザーの作成

root ユーザー以外のユーザーの作成を行います。ここでは特に作成は行いませんので、「進む」をクリックします。

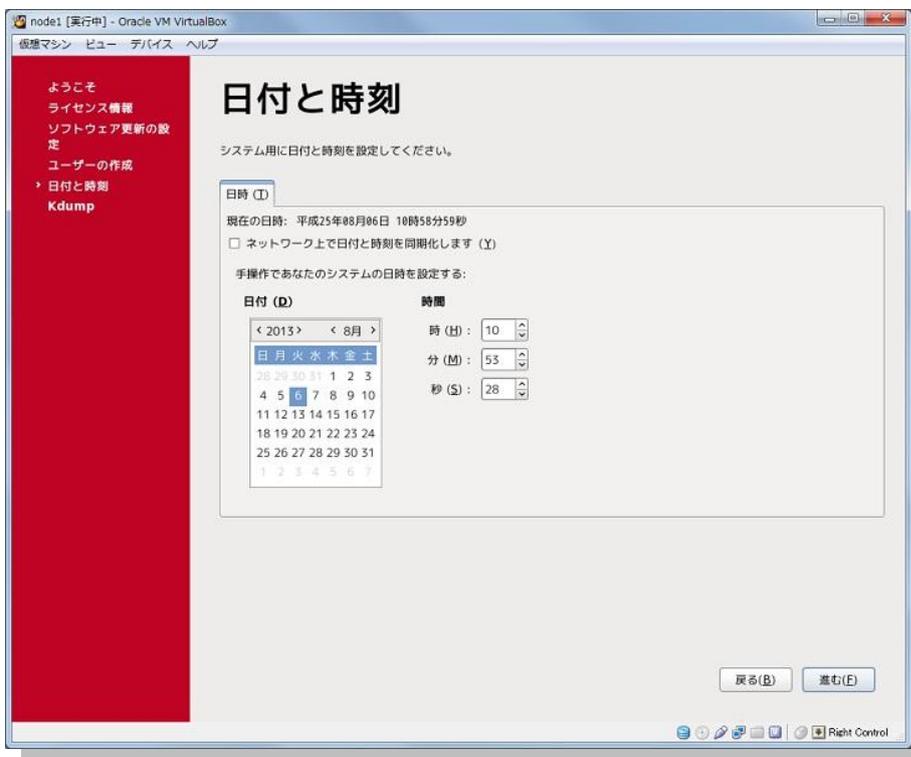


確認のためメッセージが表示されますので、確認の上「続ける」をクリックします。



## 20. 日付と時刻

日付と時刻を設定します。現在の時刻を確認して必要であれば適宜修正を行い、「進む」をクリックします。



## 21. Kdump

Kdump を設定します。ここでは、特に有効化せずに作業を続行しますので「終了」をクリックして、システムを再起動します。



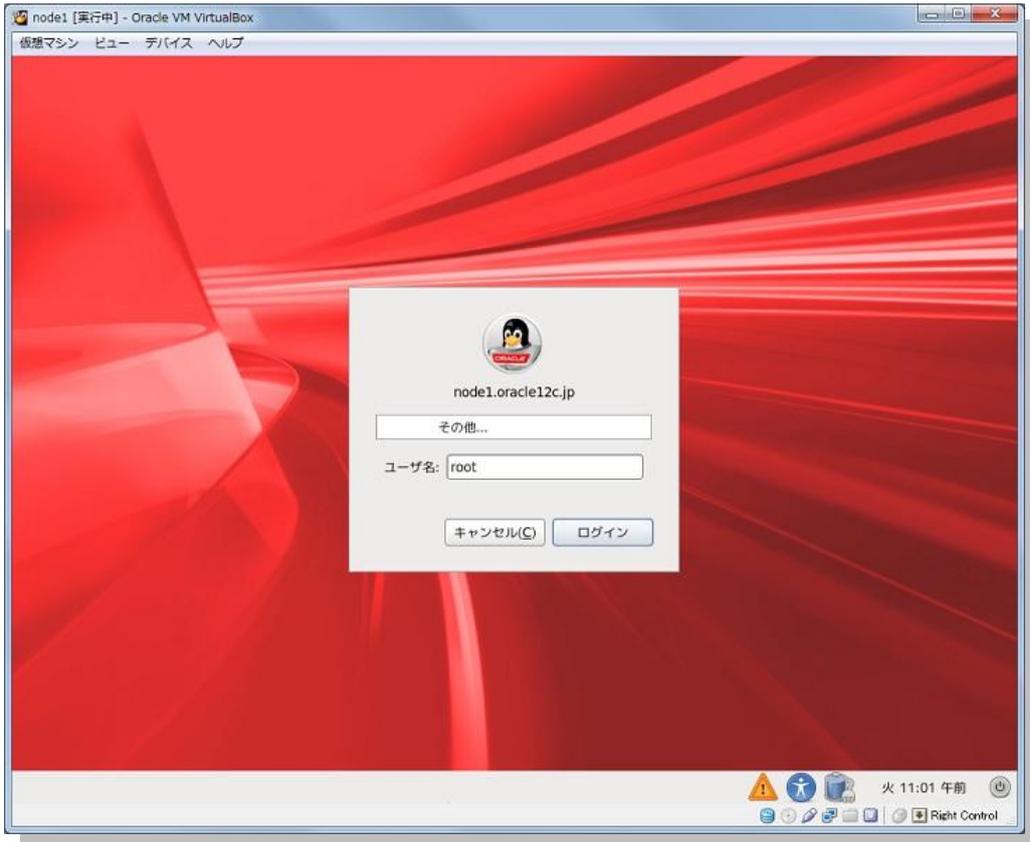
## 4.3 インストール後の設定

仮想マシンへの Oracle Linux インストール後の設定として、Oracle VM VirtualBox Guest Additions のインストールとネットワークの設定などを実施します。

### 1. Oracle VM VirtualBox Guest Additions のインストール

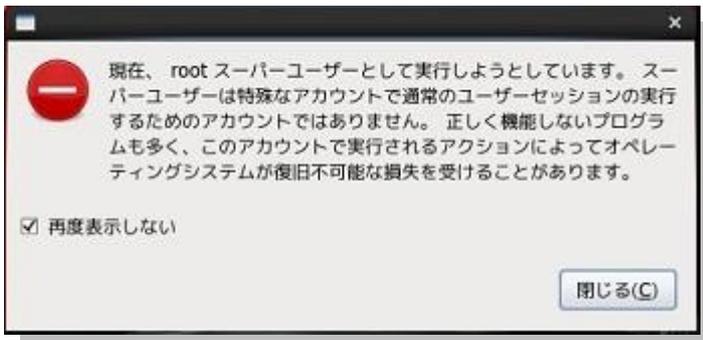
Oracle Linux のインストールを行い、再起動した仮想マシンに root ユーザーでログインします。

「その他」を選択し、ユーザー名に「root」と入力して「ログイン」をクリックします。

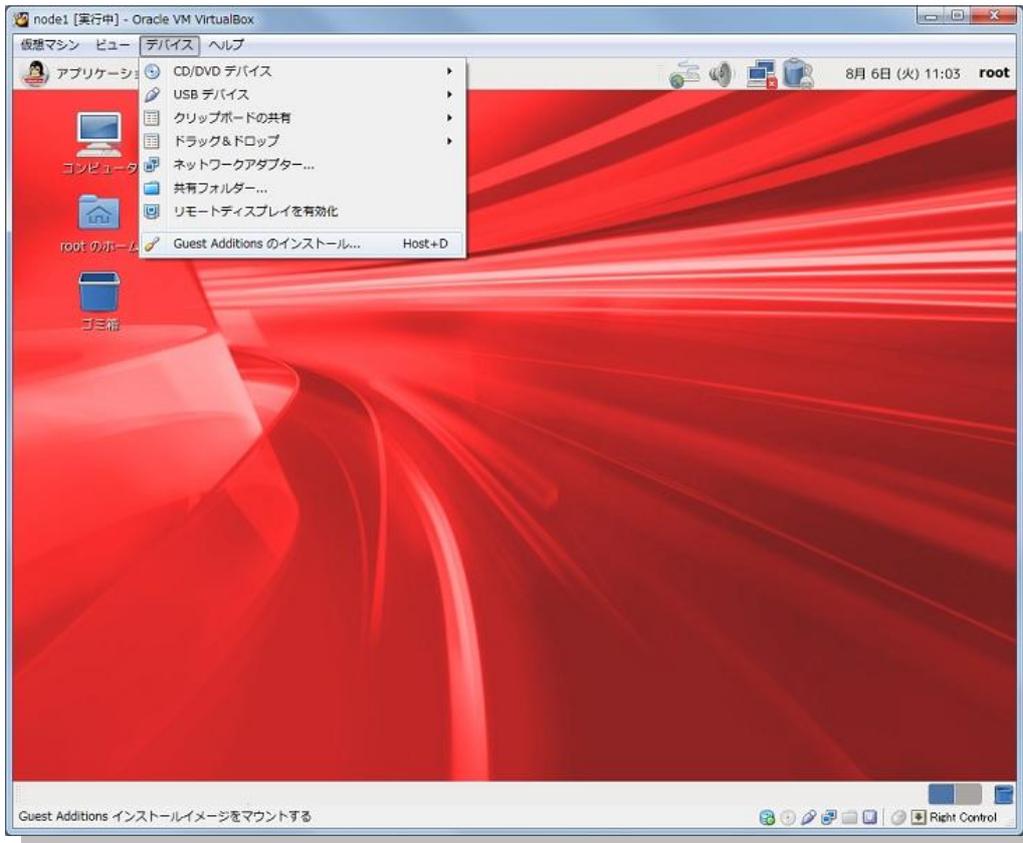


続いて、root ユーザーに設定したパスワードを入力してログインします。

ログイン後、次のような警告が表示された場合は確認の上「再度表示しない」をチェック (☑) して「閉じる」をクリックします。



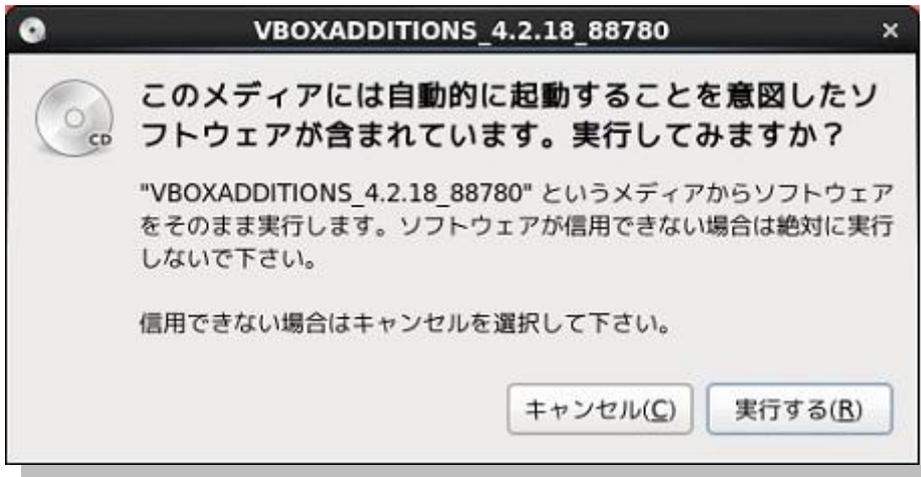
ログイン後は、「デバイス」から「Guest Additions のインストール」を選択します。



表示されたメッセージを確認して「OK」をクリックします。



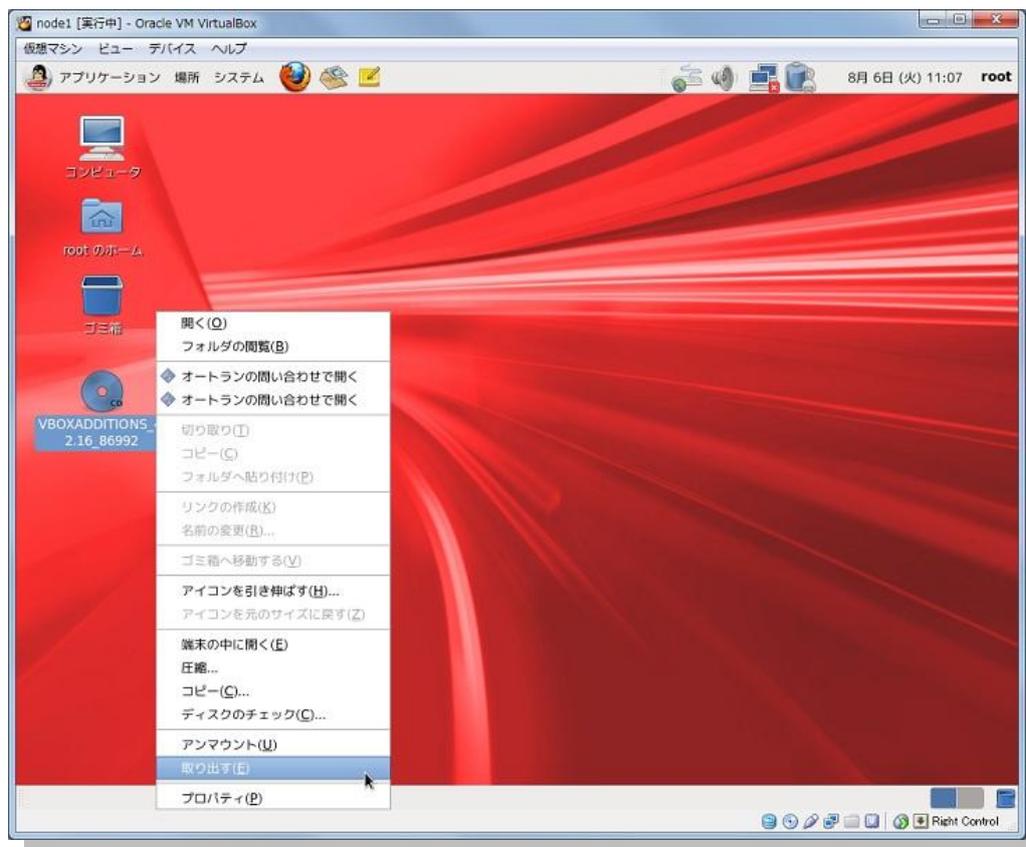
続いて、確認メッセージが表示されますので「**実行する**」をクリックします。新たに端末が開かれ、その端末内でインストールが実行されます。



実行が完了したら、Return キーを入力して完了です。



画面上に表示されている Oracle VM VirtualBox Guest Additions のイメージも取り出しておきます。イメージの取り出しは、アイコンを右クリックしてメニューを表示し、その中から「取り出す」を選択します。

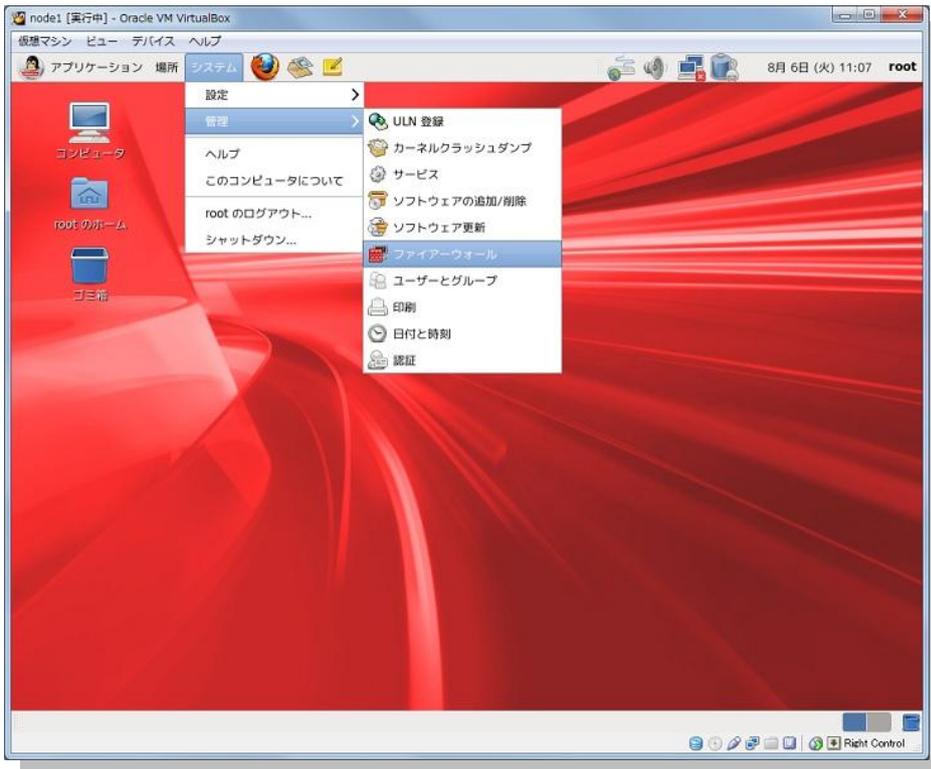


## 2. ファイアーウォールと Security-Enhanced Linux (SELinux) の無効化

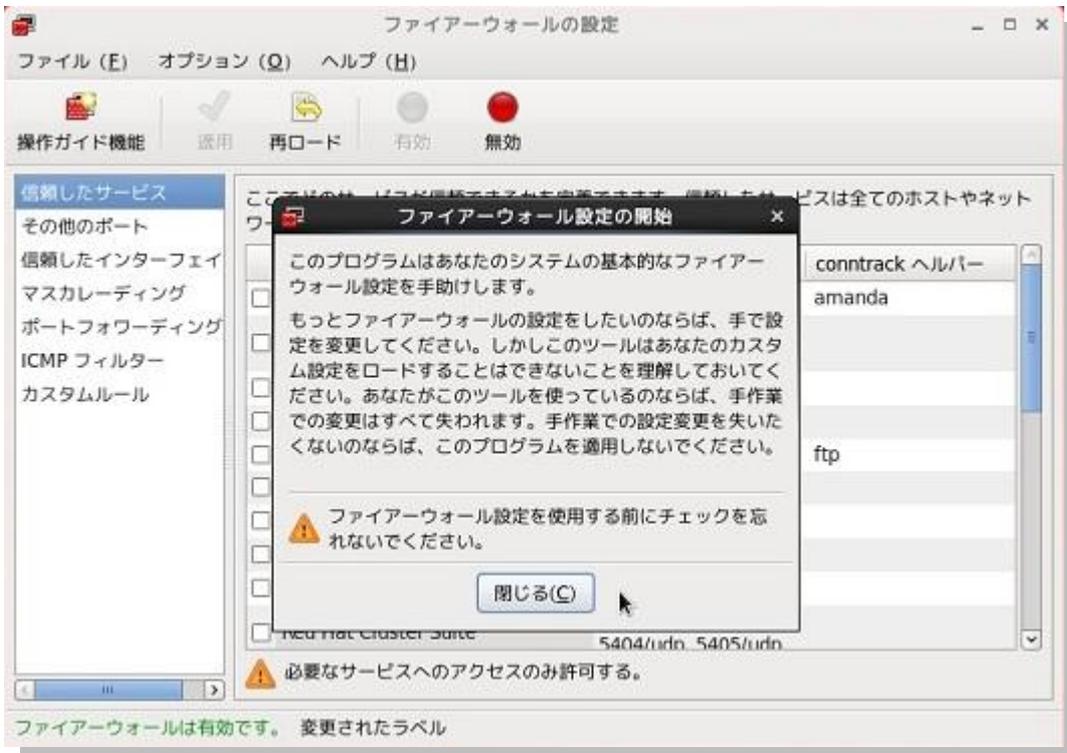
Oracle Linux 6 Update 4 では、OS インストール時にファイアーウォールと SELinux の設定を変更することができません。OS インストール直後はどちらも有効な状態になっています。これらを無効にする設定を実施します。

(補足) Oracle Database 12c Release 1 は SELinux に対応しているため SELinux の無効化は必須ではありません。今回は、検証環境として構築するため無効に設定するものとします。なお、SELinux は Oracle ACFS ファイル・システムではサポートされていません。

「システム」メニューの「管理」の中から「ファイアーウォール」をクリックします。



「ファイアーウォールの設定の開始」画面が表示されますので、「閉じる」をクリックします。



表示された設定画面で、「無効」をクリックします。



続いて「適用」をクリックします。



確認ウィンドウが表示されますので、「はい」をクリックします。



「ファイル」メニューから「終了」を選択し、設定完了です。



また、再起動時に起動しないように自動起動の設定を無効にしておきます。root ユーザーで次のコマンドを実行します。

※ コマンドを実行するための端末は、Oracle VM VirtualBox 画面上から「アプリケーション」>「システムツール」>「端末」を選択して用意できます。

```
# chkconfig iptables --list
```

```
# chkconfig iptables off
```

```
# chkconfig iptables --list
```

## &lt;実行例&gt;

```
[root@node1 ~]# chkconfig iptables --list
iptables    0:off  1:off  2:on   3:on   4:on   5:on   6:off
[root@node1 ~]# chkconfig iptables off
[root@node1 ~]# chkconfig iptables --list
iptables    0:off  1:off  2:off  3:off  4:off  5:off  6:off
```

本ガイドで構築する環境は、検証用途が目的であるため、SELinux は無効に設定します。SELinux の無効化は設定ファイルの編集で行います。端末を起動し、root ユーザーで編集を実施します。

```
# vi /etc/selinux/config
```

<記述例> ※「**enforcing**」となっている行をコメントアウトし、新たに「**disabled**」の行を追記します。

```
#SELINUX=enforcing
SELINUX=disabled
```

今回は、使用しないネットワーク・インターフェースを削除しておきます。

```
# virsh net-list --all
```

```
# virsh net-destroy default
```

```
# virsh net-autostart default --disable
```

```
# virsh net-list --all
```

## &lt;実行例&gt;

```
[root@node1 ~]# virsh net-list --all
名前                状態    自動起動    永続
-----
default             動作中   はい (yes)   はい (yes)
[root@node1 ~]# virsh net-destroy default
ネットワーク default は強制停止されました
[root@node1 ~]# virsh net-autostart default --disable
ネットワーク default の自動起動設定が解除されました
[root@node1 ~]# virsh net-list --all
名前                状態    自動起動    永続
-----
default             停止状態 いいえ (no)   はい (yes)
```

### 3. ネットワークの設定

続いて、仮想マシンのネットワーク設定を変更します。「システム」メニューの「設定」の中から「ネットワーク接続」をクリックします。



「System eth0」を選択して「編集」をクリックします。



接続名を「eth0」に変更して、「自動接続する」にチェック (☑) します。続いて「IPv4 の設定」タブを選択して方式を「手動」に変更後、「追加」ボタンをクリックして次のようにアドレスや DNS サーバーの情報を入力します。

- アドレス : 192.168.56.101
- ネットマスク : 255.255.255.0
- DNS サーバー : 192.168.56.254
- ドメインを検索 : oracle12c.jp

設定後、「適用」をクリックします。



ネットワーク接続の画面は、「閉じる」をクリックして閉じます。

続いて Oracle VM VirtualBox マネージャー画面より設定作業のため、一旦仮想マシンを停止します。ここでは、以下のコマンドを root ユーザーで実行して仮想マシンを正常終了します。

```
# shutdown -h now
```

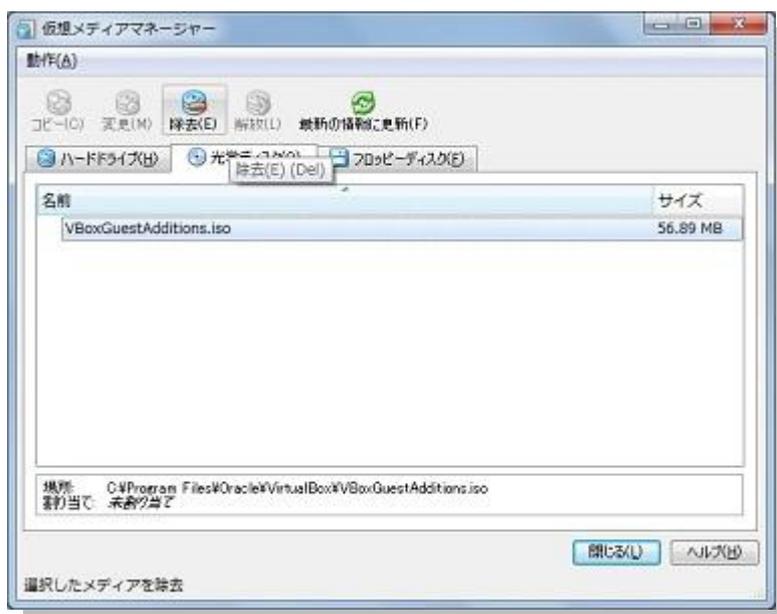
※ 仮想マシンの停止は、「システム」メニューの「シャットダウン」からも実行することができます。

#### 4. 使用済みイメージの削除

仮想マシンの停止後、Oracle VM VirtualBox マネージャーの「ファイル」メニューより、「仮想メディアマネージャー」を選択します。



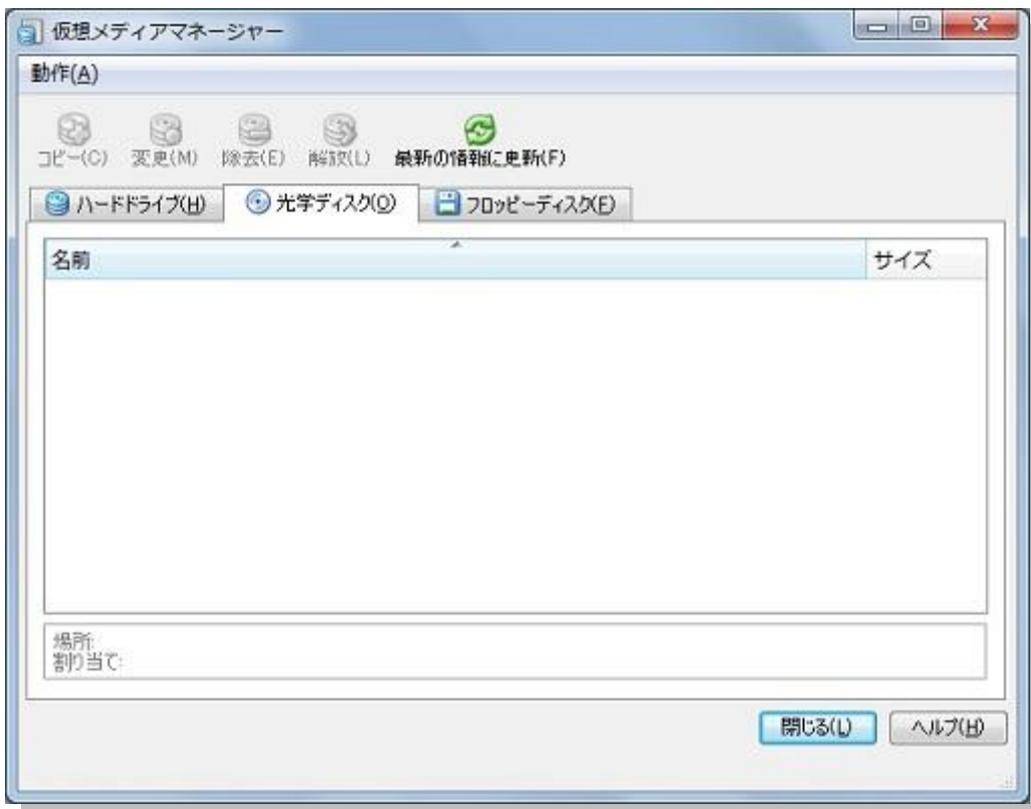
仮想メディアマネージャー画面で、「光学メディア」タブを選択し、Oracle Linux および VirtualBox Guest Additions のイメージをそれぞれ選択して「除去」をクリックします。



確認メッセージが表示されたら「**除去**」をクリックして続けます。



すべてのイメージを削除した後、「**閉じる**」で終了します。

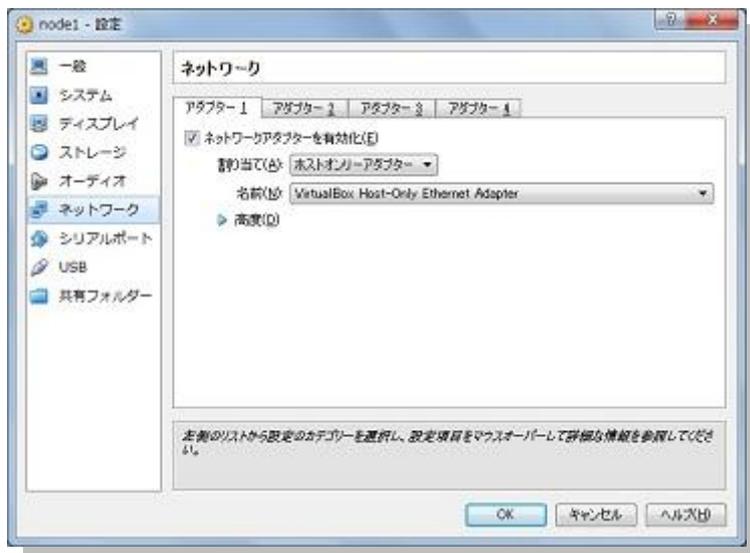


## 5. ネットワークの設定

Oracle VM VirtualBox マネージャー画面の「ネットワーク」(「ネットワーク」セクションの名前部分)をクリックするか、「設定」をクリックして設定画面を表示します。



ネットワークの設定として、アダプター1の設定を変更します。割り当てを「**ホストオンリーアダプター**」に変更して、名前には「**VirtualBox Host-Only Ethernet Adapter**」を選択します。設定後、「OK」をクリックします。



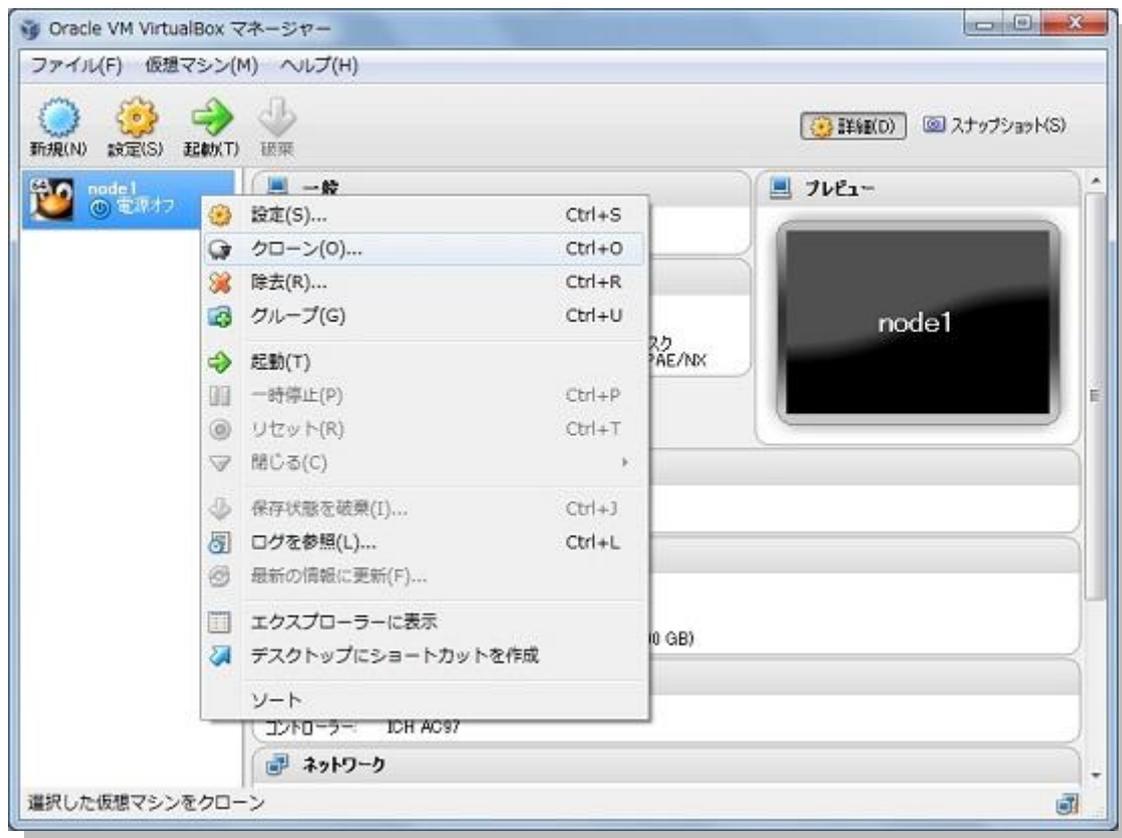
## 4.4 DNS サーバー用の仮想マシンの作成

DNS サーバーとして利用する仮想マシンを準備します。ここでは、node1 をクローンして DNS サーバー用の仮想マシンを準備するものとします。

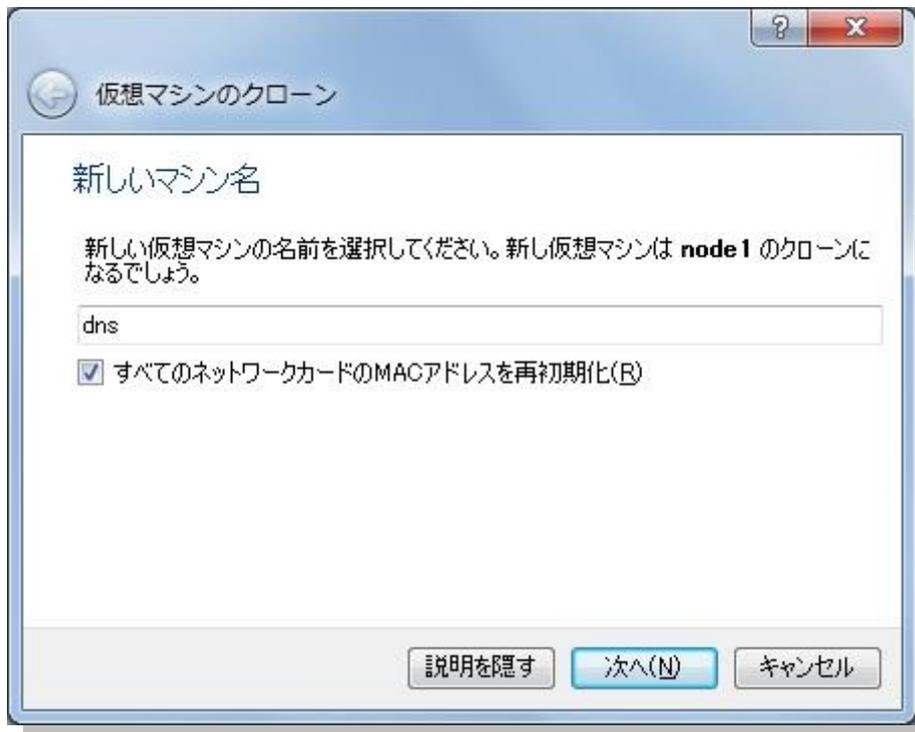
既存の DNS サーバーがあるといったその他の場合については、「Appendix 1. DNS サーバーの準備」を参照してください。

### 1. 仮想マシンのクローン

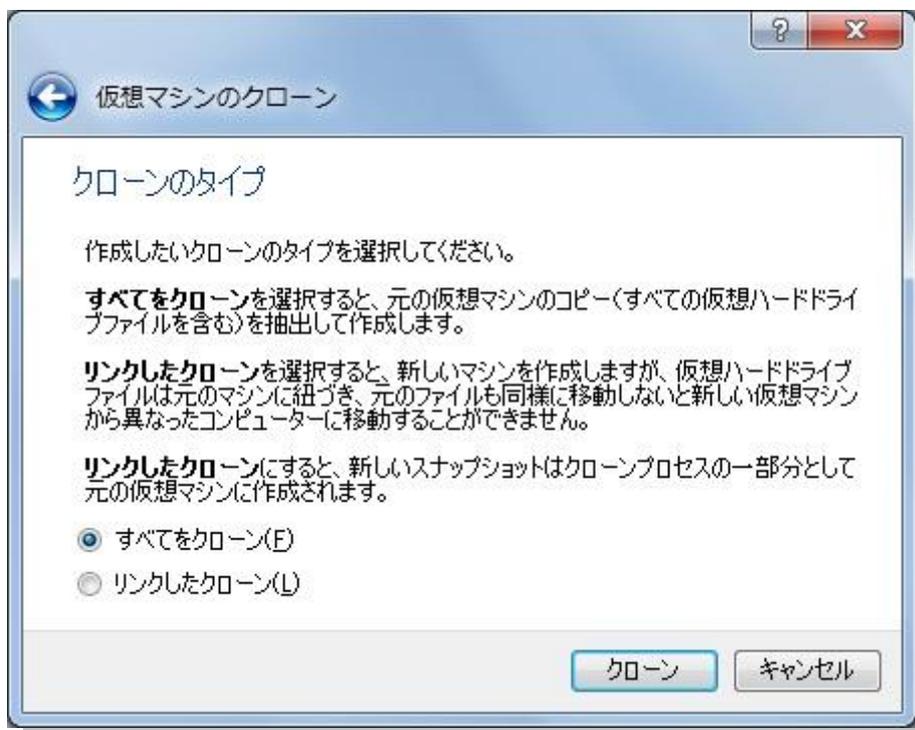
「node1」を選択した状態で右クリックでメニューを表示し、「クローン」を選択します。または、「仮想マシン」メニューから「クローン」を選択します。



作成する仮想マシンの名前として、ここでは「dns」と入力します。また、「すべてのネットワークカードのMACアドレスを再初期化」にチェック (☑) をつけます。入力後、「次へ」をクリックします。



クローンのタイプとして「すべてをクローン」を選択して「クローン」をクリックして実行します。

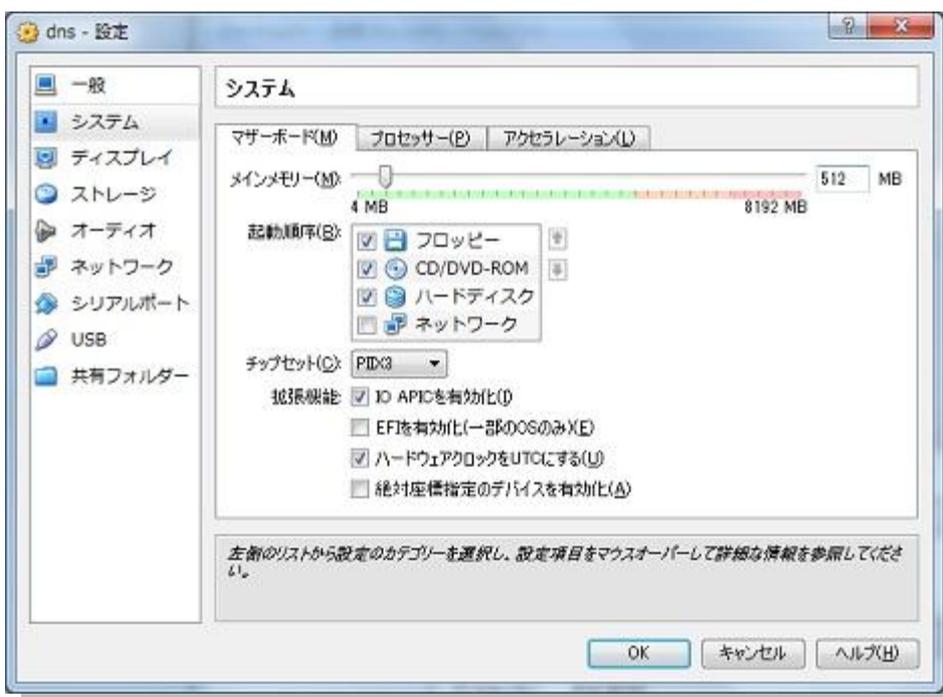


## 2. メモリの変更

クローンの完了後、メモリ設定の変更を実施します。仮想マシン「dns」を選択した後、「システム」をクリックして設定画面を表示します。



設定画面では、「システム」セクションのメインメモリーを「512」MBに設定して「OK」をクリックします。



続いて、仮想マシンを起動して設定を続けます。「dns」を選択して、「起動」をクリックします。



### 3. クローン後の設定変更

dns の起動後、まずホスト名を変更します。root ユーザーで次のコマンドを実行して設定ファイル内の記述を変更します。

```
# vi /etc/sysconfig/network
```

<編集例> ※ HOSTNAME の設定に関する一行で、node1 という部分を dns に変更します。

```
[root@node1 ~]# vi /etc/sysconfig/network
```

< 次の一行を変更する >

```
HOSTNAME=node1.oracle12c.jp
```

< 変更後 >

```
HOSTNAME=dns.oracle12c.jp
```

続いて、ネットワークを設定します。「システム」メニューの「設定」の中から「ネットワーク接続」を選択します。

「eth0」を選択して「削除」をクリックします。



確認メッセージが表示されたら「削除」をクリックします。



続いて「Auto eth1」を選択して「編集」をクリックします。



接続名に「eth0」、方式を「手動」に設定して「追加」からアドレスとネットマスクを入力します。



ここではアドレスに「192.168.56.254」、ネットマスクに「255.255.255.0」と設定するものとします。設定後、「適用」をクリックします。

ネットワークの設定変更は以上です。「閉じる」をクリックします。



ネットワーク・インターフェースの情報を初期化するために設定ファイルを削除します。root ユーザーで次のコマンドを実行します。

```
# cd /etc/udev/rules.d
```

```
# ls 70-persistent-net.rules
```

```
# rm -rf 70-persistent-net.rules
```

<実行例>

```
[root@node1 ~]# cd /etc/udev/rules.d
[root@node1 rules.d]# ls 70-persistent-net.rules
70-persistent-net.rules
[root@node1 rules.d]# rm -rf 70-persistent-net.rules
```

設定を反映するために、一旦再起動します。root ユーザーで次のコマンドを実行します。

```
# shutdown -r now
```

#### 4. DNS の起動

再起動後、DNS で名前解決を行うための情報を設定します。今回は DNS サーバーに dnsmasq を使用します。root ユーザーで hosts ファイルに次の記載を追加します。

```
# vi /etc/hosts
```

<記入例>

```
[root@dns ~]# vi /etc/hosts

< 次の記述を追加>
192.168.56.101      node1.oracle12c.jp  node1
192.168.56.102      node2.oracle12c.jp  node2
192.168.56.111      node1-vip.oracle12c.jp  node1-vip
192.168.56.112      node2-vip.oracle12c.jp  node2-vip
192.168.56.201      scan.oracle12c.jp    scan
192.168.56.202      scan.oracle12c.jp    scan
192.168.56.203      scan.oracle12c.jp    scan
192.168.56.254      dns.oracle12c.jp     dns
```

DNS のサービスを起動します。root ユーザーで次のコマンドを実行します。

```
# chkconfig dnsmasq --list
```

```
# chkconfig dnsmasq on
```

```
# chkconfig dnsmasq --list
```

```
# service dnsmasq start
```

<実行例>

```
[root@dns ~]# chkconfig dnsmasq --list
dnsmasq    0:off  1:off  2:off  3:off  4:off  5:off  6:off
[root@dns ~]# chkconfig dnsmasq on
[root@dns ~]# chkconfig dnsmasq --list
dnsmasq    0:off  1:off  2:on   3:on   4:on   5:on   6:off
[root@dns ~]# service dnsmasq start
Starting dnsmasq:          [ OK ]
```

DNS サーバー用の仮想マシンの設定は以上です。

## 5. インストール前の事前準備

本ガイドの構成での Oracle Database のインストール前に実施すべき、インストール前の事前設定について以下の順で説明します。

- 5.1 仮想マシンの準備
- 5.2 oracle-validated-verify の実行
- 5.3 OS グループ、ユーザー、およびディレクトリの作成
- 5.4 ハードウェア要件とメモリの確認
- 5.5 ネットワーク要件の確認
- 5.6 ソフトウェア要件の確認
- 5.7 環境変数とリソース制限の設定

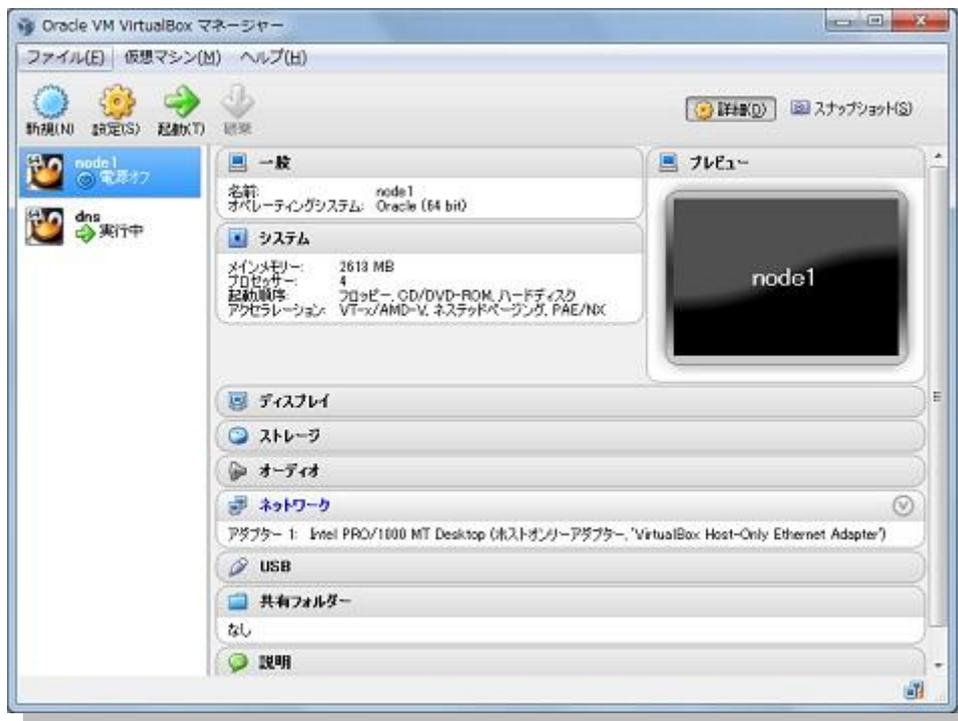
本文書では、Oracle Linux 6 Update 4 のインストール時に Oracle Validated RPM パッケージをインストールしています。Oracle Validated RPM は Oracle Database のインストールに必要な構成タスクを実施するものですが、ここで紹介しているインストール前の事前設定を完全に補うものではありませんのでご注意ください。つまり Oracle Validated RPM を使用した場合も、インストール前の事前設定について確認を行い、適宜設定を実施するようにします。

### 5.1 仮想マシンの準備

これからインストール前の事前準備をする仮想マシンを用意します。ここでは node1 に設定を行うものとします。node1 を起動する前に、ネットワークの設定を変更します。

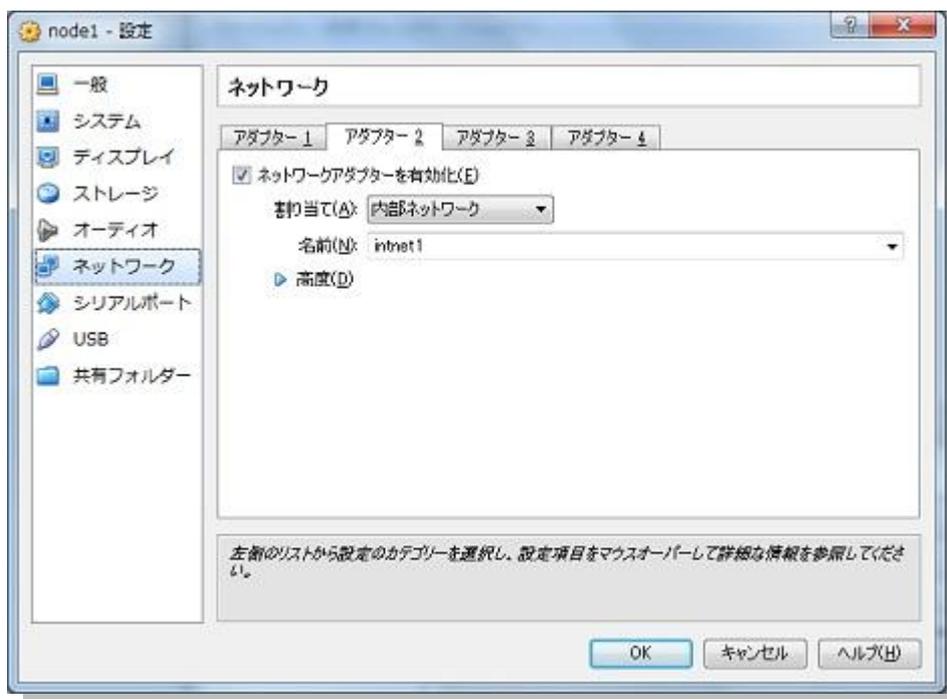
## 1. ネットワークの追加設定

仮想マシン「**node1**」を選択した後、「**ネットワーク**」をクリックして設定画面を表示します。



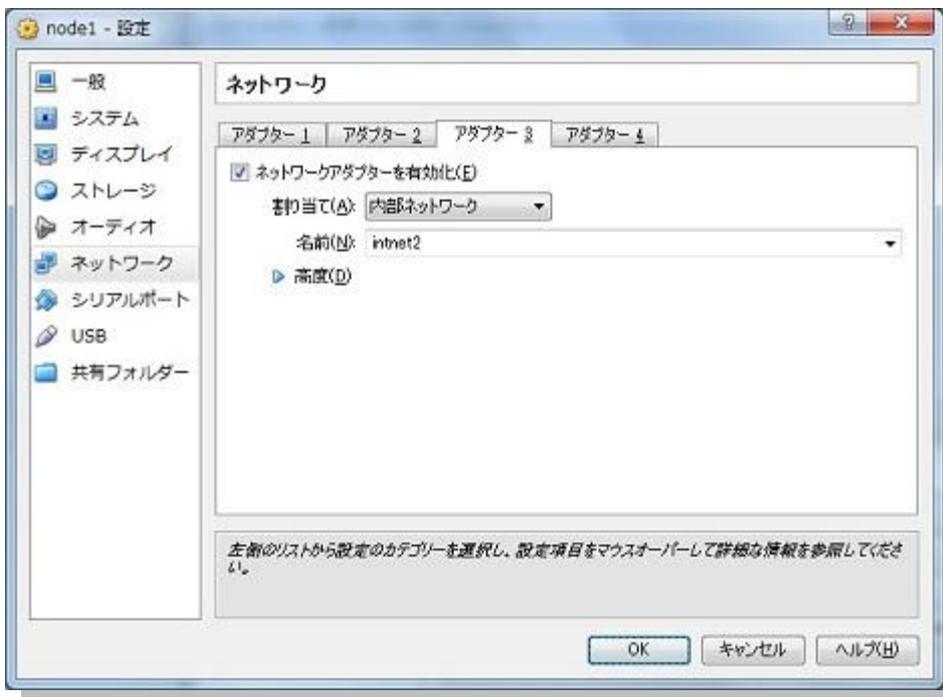
ネットワークの設定として、**アダプター2**と**アダプター3**の設定を変更します。

ネットワーク設定の画面で「**アダプター 2**」タブを選択し、「**ネットワークアダプターを有効化**」にチェック ()します。続いて、割り当てを「**内部ネットワーク**」に変更して、名前には「**intnet1**」を入力します。

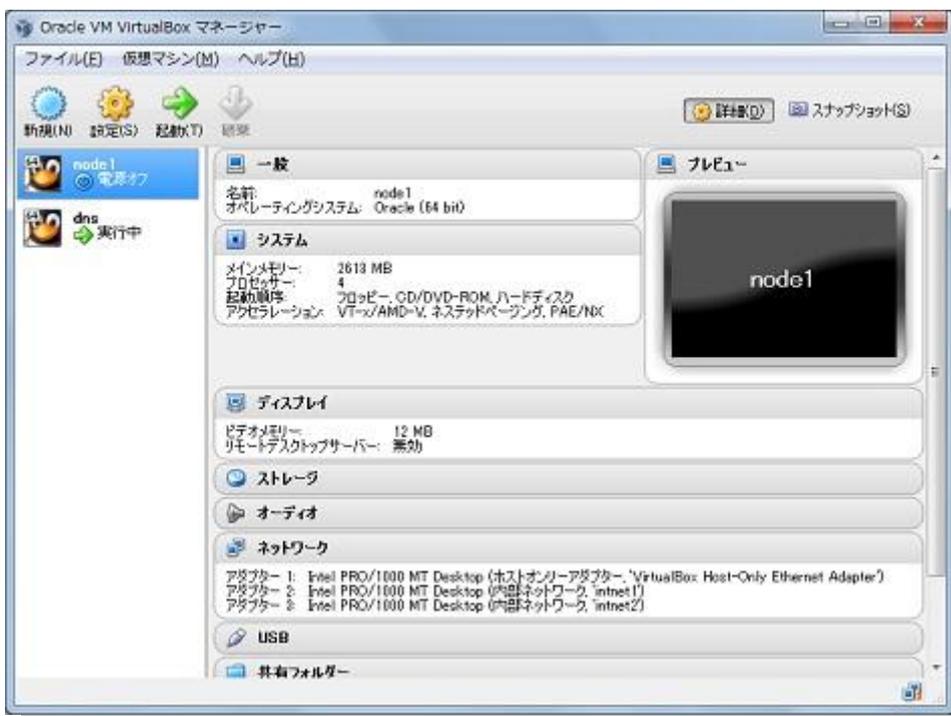


「アダプター 3」タブを選択し、「ネットワークアダプターを有効化」にチェック (☑)します。続いて、割り当てを「内部ネットワーク」に変更して、名前には「intnet2」を入力します。

設定後、「OK」をクリックします。



「ネットワーク」セクションに3つのアダプターが表示されたことを確認して完了です。続いて node1 の設定のため、「node1」を選択して「起動」をクリックします。



## 2. 仮想マシンの設定

node1 の起動後、root ユーザーでログインします。

ログイン後、「システム」メニューの「設定」の中から「ネットワーク接続」を選択します。設定画面が表示されたら「Auto eth1」を選択して「編集」をクリックします。



接続名を「eth1」に変更して、「自動接続する」にチェック (☑) がされていることを確認します。続いて「IPv4 のセッティング」タブより、方式を「手動」にして「追加」からアドレスとネットマスクを入力します。

ここではアドレスに「192.168.100.101」、ネットマスクに「255.255.255.0」と設定するものとします。設定後、「適用」をクリックします。



続いて「Auto eth2」を選択して「編集」をクリックします。



接続名を「eth2」に変更して、「自動接続する」にチェック (☑) がされていることを確認します。続いて「IPv4 のセッティング」タブより、方式を「手動」にして「追加」からアドレスとネットマスクを入力します。

ここではアドレスに「192.168.200.101」、ネットマスクに「255.255.255.0」と設定するものとします。設定後、「適用」をクリックします。



ネットワークの設定変更は以上です。「閉じる」をクリックします



## 5.2 oracle-validated-verify の実行

本文書の構成では、Oracle Validated RPM パッケージはインストールされているものの、一部設定値の変更などが適用されていません。Oracle Linux 6 Update 4 のインストールを日本語環境にて実施した場合には、root ユーザーで以下のコマンドを実行して、英語環境で Oracle Validated RPM による設定を実施します。

```
# export LANG=C
```

```
# oracle-rdbms-server-11gR2-preinstall-verify
```

<実行例>

```
# export LANG=C
# oracle-rdbms-server-11gR2-preinstall-verify
```

12c 用の Oracle Pre-Install RPM パッケージをインストールした場合は、上記コマンドの代わりに oracle-rdbms-server-12cR1-reinstall-verify コマンドを実行します。

## 5.3 OS グループ、OS ユーザー、およびディレクトリの作成

続いて、インストールに必要な OS グループ、OS ユーザー、およびディレクトリを作成します。

次の図は、今回の構成について概要を示します。

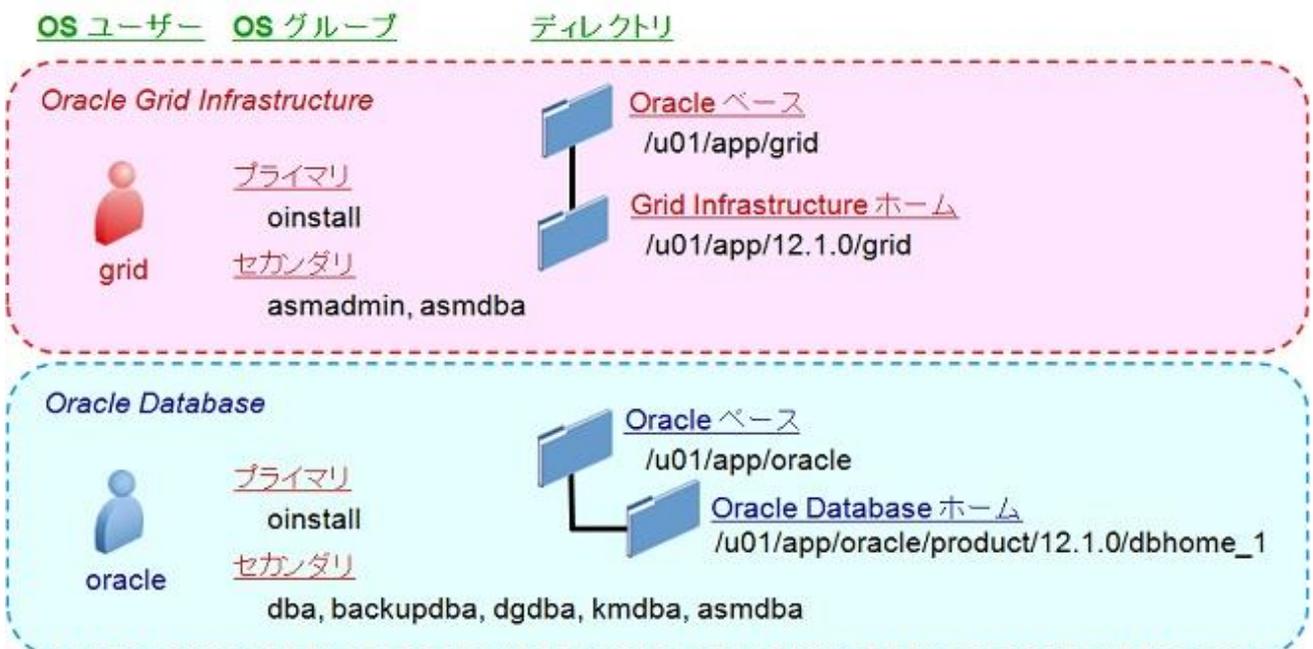


図 4 : OS ユーザー、OS グループ、ディレクトリ構成の概要

Oracle Validated RPM により oracle ユーザーと必要最小限の OS グループとして、ここでは oinstall と dba が作成されています。今回のように oracle ユーザー以外の OS ユーザーを使用してインストールを行う場合や任意に作成した OS グループを使用して Database インスタンスや ASM インスタンスに対して高度な管理を行う場合には、oinstall および dba 以外の OS グループも使用するため、ここで以下のコマンドを実行することにより追加で作成しておきます。

以下のコマンドを root ユーザーで実行します。

```
# groupadd -g 1101 oper
# groupadd -g 1102 backupdba
# groupadd -g 1103 dgdba
# groupadd -g 1104 kmdba
# groupadd -g 1200 asmadmin
# groupadd -g 1201 asmdba
# groupadd -g 1202 asmoper
```

<実行例>

```
# groupadd -g 1101 oper
# groupadd -g 1102 backupdba
# groupadd -g 1103 dgdba
# groupadd -g 1104 kmdba
# groupadd -g 1200 asmadmin
# groupadd -g 1201 asmdba
# groupadd -g 1202 asmoper
```

続いてOSユーザーを作成します。ここでは、gridユーザーを新規に作成します。oracleユーザーは、すでに作成されているため、oracleユーザーについてはOSグループの設定変更を実施するものとします。(oracleユーザーに対して設定されている初期パスワードは oracle です。)

以下のコマンドを root ユーザーで実行します。

```
# useradd -u 1100 -g oinstall -G asmadmin,asmdba,asmoper grid
# usermod -u 54321 -g oinstall -G dba,backupdba,dgdba,kmdba,asmdba oracle
# passwd grid
# passwd oracle
```

<実行例>

```
# useradd -u 1100 -g oinstall -G asmadmin,asmdba,asmoper grid
# usermod -u 54321 -g oinstall -G dba,backupdba,dgdba,kmdba,asmdba oracle
# passwd grid
Changing password for user grid.
New UNIX password:
Retype new UNIX password:
passwd: all authentication tokens updated successfully.
# passwd oracle
Changing password for user oracle.
New UNIX password:
Retype new UNIX password:
passwd: all authentication tokens updated successfully.
```

作成後は、以下のコマンドでユーザーの設定を確認することができます。

```
# id grid
```

```
# id oracle
```

<実行例>

```
# id grid
uid=1100(grid) gid=54321(oinstall) 所属グループ
=54321(oinstall),1200(asmadmin),1201(asmdba),1202(asmoper)
# id oracle
uid=54321(oracle) gid=54321(oinstall) 所属グループ
=54321(oinstall),54322(dba),1201(asmdba),1102(backupdba),1103(dgdba)
,1104(kmdba)
```

最後に、以下のコマンドを root ユーザーで実行してインストールに必要なディレクトリを作成します。

```
# mkdir -p /u01/app/grid
```

```
# mkdir -p /u01/app/oracle
```

```
# mkdir -p /u01/app/12.1.0/grid
```

```
# chown -R grid:oinstall /u01
```

```
# chown -R oracle:oinstall /u01/app/oracle
```

```
# chmod -R 775 /u01
```

<実行例>

```
# mkdir -p /u01/app/grid
# mkdir -p /u01/app/oracle
# mkdir -p /u01/app/12.1.0/grid
# chown -R grid:oinstall /u01
# chown -R oracle:oinstall /u01/app/oracle
# chmod -R 775 /u01
```

## 5.4 ハードウェア要件とメモリの確認

ここでは、ハードウェアに関する要件とメモリを確認します。

- システムのアーキテクチャ

以下のコマンドを実行してシステムのアーキテクチャを確認することができます。

```
# uname -m
```

<実行例>

```
# uname -m
x86_64
```

- システムの実行レベル

以下のコマンドを root ユーザーで実行して、システムが実行レベル 3 か 5 で起動していることを確認します。

```
# runlevel
```

<実行例>

```
# runlevel
N 5
```

- OS バイナリ

今回の環境では実施する必要はありませんが、その他の要件としては構成に使用するすべてのノードで、同一の OS バイナリが使用されている必要があります。

- ディスプレイ解像度

また、Oracle Universal Installer (OUI) の起動に必要なディスプレイ解像度として、最低 1024 x 768 を満たしている必要があります。

- 物理メモリ

Linux x86\_64 の環境における物理メモリの要件は 4GB です。次のコマンドで確認します。(今回は手順の確認を目的としているため 2.5GB で構築するものとします。可能であれば 4GB に設定してください。)

```
# grep MemTotal /proc/meminfo
```

- スワップ領域

スワップ領域の最低要件は、システムのアーキテクチャと物理メモリの容量によって異なりますので以下を参考してください。以下は、Linux x86\_64 環境における要件です。

使用可能な物理の容量	スワップ領域として必要な容量
4GB から 16GB	物理メモリと同量
16GB 以上	16GB

システムのスワップ領域は、以下のコマンドを実行して確認します。

```
# grep SwapTotal /proc/meminfo
```

スワップ領域の拡張が必要な場合には、OS のドキュメントなどを参照してスワップ領域の拡張手順を確認した上で実行します。

以下のコマンドで、前述の物理メモリとあわせてスワップ領域の空き容量を確認することができます。

```
# free
```

- 一時領域

一時領域として、/tmp に最低 1GB (1024MB) の空き領域があることも確認しておきます。

```
# df -h /tmp
```

- ディスクの空き容量

また Linux x86\_64 環境では、ソフトウェアやデータファイルの配置用として以下の空き容量が必要です。

- Oracle Grid Infrastructure のベース・ディレクトリ：最低 3.5 GB
- Oracle Grid Infrastructure のホーム・ディレクトリ：最低 8 GB
  - 上記の値は、ソフトウェアをインストールするために必要な容量に加えて、ログの領域や Oracle Cluster Health Monitor のリポジトリを含みます。
- Oracle Database のベース・ディレクトリ：5.8 GB
  - 上記の値は、自動バックアップを構成する場合には、別途、高速リカバリ領域用として追加で領域を用意する必要があります。

- /dev/shm ファイルシステム

自動メモリ管理 (MEMORY\_TARGET 初期化パラメータ、あるいは MEMORY\_MAX\_TARGET 初期化パラメータ) を使用する場合には、その値より大きなサイズで /dev/shm がマウントされている必要があります。自動メモリ管理を使用せずに、SGA\_TARGET 初期化パラメータ、および PGA\_AGGREGATE\_TARGET 初期化パラメータを使用する場合には、/dev/shm の確保は特に必要ありません。

以下のコマンドで、現在の値を確認します。ここでは、実行例にあるように領域が確保されているので、確認のみ実施し、明示的な設定変更などは必要ないものとします。

```
# df -k
```

<実行例>

```
# df -k
Filesystem          1K-ブロック   使用   使用可  使用%  マウント位置
/dev/mapper/VolGroup00-LogVol100
                    20726940    3494812 16162256  18% /
/dev/sda1           101086      23318   72549    25% /boot
tmpfs               1309284      0       1309284  0% /dev/shm
```

もし、/dev/shm がマウントされていない場合には、以下のコマンドを root ユーザーで実行してマウント・ポイントを作成します。以下は、1500MB で作成する際の例です。

```
# mount -t tmpfs tmpfs -o size=1500m /dev/shm
```

システムの再起動後にも自動でマウントされるようにするためには、/etc/fstab ファイルに以下のように追記します。

<追記例>

```
# vi /etc/fstab

<以下の内容を追記>
tmpfs          /dev/shm      tmpfs          size=1500m    0 0
```

## 5.5 ネットワーク要件の確認

次に、ネットワークの要件を確認します。

- ネットワーク・インターフェース・カード (NIC)

クラスタを構成する各ノードに最低 2 つの NIC が必要です。1 つをパブリック・ネットワーク用として、もう 1 つをプライベート (インターコネクト) ・ネットワーク用として使用します。本ガイドでは、インターコネクトの冗長化を行うため、プライベート・ネットワーク用として 2 つの仮想 NIC を用意しています。

各ネットワークの要件については次を参考にしてください。

- パブリック・ネットワーク
  - TCP / IP をサポートしていること
- プライベート・ネットワーク (インターコネクト・ネットワーク)
  - TCP / IP をサポートしていること
  - クロスオーバー・ケーブルは非サポート
  - PSR 11.2.0.2 以降を使用する場合には、インターコネクトの冗長化機能を利用して複数のネット

## ワーク・インターフェースをインターコネクトとして構成可能

## ● IP アドレス

クラスタを構成する各ノードに、次の IP アドレスが必要です。

- パブリック IP アドレス
- 仮想 IP (VIP) アドレス
- プライベート IP アドレス
- Single Client Access Name (SCAN) IP アドレス
  - データベースへのアクセスに使用される単一のエイリアス
  - 単一の SCAN 名 (ホスト名) に対して、3 つの IP アドレスを名前解決できるように設定

IP アドレスの構成には、以下 2 つのオプションがあります。

1. DNS サーバーを用いた静的な IP アドレスの割り当て
2. Grid Naming Service (GNS) と DHCP サーバーを用いた構成による動的な IP アドレスの割り当て

ここでは、DNS サーバーを使用して必要な IP アドレスの構成を実施します。

## ● Oracle Flex ASM 利用時のネットワーク

ストレージ構成に Oracle Flex ASM 構成を利用する場合には、ASM で利用するネットワークが必要です。このネットワークは、インターコネクトとして使用するプライベート・ネットワークと別でも構いませんし、同一のネットワークを利用することもできます。

今回は、インターコネクトとして使用するプライベート・ネットワークと同一のネットワークを利用するものとします。

## ● ブロードキャスト通信 (ADP および UDP)

PSR 11.2.0.2 以上のバージョンを使用する場合には、すべてのノードにおいてパブリックおよびプライベート・ネットワーク用のインターフェースでブロードキャスト通信を行えるようにします。

## ● マルチキャスト通信

クラスタを構成するすべてのノードにおいて、プライベート・ネットワークのマルチキャスト通信を行えるように設定します。

## ● 複数プライベート・ネットワーク利用時の設定

RAC 構成において複数のネットワークをプライベート・ネットワークとして利用する場合には、rp\_filter パラメータの設定が必要です。このカーネル・パラメータの設定を 0 (フィルタなし) または 2 (緩いフィルタ) に設定します。

この設定は、使用している Oracle Linux カーネルのバージョンによっては既に 2 に設定されている場合が

あります。今回の構成では、設定の変更が必要であるため手順を実施します。

## 1. カーネル・パラメータの設定変更

root ユーザーで /etc/sysctl.conf ファイルの最後に次の設定を追記します。

```
# vi /etc/sysctl.conf
```

<記述例>

```
# vi /etc/sysctl.conf  
  
< 以下の記載を追記 >  
net.ipv4.conf.eth1.rp_filter = 2  
net.ipv4.conf.eth2.rp_filter = 2
```

追記した内容を保存して、次のコマンドを実行して変更した値を反映します。

```
# /sbin/sysctl -p
```

## 2. 名前解決のための記述を追記します。

/etc 配下の hosts ファイルに次の設定を記述します。

```
# cd /etc
```

```
# vi hosts
```

<実行例>

```
# cd /etc  
# vi hosts  
  
<以下を追記>  
192.168.56.101      node1.oracle12c.jp  node1  
192.168.56.102      node2.oracle12c.jp  node2  
192.168.56.111      node1-vip.oracle12c.jp  node1-vip  
192.168.56.112      node2-vip.oracle12c.jp  node2-vip
```

## 5.6 ソフトウェア要件の確認

続いて、ソフトウェアの要件を確認します。今回は Oracle Validated RPM パッケージで設定を行っているため、特に設定は必要ありませんが、次の項目について製品マニュアルを参照の上、最新の要件を満たしているかを確認する必要があります。

### ● RPM パッケージ

Oracle Database のインストールに必要なパッケージを確認します。必要なパッケージは、使用する OS のバージョンによって異なります。

追加インストールやインストール済みのパッケージの確認が必要な場合には、root ユーザーで rpm コマンドを使用します。

### ● カーネル・パラメータ

続いて、カーネル・パラメータの設定を確認します。推奨値は、使用する OS のバージョンによって異なります。設定値は、次のコマンドを root ユーザーで実行して確認します。

```
# sysctl -a
```

設定変更が必要な場合には、root ユーザーで /etc/sysctl.conf ファイルを編集の上、設定変更を反映するために次のコマンドを実行します。

```
# sysctl -p
```

## 5.7 システム時刻同期の設定

システム時刻は、クラスタを構成するすべてのノード間で同期されている必要があります。クラスタ内のシステム時刻の同期に使用できる方法は、次の 2 つです。

- Cluster Time Synchronization Service (CTSS)
- Network Time Protocol (NTP)

CTSS はクラスタ時刻同期化サービスとも呼ばれ、Oracle Database 11g Release 2 より提供されているシステム時刻の同期を行う新機能です。NTP が設定されていない場合は、CTSS によりクラスタ内のすべてのノードでシステム時刻が同期されます。ただし、CTSS は Oracle Clusterware により提供されているサービスであるため、システム時刻の同期はあくまでもクラスタ内のノードが対象です。クラスタ外のノードともシステム時刻を同期する場合には NTP を使用します。

本ガイドでは、システム時刻の同期に CTSS を使用します。システム時刻の同期に CTSS を使用する場合と NTP を使用する場合の設定方法を以下に記載します。

### CTSS を使用する場合

CTSS を使用する場合には、NTP を停止して構成を削除します。手順は root ユーザーで実施します。

1. ntpd サービスの起動状態を確認して、稼働している場合には停止します。

```
# service ntpd status
```

```
# service ntpd stop
```

<実行例>

```
# service ntpd status
ntpd は停止しています

<稼働している場合には次のコマンドで停止>
# service ntpd stop
ntpd を停止中: [ OK ]
```

2. ntpd サービスの設定を確認して、有効化されている場合は無効化します。

```
# chkconfig --list ntpd
```

```
# chkconfig ntpd off
```

<実行例>

```
# chkconfig --list ntpd
ntpd          0:off  1:off  2:off  3:off  4:off  5:off  6:off

<有効化 (on) に設定されている場合には次のコマンドで無効化>
# chkconfig ntpd off
```

3. 設定ファイルがある場合には、削除あるいは別名保存します。

```
# rm /etc/ntp.conf (または # mv /etc/ntp.conf /etc/ntp.conf.org など)
```

<実行例>

```
# rm /etc/ntp.conf
rm: remove 通常ファイル `/etc/ntp.conf'? y
```

4. ntpd.pid ファイルが存在する場合にはファイルを削除します。

```
# ls -l /var/run/ntpd.pid
```

<実行例>

```
# ls -l /var/run/ntpd.pid
ls: /var/run/ntpd.pid: そのようなファイルやディレクトリはありません

<存在する場合には次のコマンドで削除>
# rm /var/run/ntpd.pid
```

## NTP を使用する場合

NTP を使用する場合には、-x オプションを指定して時刻調整の方法を slew モードに設定します。slew モード

では、システム時刻を同期する際に、時刻の後戻りが発生することがありません。手順は root ユーザーで実施します。

1. /etc/sysconfig/ntpd ファイルに、-x オプションの指定を追加します。

```
# vi /etc/sysconfig/ntpd
```

<実行例>

```
# vi /etc/sysconfig/ntpd
# Drop root to id 'ntp:ntp' by default.
OPTIONS="-x -u ntp:ntp -p /var/run/ntpd.pid"
# Set to 'yes' to sync hw clock after successful ntpdate
SYNC_HWCLOCK=no
# Additional options for ntpdate
NTPDATE_OPTIONS=""
```

2. ntpd サービスが無効化されている場合は有効化します。

```
# chkconfig --list ntpd
```

```
# chkconfig ntpd on
```

<実行例>

```
# chkconfig --list ntpd
ntpd          0:off  1:off  2:off  3:off  4:off  5:off  6:off
<無効化 (off) に設定されている場合には次のコマンドで有効化>
# chkconfig ntpd on
```

3. ntpd サービスを起動します。すでに起動している場合には再起動します。

```
# service ntpd start (再起動の場合は # service ntpd restart )
```

<実行例>

```
# service ntpd start
ntpd を起動中: [ OK ]
```

4. ntpd サービスが-x オプション付きで起動していることを確認します。

```
# ps -ef |grep ntpd
```

<実行例>

```
# ps -ef |grep ntpd
ntp  4476  1  0 12:00 ? 00:00:00 ntpd -x -u ntp:ntp -p
/var/run/ntpd.pid -g
```

## 5.8 cvuqdisk パッケージのインストール

Cluster Verification Utility (CVU) で使用する cvuqdisk パッケージをインストールします。

CVU は、クラスタ検証ユーティリティとも呼ばれ、クラスタ環境や RAC を構成する各コンポーネントの構成や稼働状態を確認することができるユーティリティです。インストールおよび構成前後の検証に加え、RAC 構築時や運用、管理作業におけるトラブルシューティングにも利用できます。Oracle Database 11g Release 2 からは、一部の検証項目について適切な設定への修正を可能にする修正スクリプトを生成する機能も実装されています。

cvuqdisk パッケージを各ノードにインストールするためには、次の 2 つの方法があります。

- rpm コマンドを使用した手動でのインストール (cvuqdisk パッケージはソフトウェアに同梱)
- CVU での検証により生成された修正スクリプトを使用したインストール

ここでは、OUI でのインストール時に CVU による検証で生成される修正スクリプトを使用して cvuqdisk パッケージのインストールを行うものとします。

## 5.9 Intelligent Platform Management Interface (IPMI) の設定

Intelligent Platform Management Interface (IPMI) を利用して、障害ノードを停止することができます。IPMI を利用できるハードウェアを用いた環境の場合には、IPMI の構成した上で Oracle Clusterware に IPMI の使用設定を行うことで、障害ノードに対して Oracle Clusterware や OS に依存せずに Baseboard Management Controller (BMC) を利用した LAN 経由でのノード停止を可能にします。

本ガイドでは、IPMI の設定および Oracle Clusterware への使用設定はしないものとします。

## 5.10 Secure Shell (SSH) の構成

クラスタを構成しているすべてのノード間では、パズフレーズなしでの Secure Shell (SSH) 接続が確立されている必要があります。SSH 接続の確立には、次の 2 つの方法があります。

- インストール前にノードごとに手動での構成を実施する
- インストール中にすべてのノードで OUI を使用した構成を実施する

本ガイドでは、OUI を使用してインストール中に SSH 接続を確立する方法をとるものとします。

## 5.11 環境変数とリソース制限の設定

環境に応じて、ソフトウェアをインストールする OS ユーザー (今回は oracle) に環境変数とリソース制限を設定します。

OUI を日本語で表示したい場合には、インストールを実施するユーザーの環境変数 LANG を確認し、LANG=ja\_JP.UTF-8 に設定して OUI を起動します。

次に、リソース制限を設定します。リソース制限は、インストールに使用する OS ユーザーに対して設定します。設定には各ノードの /etc/security 配下にある limits.conf ファイルを使用します。

本ガイドでは Oracle Validated RPM パッケージを使用することにより oracle ユーザーに対する一部の設定は完了しているため、追加で作成した grid ユーザーについて設定を行います。

root ユーザーで /etc/security/limits.conf ファイルを編集して、ファイルの記述の最後に次の記述を追記します。

```
# vi /etc/security/limits.conf
```

<記述例>

```
# vi /etc/security/limits.conf

<以下を追記>
grid soft nproc 2047
grid hard nproc 16384
grid soft nofile 1024
grid hard nofile 65536
grid soft stack 10240
grid hard stack 32768
```

追記後、設定を保存します。

## 5.12 記憶域の確認と準備

Oracle Database 12c Release 1 では、OUI や Database Configuration Assistant (DBCA) といった GUI のツールで次の記憶域が使用できます。

- Oracle Automatic Storage Management (ASM)
- 共有ファイルシステム

RAW デバイスあるいはブロック・デバイス上に Oracle Cluster Registry (OCR) や投票ディスクを含む Oracle Clusterware 関連のファイルや Oracle Database 関連のファイルを格納することはできません。

本ガイドでは Oracle ASM を記憶域として選択するものとします。各ノードで共有されるように設定された仮想ディスク・デバイスごとに単一パーティションを作成して、それぞれを 1 本の ASM ディスクとして使用します。

設定を行うために、一旦仮想マシンを停止して Oracle VM VirtualBox マネージャーから操作を行います。

root ユーザーで次のコマンドを実行し、仮想マシンを停止します。

```
# shutdown -h now
```

仮想マシンの停止後、複数のゲスト OS (仮想マシン) から利用可能な共有ストレージの設定を開始します。ここでは共有ストレージ用の領域 (フォルダ) として C:\%VBox%\shared\_disk を利用するものとし、5GB の仮想ディスクを 6 本作成するものとします。

各仮想ディスクの名称と、ゲスト OS (仮想マシン) のデバイス名の対応は以下に記載します。

<仮想ディスク一覧>

ホスト OS 上における仮想ディスク名	ゲスト OS 上 (/dev 配下) におけるデバイス名
disk01.vdi	sdb
disk02.vdi	sdc
disk03.vdi	sdd
disk04.vdi	sde
disk05.vdi	sdf
disk06.vdi	sdg

## 1. フォルダの作成

まず、ホスト OS 上でコマンドプロンプトを用意し、次のコマンドを実行してフォルダを作成します。

コマンドプロンプトは Windows の「スタート」メニューから「ファイル名を指定して実行」を選択し、「cmd」を入力して「OK」をクリックします。あるいは、「スタート」メニューから「すべてのプログラム」より「アクセサリ」を選択し、「コマンドプロンプト」を選択します。

コマンドプロンプトが表示されたら、次のコマンドを実行します。

```
mkdir C:¥VBox¥shared_disk
```

<実行例>

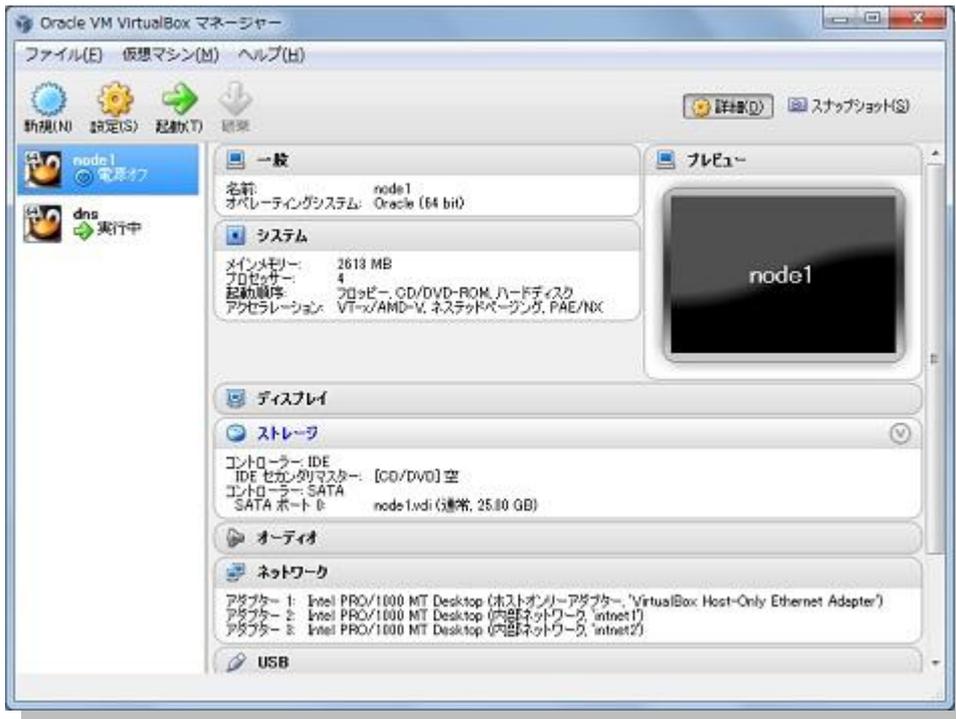
```
C:¥Users¥candy> mkdir C:¥VBox¥shared_disk
```

フォルダーの作成は、コマンドプロンプトではなく Windows エクスプローラーを使用しても構いません。

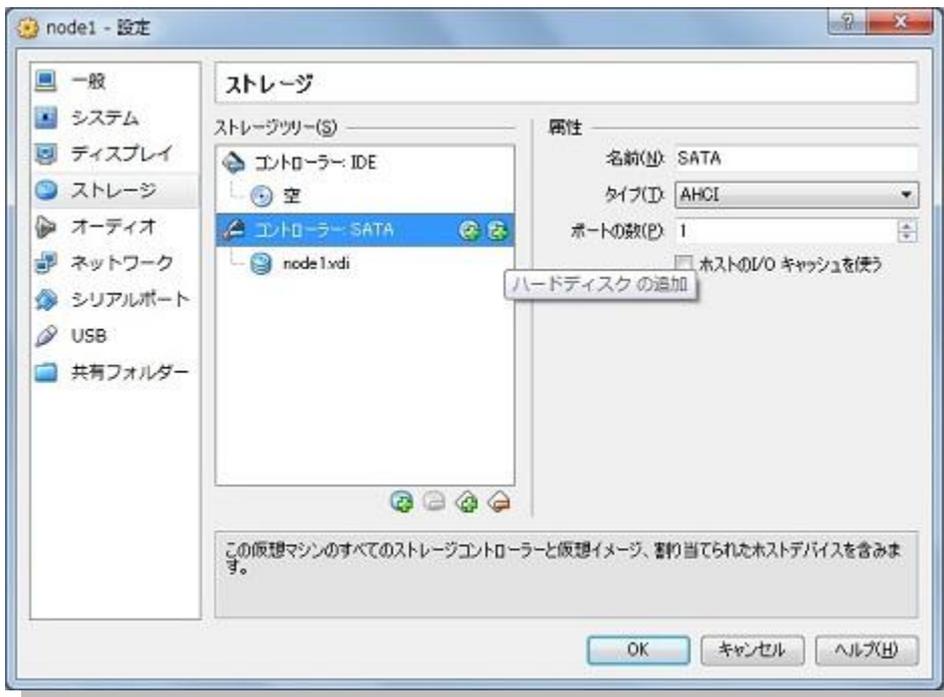
## 2. 仮想ディスクの作成

続いて、仮想ディスクを作成します。Oracle VM VirtualBox マネージャーより操作を実行します。

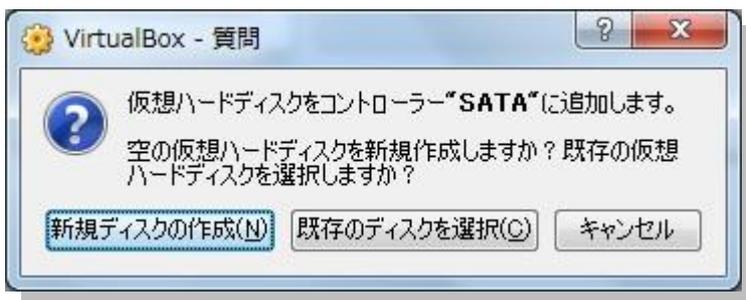
仮想マシン「node1」を選択した後、「ストレージ」をクリックして設定画面を表示します。



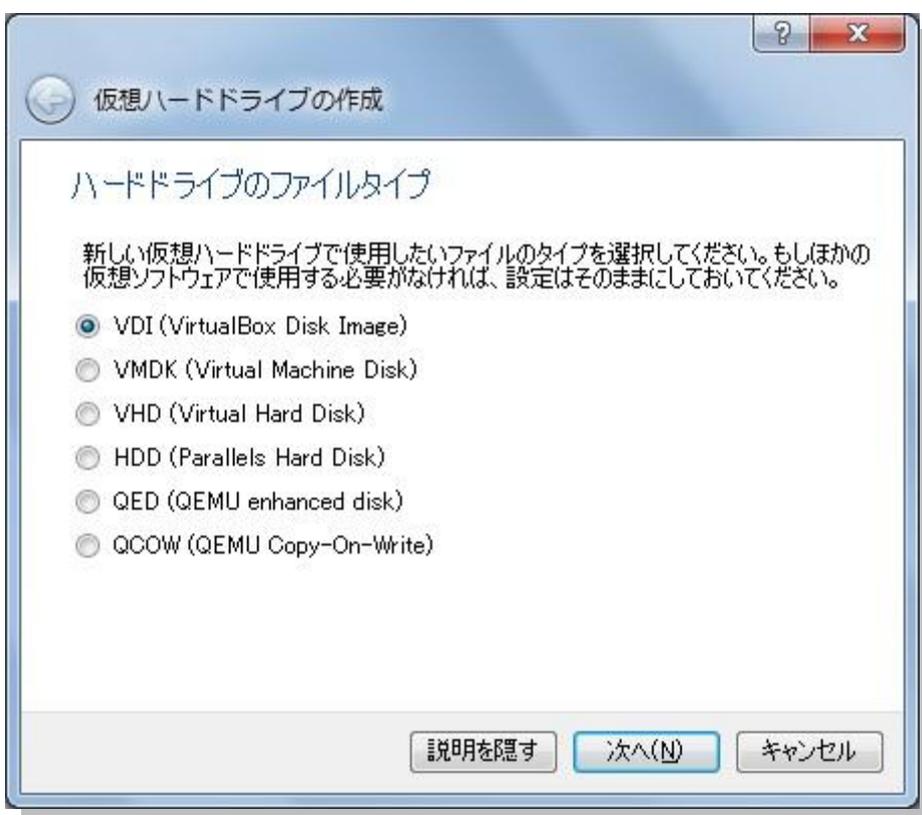
設定画面より、コントローラー : SATA の「ハードディスクの追加」のアイコンをクリックします。



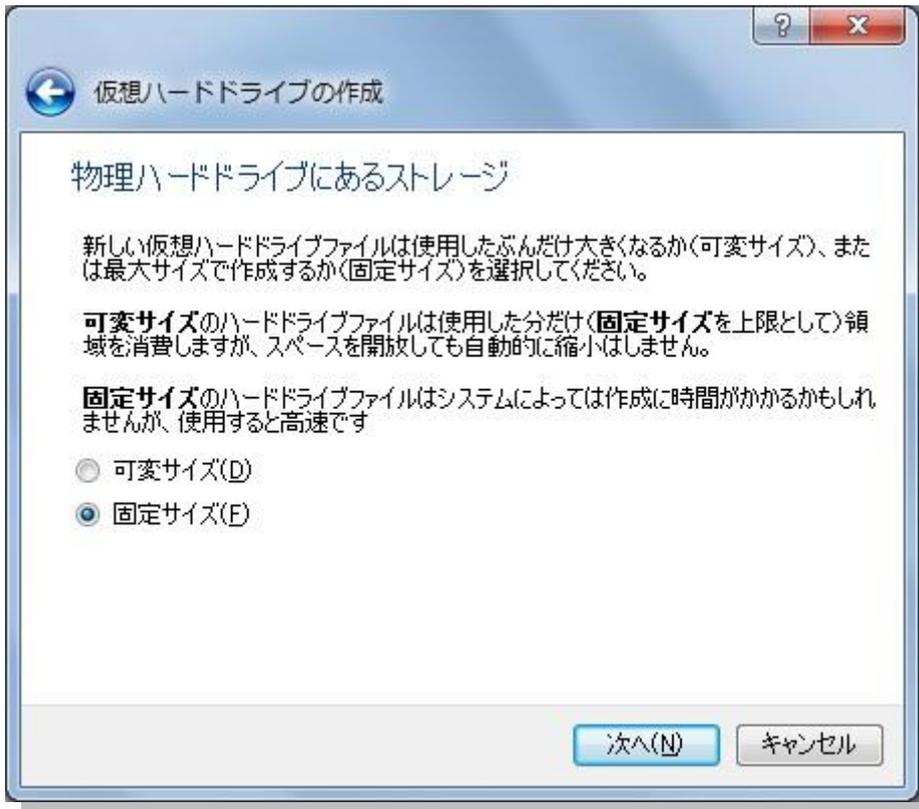
追加する仮想ハードディスクについて、「**新規ディスクの作成**」をクリックします。



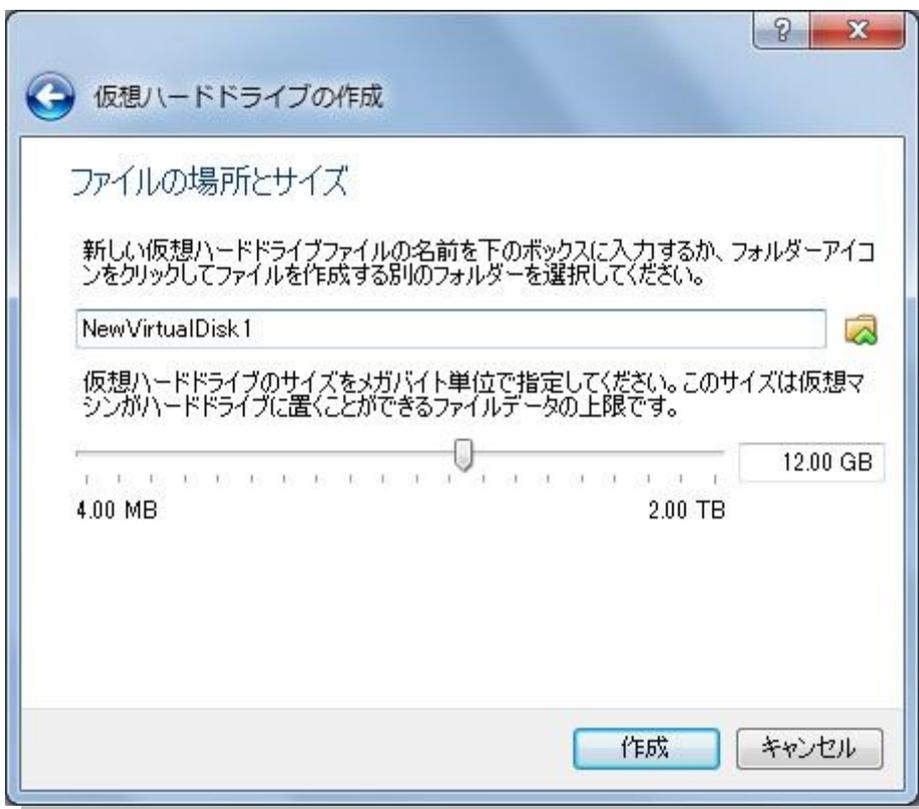
作成する仮想ハードドライブのファイルタイプには、「**VDI (VirtualBox Disk Image)**」を選択して「**次へ**」をクリックします。



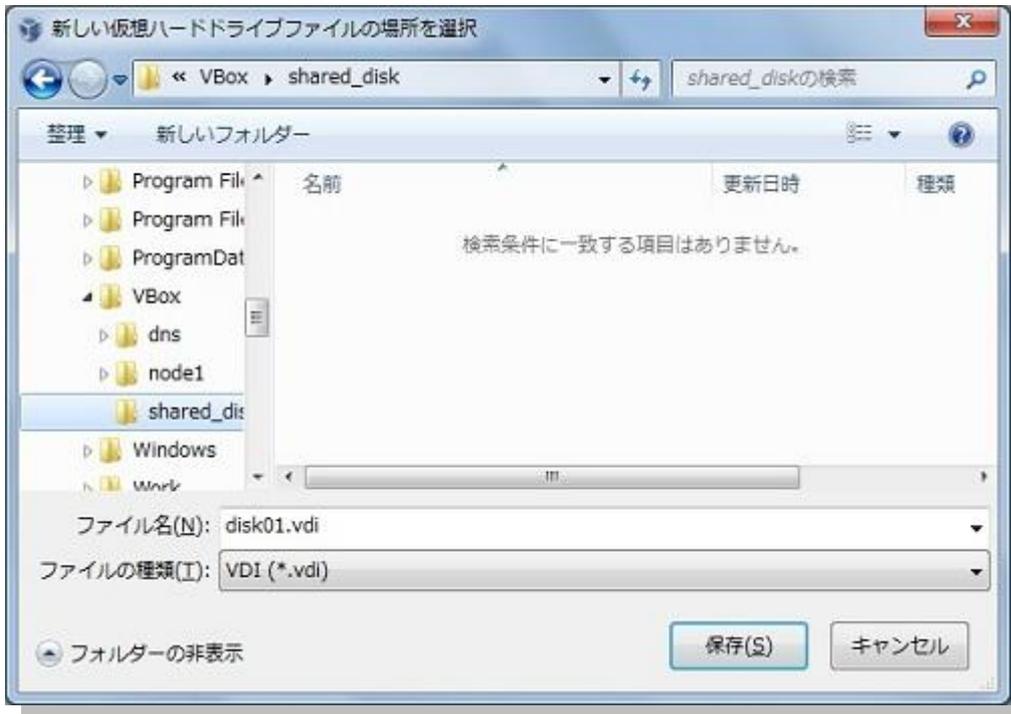
続いて、「固定サイズ」を選択して「次へ」をクリックします。



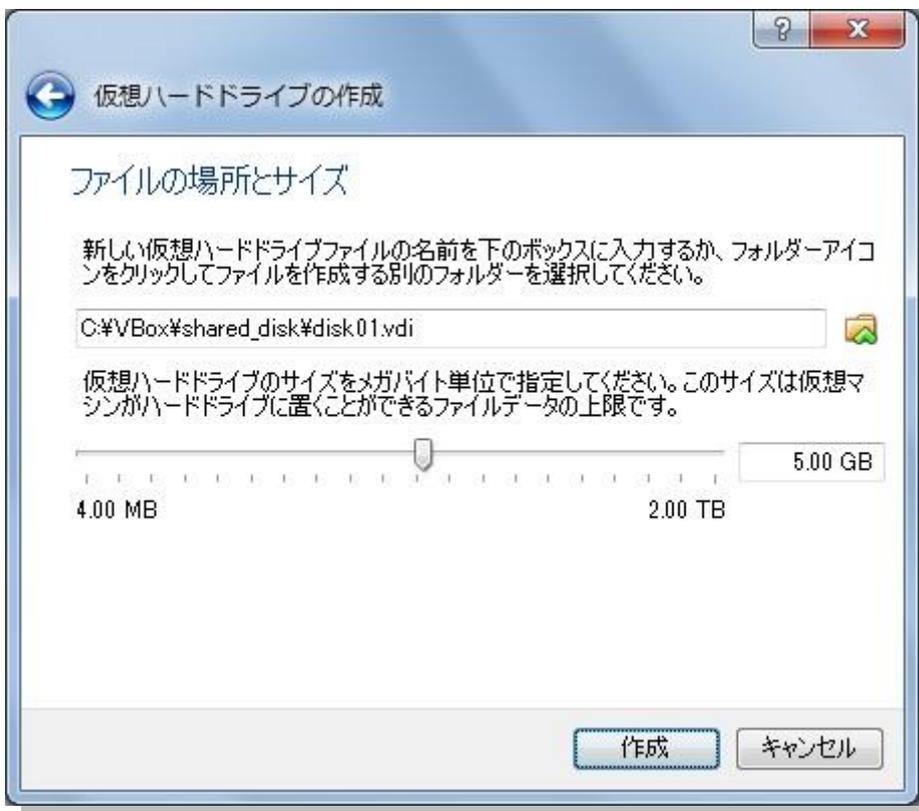
ファイルの場所とサイズを設定します。まず画面右の「フォルダー」のアイコンをクリックします。



ここでは、C:\%VBox%\shared\_disk フォルダを選択してファイル名を「disk01.vdi」として「保存」をクリックします。



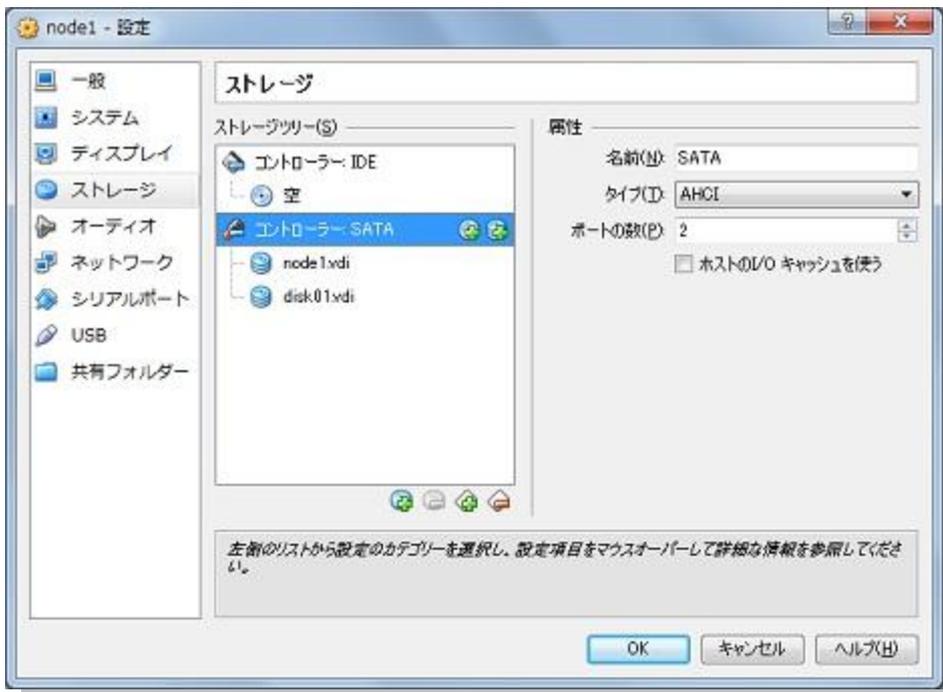
設定した場所を確認して、サイズには「5.00」GBを入力します。入力後「作成」をクリックします。



仮想ディスクの作成が開始されます。



作成の完了後、コントローラー : SATA に disk01.vdi が表示されていることを確認します。

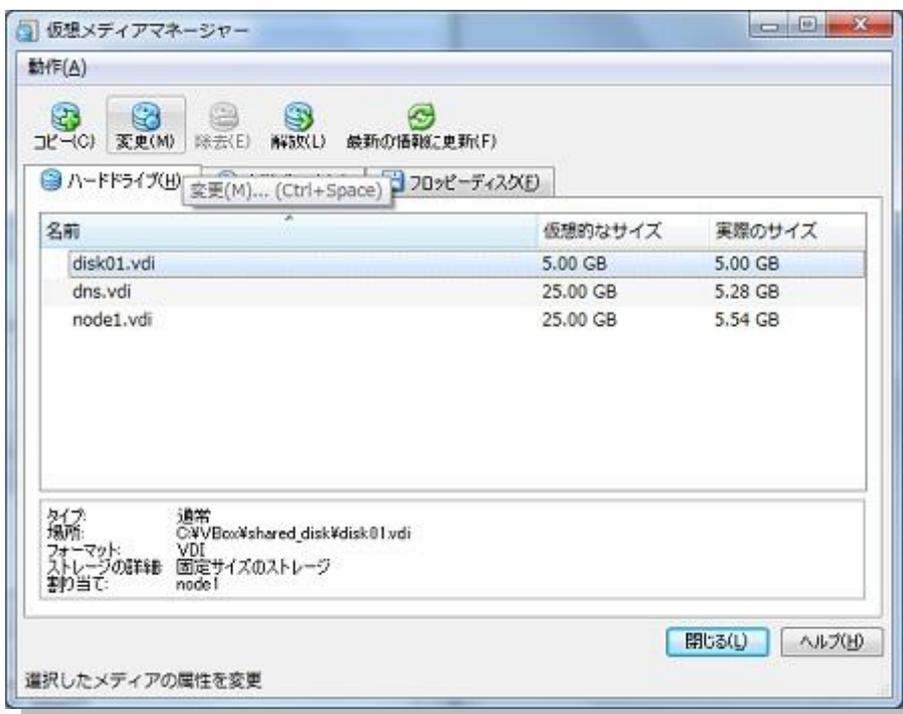


同様の手順で、他の仮想ディスクを作成します。今回は disk02.vdi から disk06.vdi までの残り5本を作成します。

必要な仮想ディスクの作成後、複数の仮想マシンから共有可能な設定に変更しておきます。「ファイル」メニューから「仮想メディアマネージャー」を選択します。



設定を変更したい仮想ディスクを選択して「変更」のアイコンをクリックします。ここではまず disk01.vdi を選択して設定変更を行うものとします。



属性を「共有可能」に設定して「OK」をクリックします。

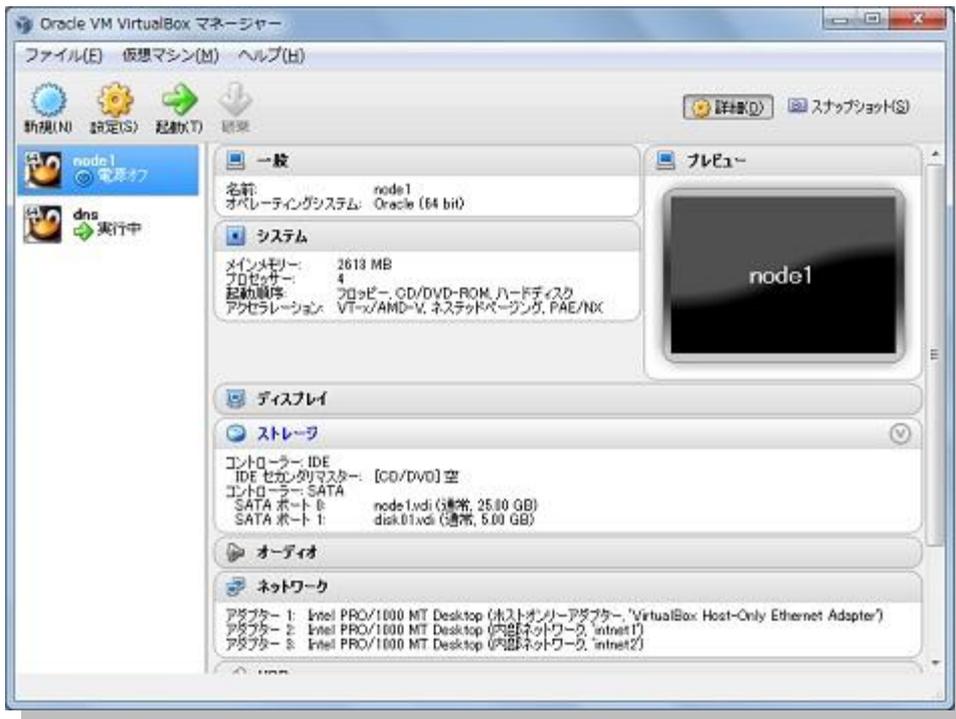


仮想メディアマネージャー画面の表示から、画面下部のタイプが「共有可能」と変更されたことを確認します。他に作成した仮想ディスクがある場合、同様の手順でタイプを共有可能に設定変更します。今回は disk02.vdi から disk06.vdi の 5 本に対して同様の手順で設定変更を行います。

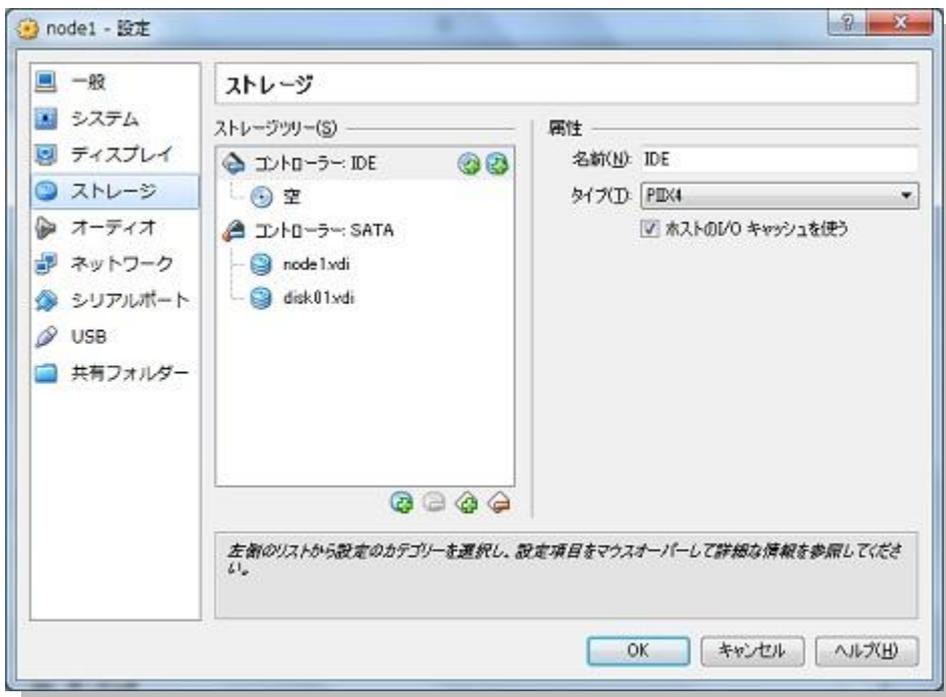
すべての仮想ディスクに設定変更した後、「閉じる」をクリックします。



続いて設定を反映させるため、「ストレージ」をクリックして設定画面を表示します。



コントローラー : SATA にすべてのディスクが表示されていることを確認して「OK」をクリックします。



「ストレージ」セクションの仮想ディスクの表示が「(共有可能 5.00 GB)」と変更されたことを確認します。



ここまでの手順を作成した仮想ディスク (disk02 から disk06 まで) に対して実施します。

仮想ディスクの作成は完了です。続いてゲスト OS (仮想マシン) 側での設定を行うため「node1」を選択して「起動」をクリックします。起動後は root ユーザーでログインします。

### 3. パーティションの作成

各ディスク・デバイス (/dev 配下の sdb から sdg まで) に単一パーティションを作成します。

```
# fdisk /dev/sdb
```

<実行例>

```
# fdisk /dev/sdb
コマンド (m でヘルプ) : n
コマンドアクション
  e  拡張
  p  基本領域 (1-4)
p
領域番号 (1-4) : 1
最初 シリンダ (1-652, default 1) : ← デフォルトの 1 を使用するので Enter
Using default value 1
終点 シリンダ または +サイズ または +サイズ M または +サイズ K (1-391, default 652) :
                                     ← デフォルトの 652 を使用するので Enter
Using default value 652
コマンド (m でヘルプ) : w
領域テーブルは交換されました！
ioctl() を呼び出して領域テーブルを再読み込みします。
ディスクを同期させます。
```

上記のコマンドを各ディスク・デバイスに対してそれぞれ実行します。

- OS 再起動後、各ディスク・デバイスに対してアクセス権限や所有グループが適切に設定されるように、udev のルール・ファイルを設定します。今回は、99-oracle.rules というファイル名で udev のルール・ファイルを新規作成して使用します。

```
# vi /etc/udev/rules.d/99-oracle.rules
```

次の記述をファイルに記述します。

```
KERNEL=="sd[b-z]1",ACTION=="add|change",OWNER="grid",GROUP="asmadmin",MODE="0660"
```

<実行例>

```
# vi /etc/udev/rules.d/99-oracle.rules
```

(以下 ASM ディスク用に 1 行で記述)

```
KERNEL=="sd[b-z]1",ACTION=="add|change",OWNER="grid",GROUP="asmadmin",MODE="0660"
```

- ルール・ファイルの設定を反映します。OS を再起動するか、次のコマンドを実行してください。

```
# udevadm control --reload-rules
```

```
# start_udev
```

<実行例>

```
# udevadm control --reload-rules
```

```
# start_udev
```

```
udev を起動中: [ OK ]
```

次のコマンドを実行してアクセス権と所有者が設定した内容で変更されたことを確認しておきます。

```
# ls -l /dev/sd*1
```

<実行例>

```
# ls -l /dev/sd*1
brw-r----- 1 root disk      8,  1 9月 16 10:26 /dev/sda1
brw-rw----  1 grid asmadmin 8, 17 9月 16 10:46 /dev/sdb1
brw-rw----  1 grid asmadmin 8, 33 9月 16 11:04 /dev/sdc1
brw-rw----  1 grid asmadmin 8, 49 9月 16 11:04 /dev/sdd1
brw-rw----  1 grid asmadmin 8, 65 9月 16 11:04 /dev/sde1
brw-rw----  1 grid asmadmin 8, 81 9月 16 11:04 /dev/sdf1
brw-rw----  1 grid asmadmin 8, 97 9月 16 11:04 /dev/sdg1
```

## 5.13 node2 用の仮想マシンの作成

node2 として使用する仮想マシンを作成します。

ここではインストール前に必要な事前設定が完了している node1 をクローンして node2 を作成するものとします。次の手順を実行して、node1 をクローンする前に仮想マシンの停止と仮想ディスクのデタッチを行います。

### 1. 仮想マシンの停止

次のコマンドを root ユーザーで実施して、一旦仮想マシンを停止します。

```
# shutdown -h now
```

### 2. 仮想ディスクのデタッチ

仮想ディスクを node1 からデタッチします。今回は disk01.vdi から disk06.vdi までの 6 本を node1 からデタッチします。

設定画面を表示し、デタッチする仮想ディスクを選択して「割り当て除去」のアイコンをクリックします。

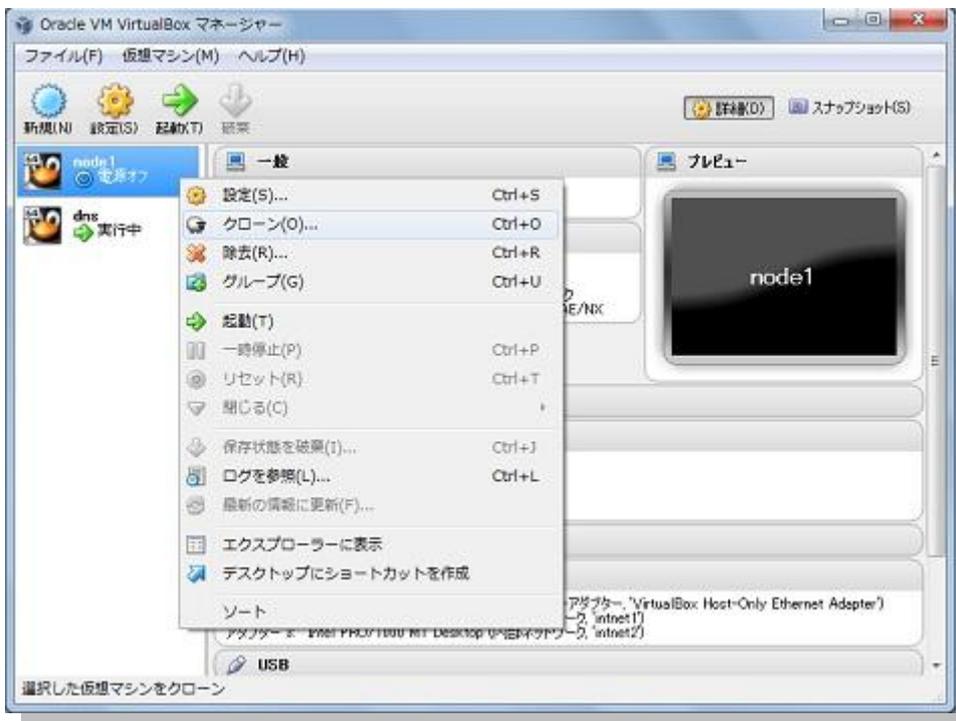


今回は disk01.vdi から disk06.vdi をデタッチします。デタッチ後、「OK」をクリックします。

※ node1.vdi はデタッチしないように注意してください。

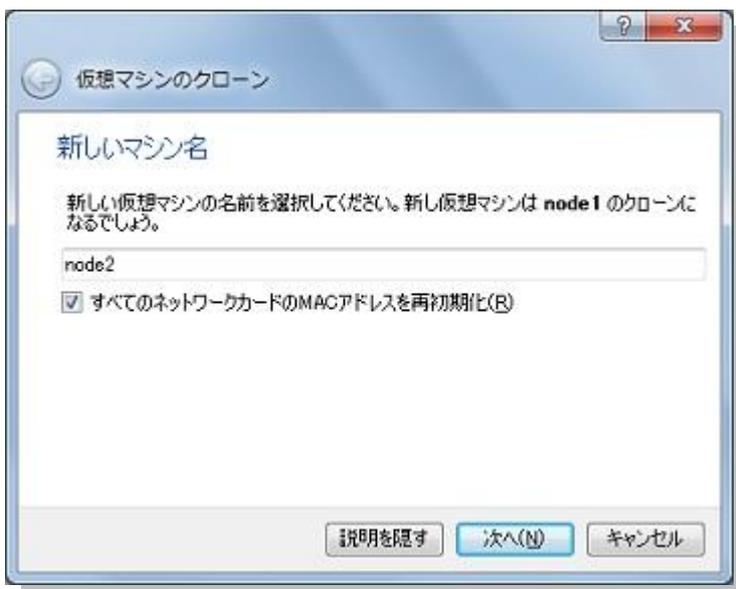
### 3. 仮想マシンのクローン

「node1」を選択して、右クリックで表示されるメニューから「クローン」を実行します。

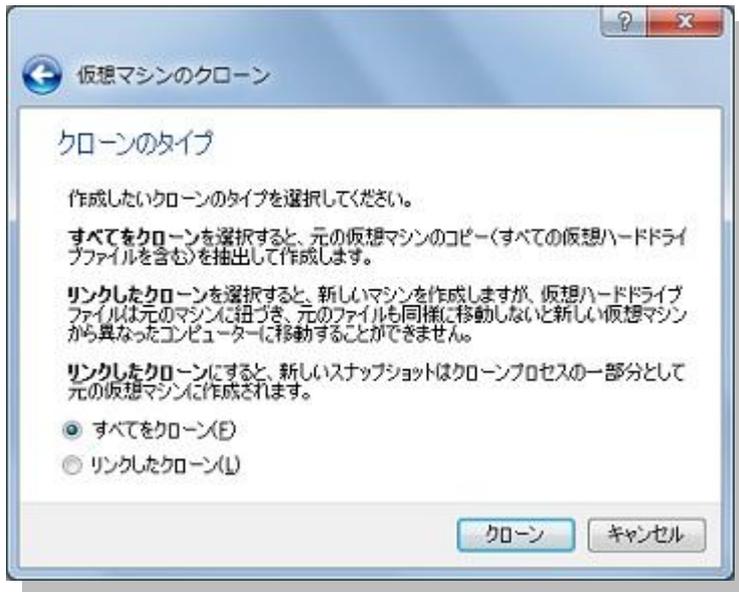


クローンによって作成する仮想マシンの名前を入力します。

ここでは入力フィールドに「node2」と入力して「すべてのネットワークカードの MAC アドレスを再初期化」にチェック (☑) します。入力後、「次へ」をクリックします。



クローンのタイプとして「すべてをクローン」を選択していることを確認して、「クローン」をクリックします。



クローンの完了後、node2 が表示されます。

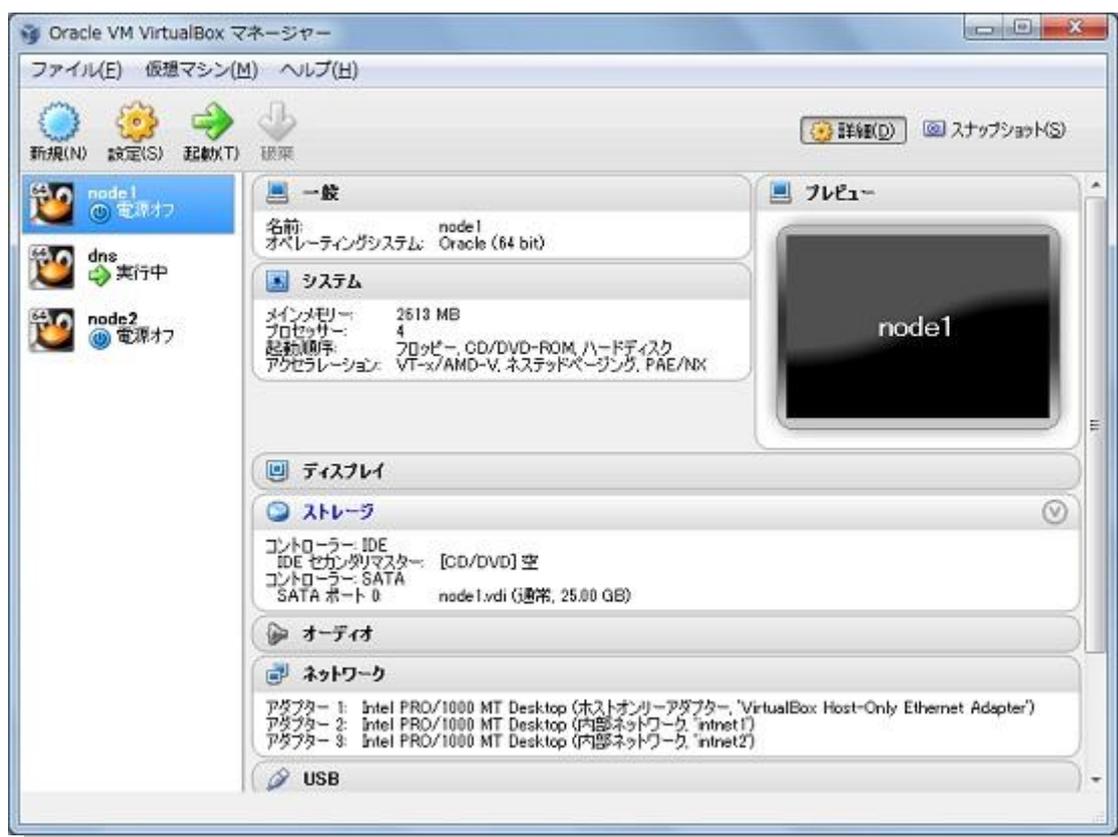


#### 4. 仮想ディスクのアタッチ

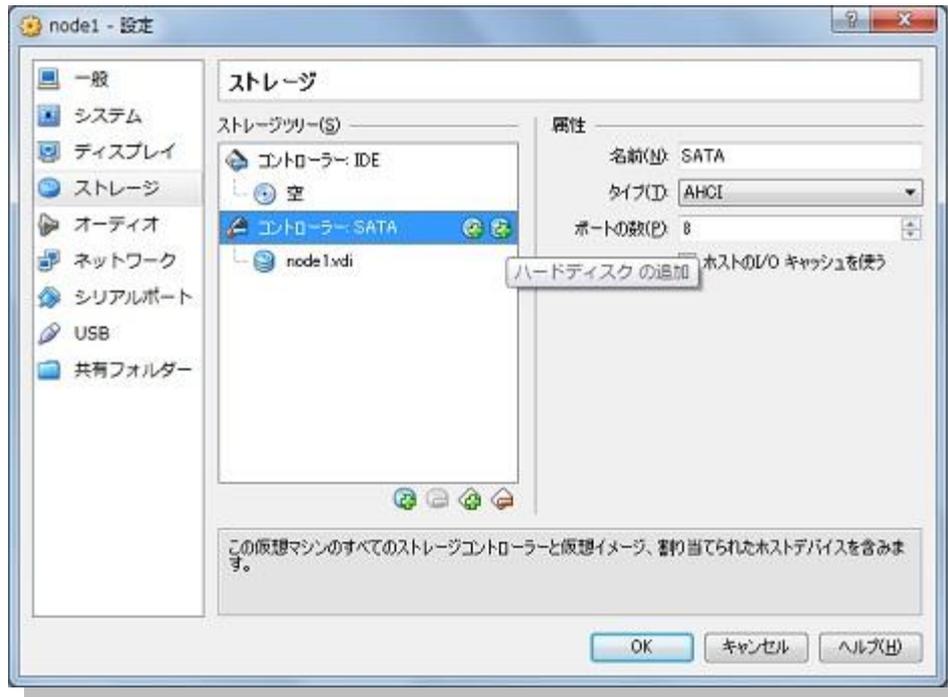
node1 および node2 に仮想ディスクをアタッチします。ここでは disk01.vdi から disk06.vdi までの 6 本をアタッチするものとします。

次の手順では、まず node1 に仮想ディスクをアタッチしています。同様の手順で node2 にも仮想ディスクのアタッチを実行してください。

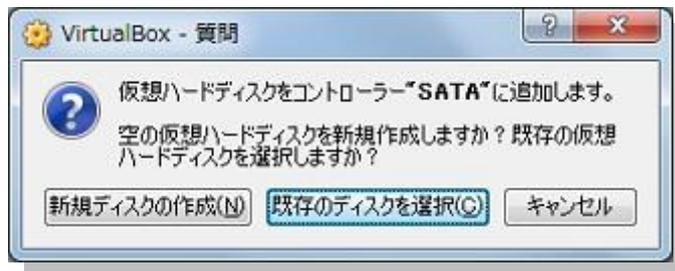
「node1」を選択して「ストレージ」をクリックして設定画面を表示します。



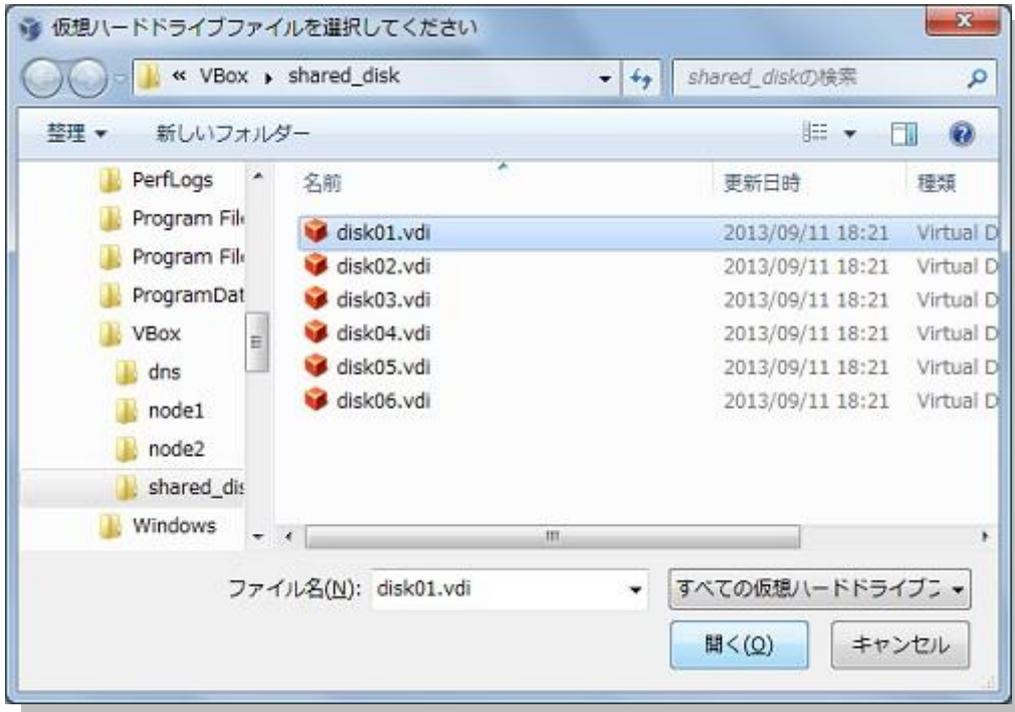
「コントローラー : SATA」を選択して「ハードディスクの追加」のアイコンをクリックします。



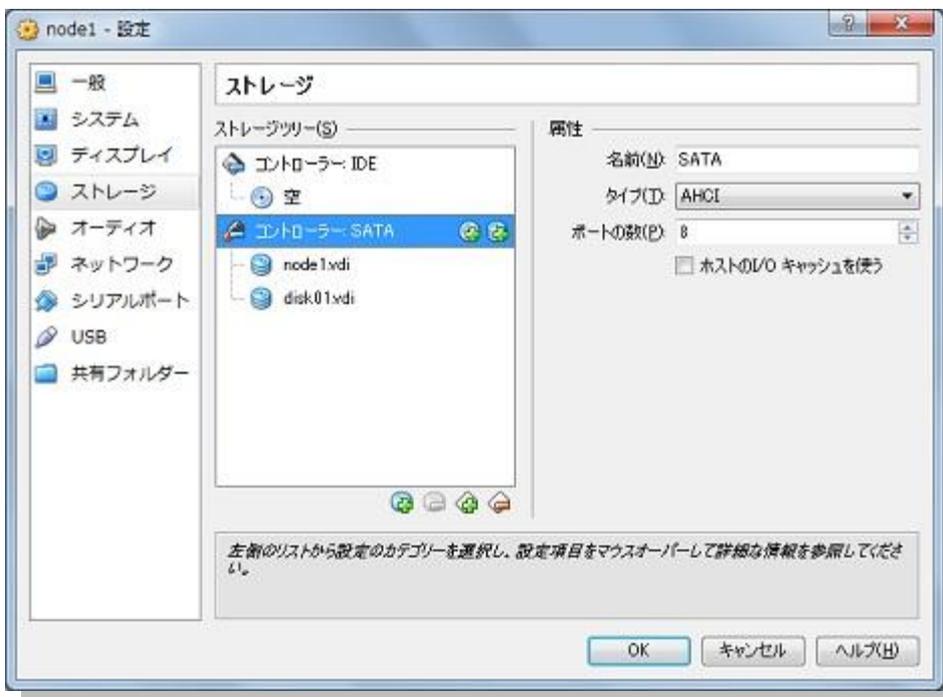
「既存のディスクを選択」を選択します。



ディスクの選択画面でアタッチする仮想ハードドライブファイルを選択します。ここではまず、「disk01.vdi」をダブルクリック、または選択して「開く」をクリックします。



選択した仮想ディスクがアタッチされていることを確認します。同様の手順ですべての仮想ディスクをアタッチしてください。

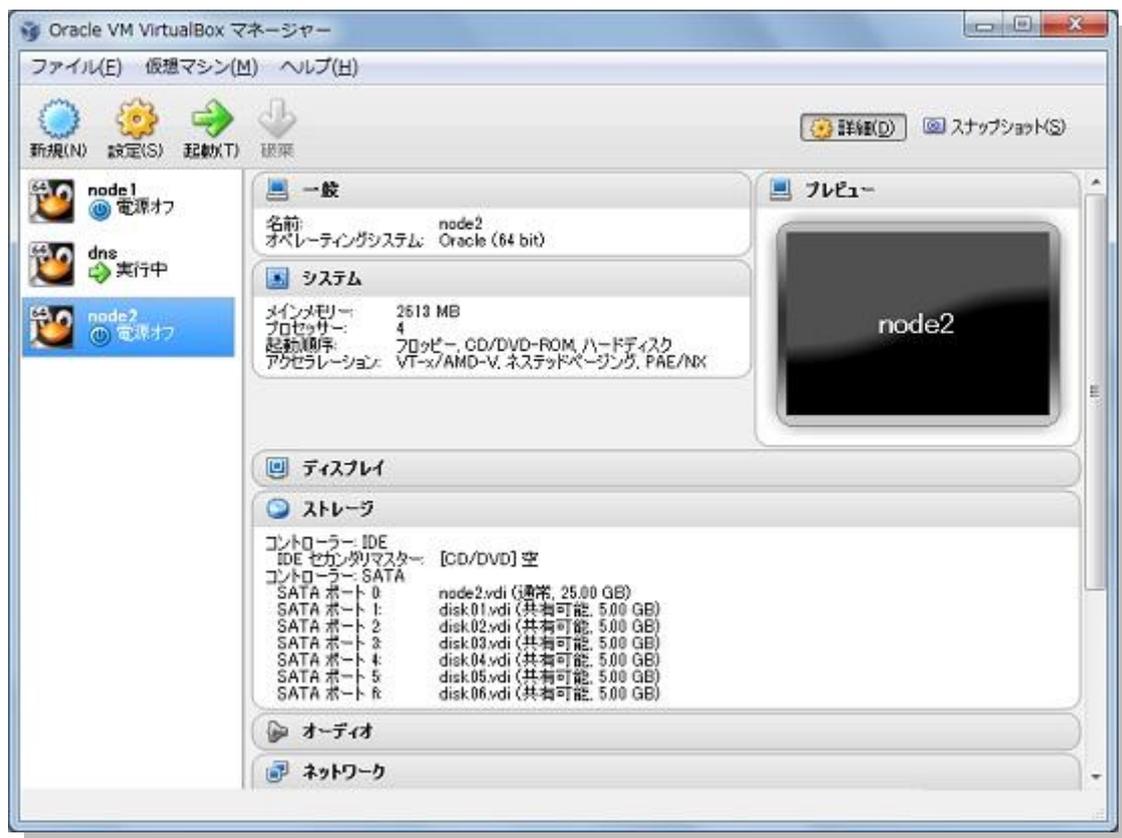


完了後、「OK」をクリックして画面を閉じます。

続いて node2 にも仮想ディスクをアタッチします。この「4. 仮想ディスクのアタッチ」で node1 に実行した手順を node2 にも実行して、仮想ディスクをアタッチします。

手順が完了した後、確認として Oracle VM VirtualBox マネージャー画面のストレージ・セクションの表示を確認します。

node1 および node2 それぞれで、すべての仮想ディスク (今回は disk01.vdi から disk06.vdi までの 6 本) が表示されていることを確認します。



## 5. クローン後の設定変更

node2 を起動して、設定変更を行います。「**node2**」を選択して「**起動**」をクリックします。

起動後、root ユーザーでログインします。/etc/sysconfig/network を編集して、ホスト名を変更します。

```
# vi /etc/sysconfig/network
```

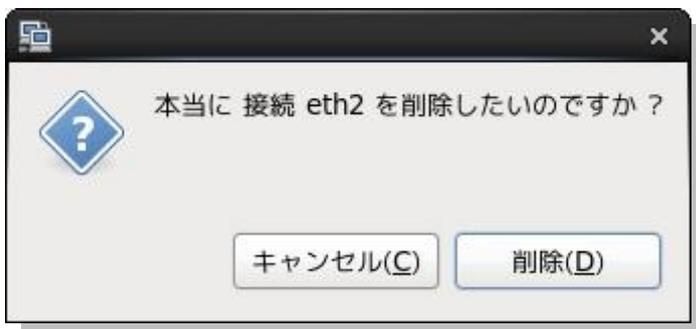
## &lt;編集例&gt;

```
[root@node1 ~]# vi /etc/sysconfig/network  
< 次の一行を変更する >  
HOSTNAME=node1.oracle12c.jp  
< 変更後 >  
HOSTNAME=node2.oracle12c.jp
```

続いて「システム」メニューの「設定」の中から「ネットワーク接続」を選択します。設定画面が表示されたら「eth0」、「eth1」、「eth2」をそれぞれ選択して「削除」をクリックします。



削除の確認メッセージを確認して「削除」をクリックします。



eth0 および eth1 も同様の手順で削除します。

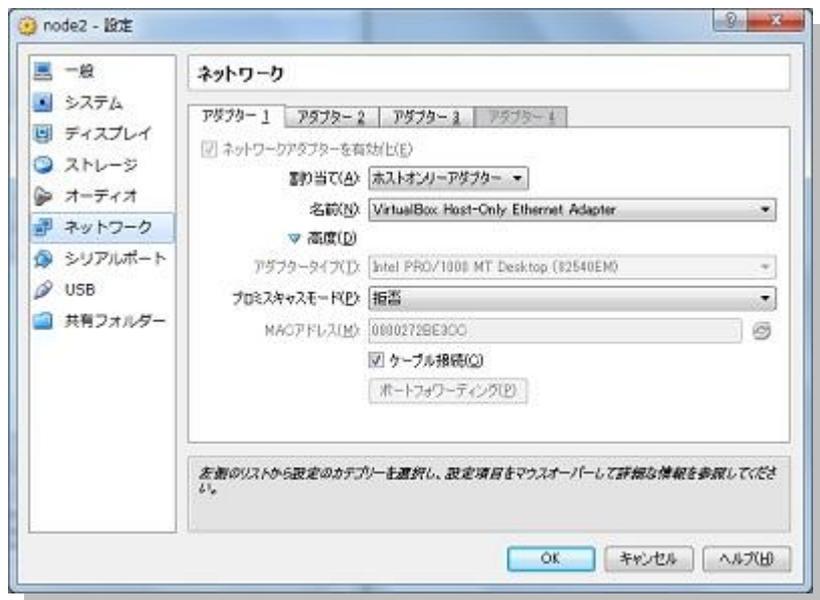
「Auto eth3」を選択して「編集」をクリックします。



接続名を「eth0」に変更します。node2 のネットワークの設定は「デバイスの MAC アドレス」に表示されている MAC アドレスと、各アダプターの MAC アドレスが一致することを確認しながら設定をします。



併せて確認する各アダプターの MAC アドレスは、Oracle VM VirtualBox マネージャーの設定画面から確認できます。グレーアウトされている部分の MAC アドレスを確認して、次のように対応するように設定します。



- アダプター1 と eth0
- アダプター2 と eth1
- アダプター3 と eth2

eth0 の設定に戻ります。「IPv4 のセッティング」タブより、方式を「手動」にして「追加」からアドレスとネットマスクを入力します。

ここではアドレスに「192.168.56.102」、ネットマスクに「255.255.255.0」と設定するものとします。DNS サーバーには「192.168.56.254」を、ドメインを検索には「oracle12c.jp」を設定します。設定後、「適用」をクリックします。



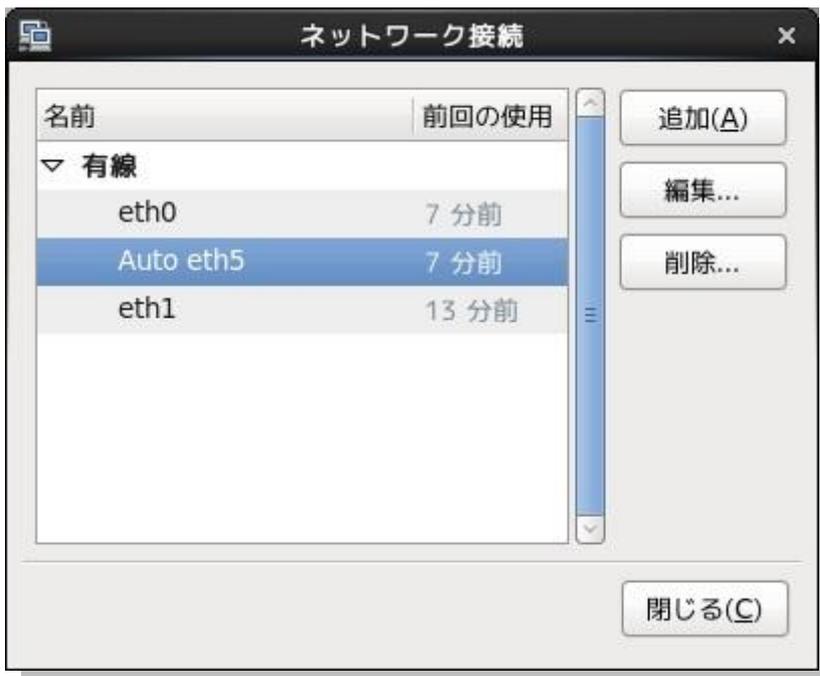
続いて「Auto eth4」を選択して、「編集」をクリックします。



接続名を「eth1」に変更して、「自動接続する」にチェック (☑) がされていることを確認します。続いて「IPv4 の設定」タブより、方式を「手動」にして「追加」からアドレスとネットマスクを入力します。ここではアドレスに「192.168.100.102」、ネットマスクに「255.255.255.0」と設定するものとします。設定後、「適用」をクリックします。



続いて「Auto eth5」を選択して「編集」をクリックします。



接続名を「eth2」に変更して、「自動接続する」にチェック (☑) がされていることを確認します。続いて「IPv4 の設定」タブより、方式を「手動」にして「追加」からアドレスとネットマスクを入力します。ここではアドレスに「192.168.200.102」、ネットマスクに「255.255.255.0」と設定するものとします。設定後、「適用」をクリックします。



ネットワークの設定変更は以上です。「閉じる」をクリックします



ネットワーク・インターフェースの情報を初期化するために設定ファイルを削除します。root ユーザーで次のコマンドを実行します。

```
# cd /etc/udev/rules.d
```

```
# ls 70-persistent-net.rules
```

```
# rm -rf 70-persistent-net.rules
```

<実行例>

```
[root@node1 ~]# cd /etc/udev/rules.d
[root@node1 rules.d]# ls 70-persistent-net.rules
70-persistent-net.rules
[root@node1 rules.d]# rm -rf 70-persistent-net.rules
```

設定を反映するために、一旦再起動します。root ユーザーで次のコマンドを実行します。

```
# shutdown -r now
```

node2 の設定は完了です。クローンの作業はすべて完了したので、node1 も起動しておきます。

この時点で、次の 3 つの仮想マシンが起動していることを確認します。

- node1
- node2
- dns

## 6. Oracle Grid Infrastructure インストールおよび構成

ここからソフトウェアのインストール手順について説明します。クラスタ環境を構築するには Oracle Grid Infrastructure と Oracle Database の 2 つのソフトウェアが必要です。

本ガイドでは、初期リリースである Oracle Database 12c Release 1 (12.1.0.1) を使用します。ソフトウェアは Oracle Technology Network (OTN) よりダウンロードすることが可能です。

( <http://www.oracle.com/technetwork/jp/database/enterprise-edition/downloads/index.html> )

はじめに、事前準備としてソフトウェアの準備とインストールを行う OS ユーザーでのログインを実施します。

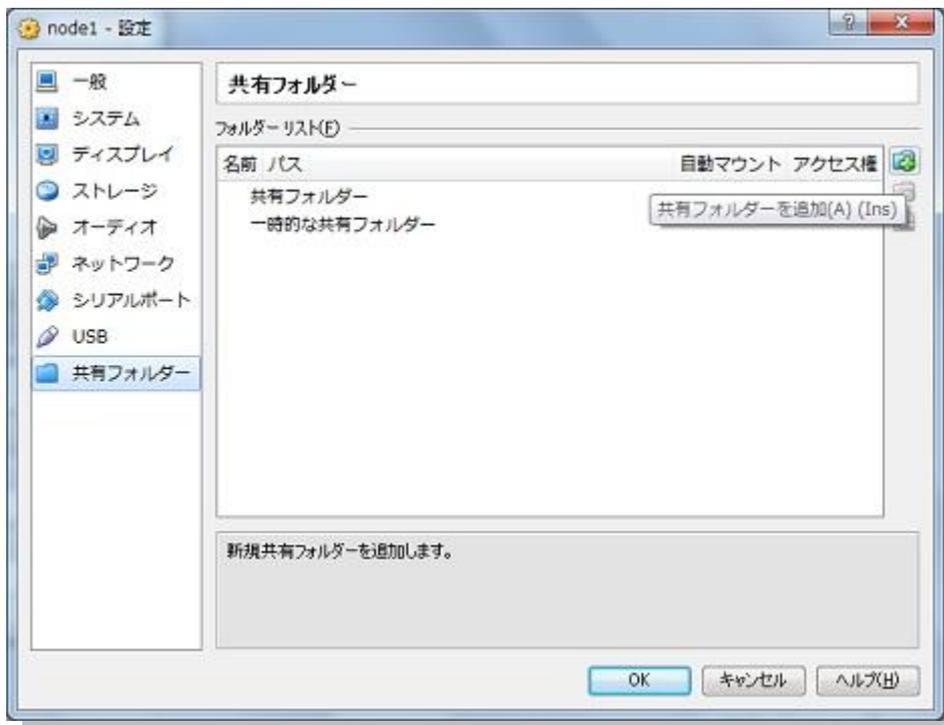
### 1. ソフトウェアの準備

はじめに、Oracle Grid Infrastructure と Oracle Database のインストールに必要なソフトウェアを仮想マシンである node1 上へ配置します。仮想マシンへ FTP、SCP プロトコルなどでソフトウェアを直接送って配置することもできますが、ここでは、Oracle VM VirtualBox の Guest Additions で提供されている共有フォルダー機能 (ホスト OS とゲスト OS 間でのファイル共有機能) を利用するものとします。

まず、ホスト OS (Oracle VM VirtualBox を起動している Windows マシン) 側で、ダウンロードしたソフトウェアを任意の場所に配置します。ここでは次の場所に配置したものとして進めます。

C:\software\oracle

Oracle VM VirtualBox マネージャー画面において「node1」を選択して「設定」をクリックします。設定画面が表示されたら、左ペインから「共有フォルダー」を選択します。続いて、右側の「共有フォルダーを追加」のアイコンをクリックします。



共有したいフォルダーのパスを「フォルダーのパス」に、ゲスト OS (Oracle Linux 6 Update 4) でマウントするときの名前を「フォルダー名」に設定します。ここでは、「C:\software\oracle」を「フォルダーのパス」に、フォルダー名に「oracle」を設定し、「OK」をクリックします。なお、node1 が稼働中の場合は、仮想マシンを再起動したときに継続して設定を有効にする「永続化する」の選択(☑)も可能です。設定の永続化は任意です。



続いて、node1 に root ユーザーでログインし、共有フォルダーをディレクトリにマウントします。本ガイドでは、マウント先のディレクトリとして「/opt/image」を作成し、マウントを行います。

```
# mkdir /opt/image
```

```
# mount -t vboxsf oracle /opt/image
```

<実行例>

```
# mkdir /opt/image
# mount -t vboxsf oracle /opt/image
```

続けて、次のコマンドでソフトウェアを展開 (unzip) しておきます。

```
# cd /opt/image
```

```
# ls
```

```
# unzip <DOWNLOADED_ZIP_FILE_NAME>
```

<実行例>

```
# cd /opt/image
# ls
linuxamd64_12c_database_1of2.zip  linuxamd64_12c_grid_1of2.zip
linuxamd64_12c_database_2of2.zip  linuxamd64_12c_grid_1of2.zip
# unzip linuxamd64_12c_grid_1of2.zip
※ unzip はソフトウェアの使用前にすべてのファイルに対して実行しておきます。
```

## 2. OUI の起動

インストールを行う OS ユーザー (ここでは grid ユーザー) で OUI を起動します。

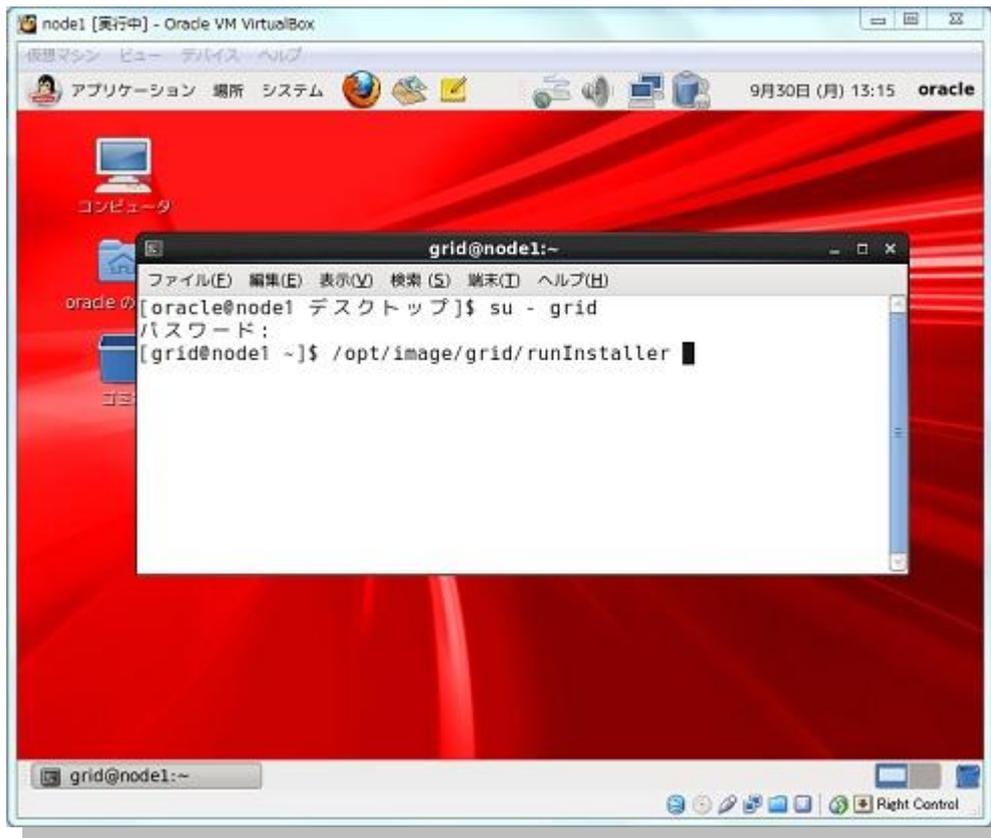
本ガイドの設定において、root ユーザーでシステムにログインしている場合、grid ユーザーにユーザーを変更して OUI の起動を試行しても OUI の起動ができません。ここではまず Oracle VM VirtualBox 画面の「システム」の「root のログアウト」を選択して、一旦 root ユーザーからログアウトします。確認画面では「ログアウト」を選択してください。



ログアウト後は、oracle ユーザーで再度ログインします。Oracle VM VirtualBox 画面上の端末から、次のコマンドでインストールを実行するユーザー (ここでは grid ユーザー) に変更します。続いて、インストールを実行するユーザーで OUI を起動します。

```
$ su - grid
```

```
$ /opt/image/grid/runInstaller
```



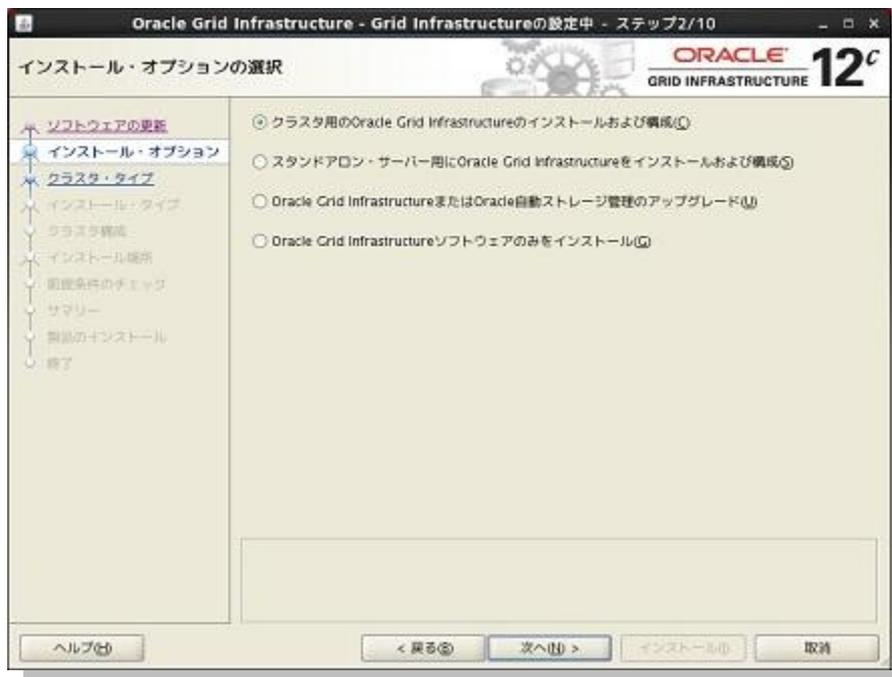
### 3. Software Update のダウンロード

インストール中に最新のパッチなどの更新をダウンロードして適用するためのオプションとして、ソフトウェア更新のダウンロードオプションが提供されています。ここでは更新のダウンロードや適用は行わないものとしますので、「ソフトウェア更新のスキップ」を選択して「次へ」をクリックします。



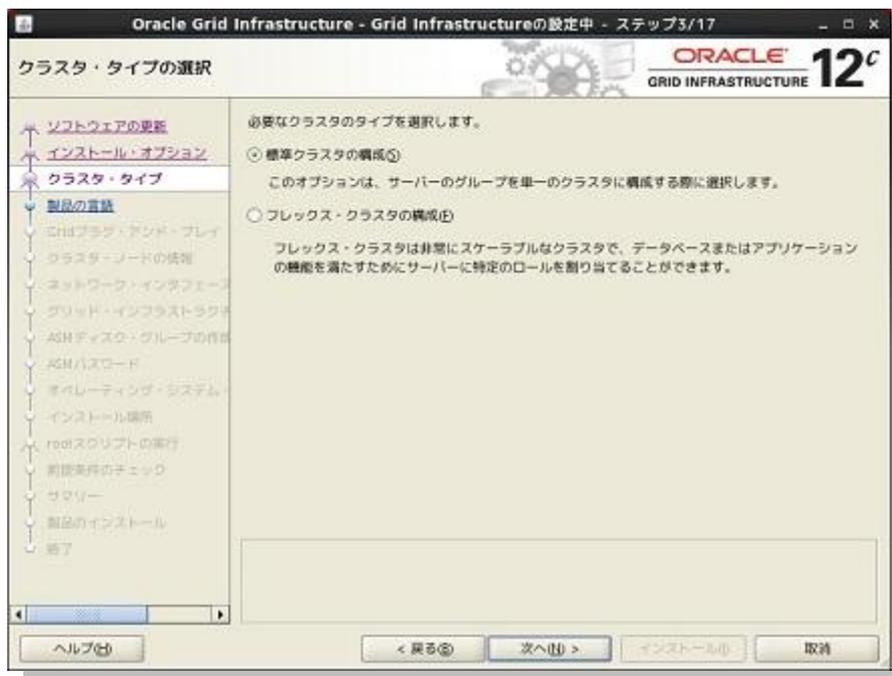
#### 4. インストール・オプションの選択

インストールのオプションを選択します。「クラスタ用の Oracle Grid Infrastructure のインストールおよび構成」を選択して、「次へ」をクリックします。



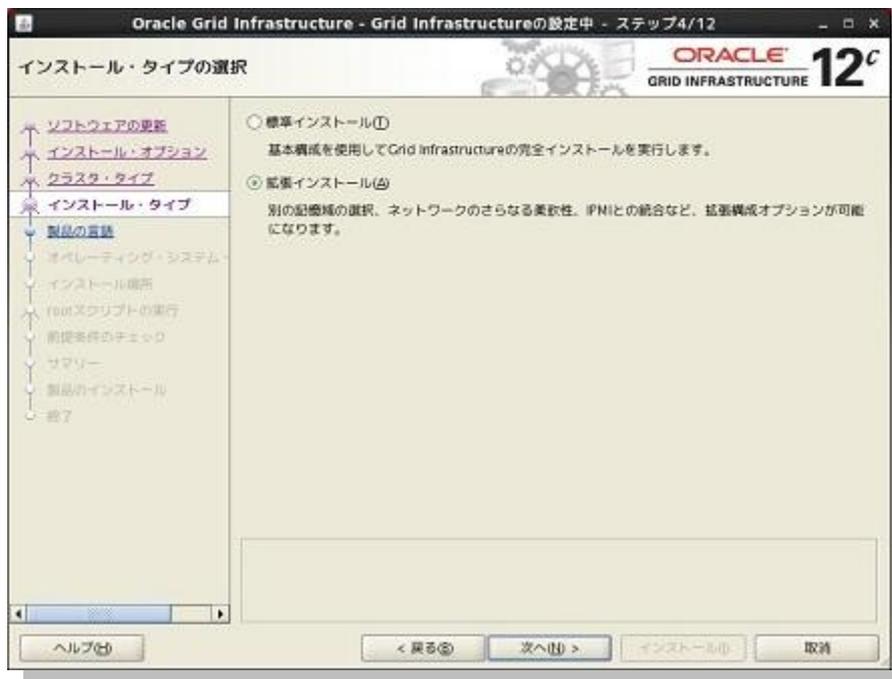
#### 5. クラスタ・タイプの選択

構成するクラスタのタイプを選択します。ここでは「標準クラスタの構成」を選択して「次へ」をクリックします。



## 6. インストール・タイプの選択

実行するインストールのタイプを選択します。ここでは、記憶域やネットワークの設定を実施するため「拡張インストール」を選択して「次へ」をクリックします。



## 7. 製品言語の選択

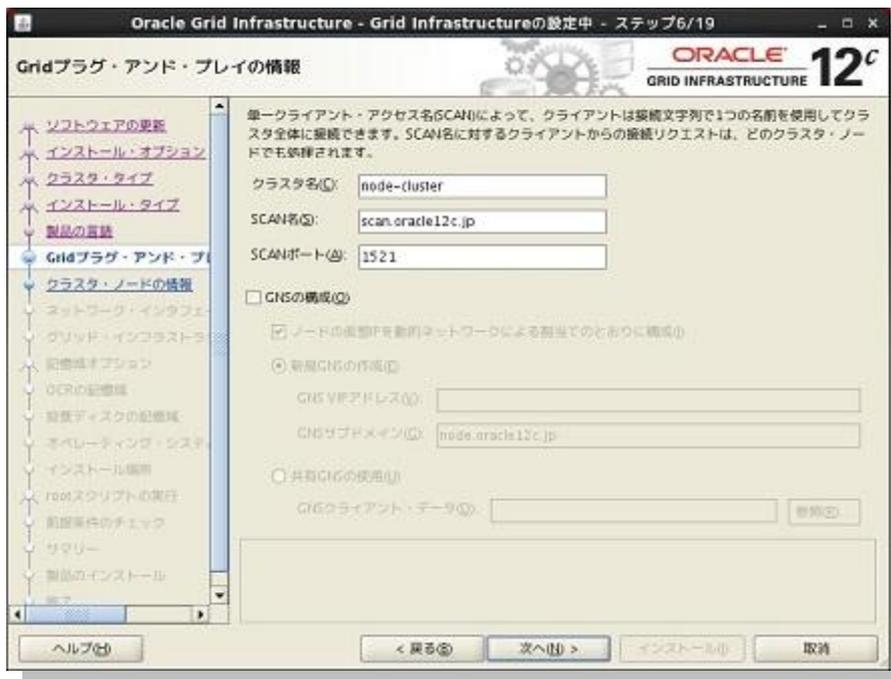
製品を実行する言語を選択します。ここでは、製品を実行する言語として「日本語」と「英語」が選択されていることを確認して「次へ」をクリックします。



## 8. Grid プラグ・アンド・プレイの情報

クラスタ名と SCAN の情報を入力します。クラスタ名と SCAN ポートは任意の値で構いません。SCAN 名には、事前に DNS あるいは GNS に登録しておいた名前を入力します。

ここではクラスタ名として「**node-cluster**」と入力し、SCAN 名には「**scan.oracle12c.jp**」と入力します。また、GNS は構成しないため「**GNS の構成**」のチェック・ボックスについているチェック (☑) を外して「**次へ**」をクリックします。

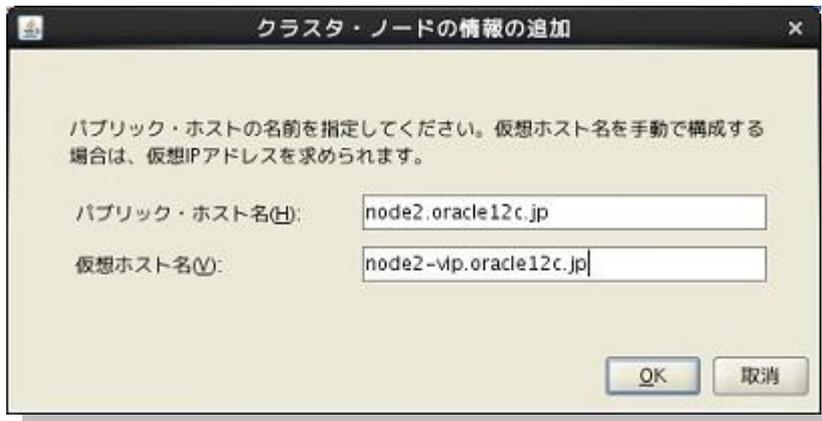


## 9. クラスタ・ノードの情報

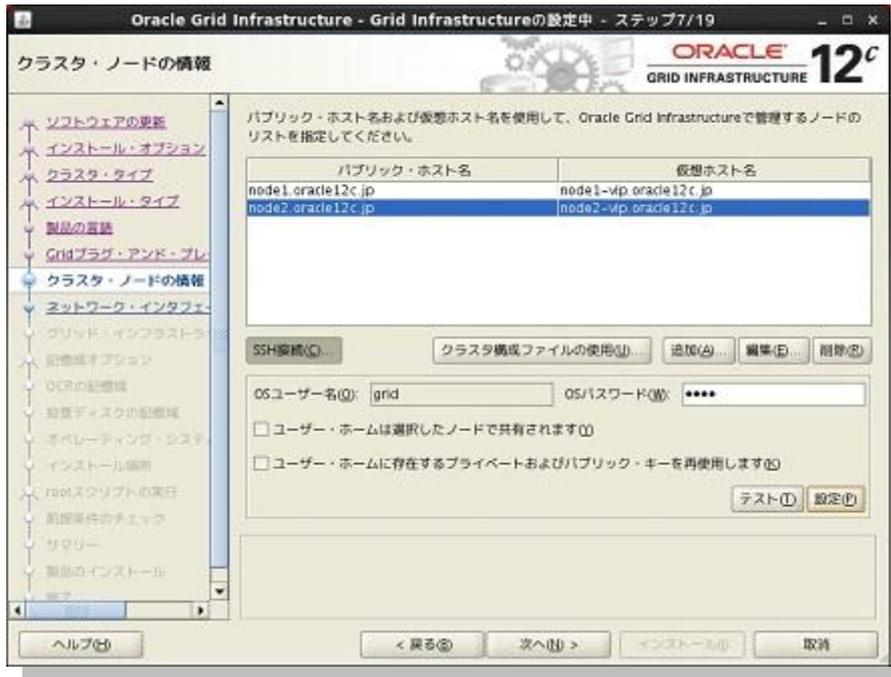
クラスタを構成するすべてのノード情報 (ホスト名と仮想 IP 名) を入力します。OUI を起動しているノード (ここでは node1) の情報は表示されているので、確認の上、「追加」をクリックして 2 ノード目の情報を追加します。



2 ノード目の情報として、ここでは「node2.oracle12c.jp」と「node2-vip.oracle12c.jp」を入力して「OK」をクリックします。



続いて、ノード間でパスワード不要の SSH 接続を構成するために「SSH 接続」をクリックします。クリック後、SSH 接続の構成に必要な情報を入力するフィールドが表示されますので、OUI を起動している OS ユーザーの OS パスワードを入力して、「設定」をクリックします。



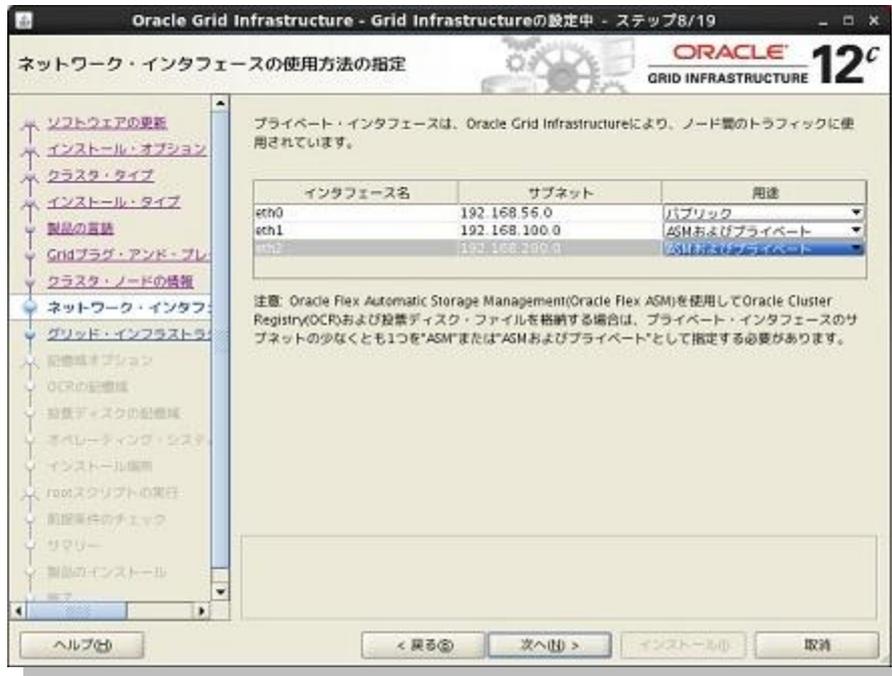
OUIによりSSH接続の確立が試行されます。次の画面が表示されたら「OK」をクリックします。



SSH接続が確立したので、「次へ」をクリックします。

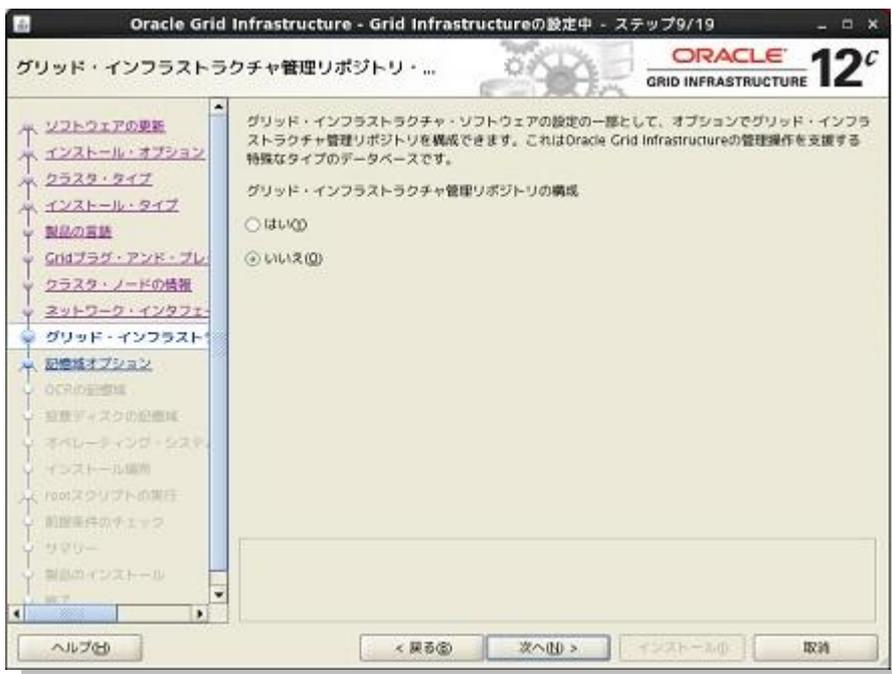
## 10. ネットワーク・インターフェースの使用法の指定

ネットワーク・インターフェースの使用法をプルダウンより選択します。ここでは、eth0 を「パブリック」、eth1 と eth2 を「ASM およびプライベート」として設定します。選択後「次へ」をクリックします。



## 11. グリッド・インフラストラクチャ管理リポジトリの構成

管理リポジトリの構成を設定します。今回の環境にはリソースに限りがあるため、ここでは「いいえ」を選択して「次へ」をクリックします。

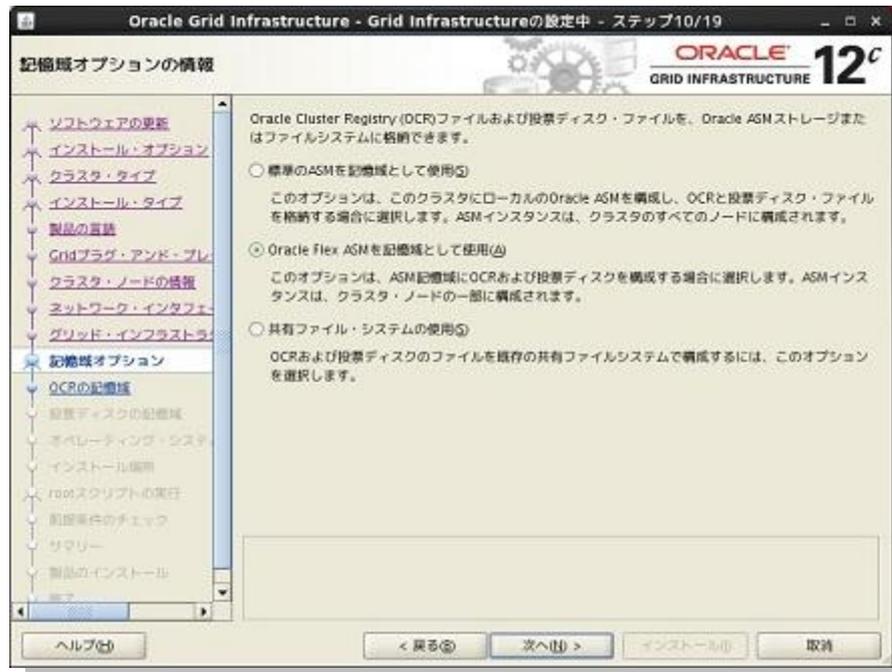


次のような警告メッセージが表示されたら、今回は確認して「はい」をクリックします。



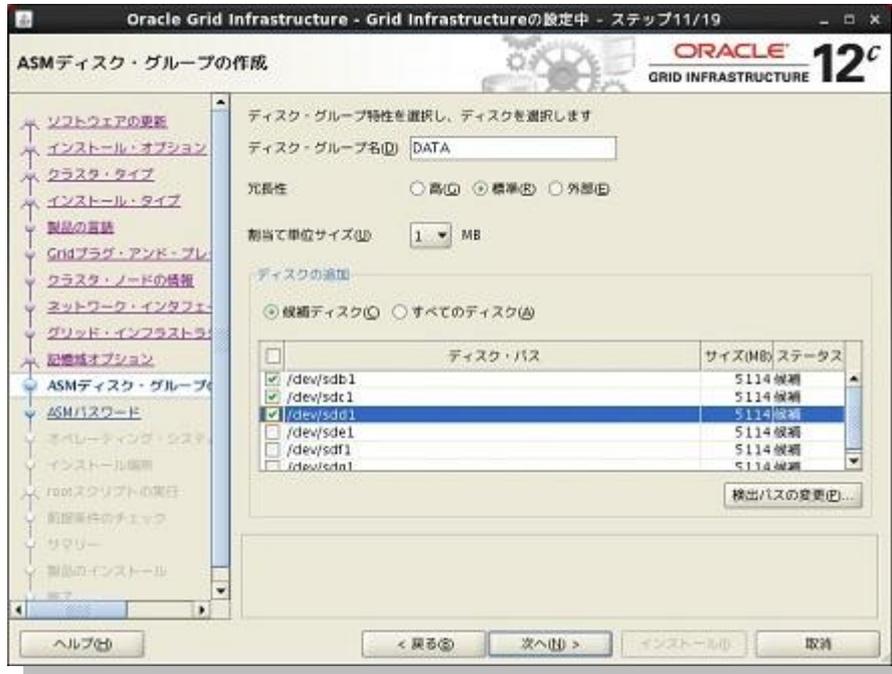
## 12. 記憶域オプションの情報

OCR と投票ディスクの格納場所を選択します。ここでは「Oracle Flex ASM を記憶域として使用」を選択して「次へ」をクリックします。



## 13. ASM ディスク・グループの作成

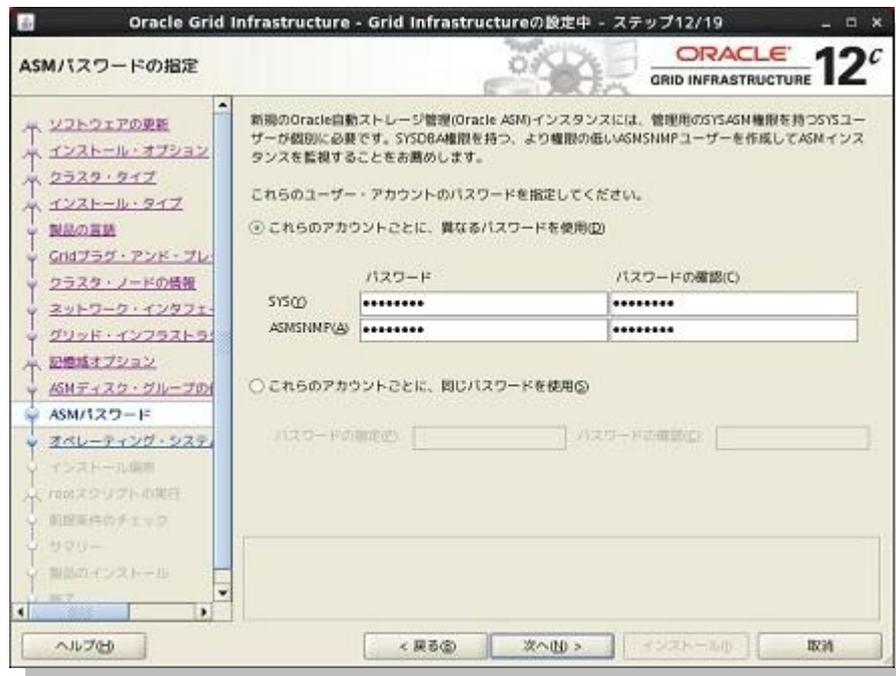
ASM ディスク・グループの作成に必要な情報を入力します。ここでは、ディスク・グループ名に「DATA」、冗長性には「標準」を選択します。続いて ASM ディスク・グループを構成するディスクとして、ここでは「/dev/sdb1、/dev/sdc1、/dev/sdd1」をチェック (☑) して、「次へ」をクリックします。



- ※ 標準の冗長性を選択した場合には ASM ディスク・グループに格納されるデータは 2 重化による冗長構成がとられます。今回のように、標準の冗長構成をとる ASM ディスク・グループに OCR と投票ディスクを格納する場合は、OCR が 2 つと投票ディスクが 3 つ作成されるため、これらのファイルを格納するために、最低でも 3 つのディスクを ASM ディスク・グループの構成に使用する必要があります。
- ※ 今回の構成とディスク名や配置場所が異なる場合には、「検出パスの変更」よりディスク検出パスの変更を行ってください。

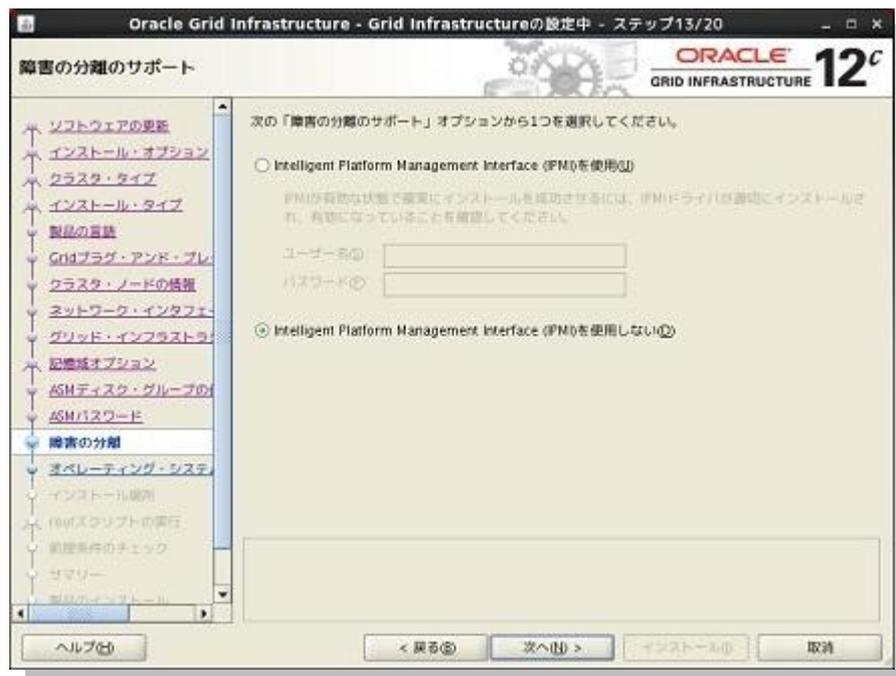
## 14. ASM パスワードの指定

ASM に対する管理者のパスワードを設定します。ここでは「これらのアカウントごとに、異なるパスワードを使用」を選択して、SYS および ASMSNMP ユーザーにそれぞれ任意のパスワードを設定します。設定後、「次へ」をクリックします。



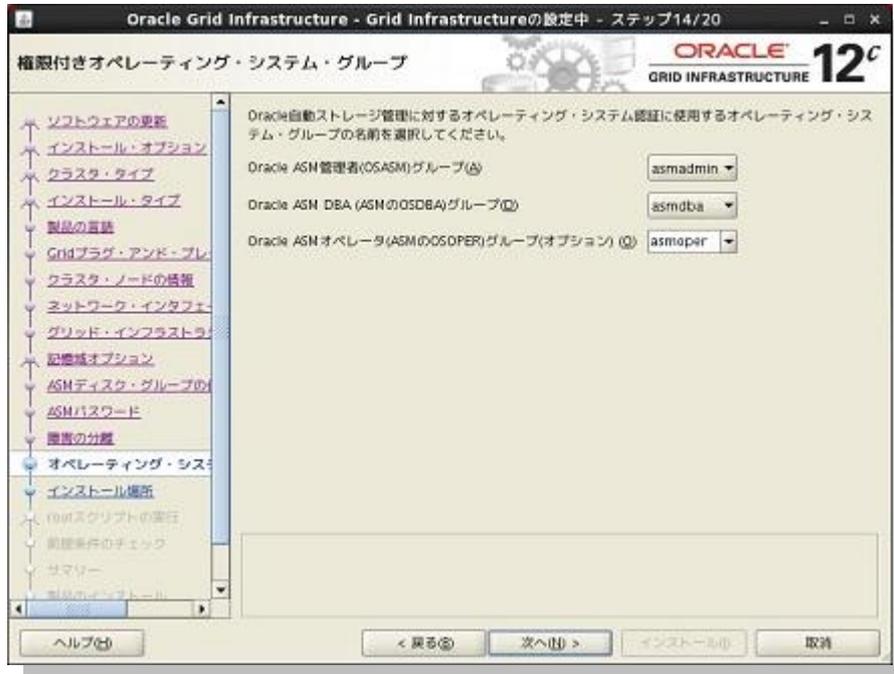
## 15. 障害の分離のサポート

障害発生時に Oracle Clusterware や OS に依存せずに障害ノードを停止させる外部メカニズムとして Intelligent Platform Management Interface (IPMI) の使用を設定します。ここでは「Intelligent Platform Management Interface (IPMI) を使用しない」を選択して「次へ」をクリックします。



## 16. 権限付き OS グループの選択

ASM に対する OS 認証に使用する OS グループを設定します。ここでは Oracle ASM 管理者 (OSASM) グループには「**asmadmin**」、Oracle ASM DBA (ASM の OSDBA) グループに「**asmdba**」、Oracle ASM オペレータ (ASM の OSOPER) グループに「**asmoper**」として「次へ」をクリックします。



- ※ Oracle ASM DBA (ASM の OSDBA) グループおよび Oracle ASM 管理者 (OSASM) グループとしてプルダウンより選択できる OS グループは Oracle Grid Infrastructure のインストール・ユーザー (ここでは grid ユーザー) が所属している OS グループです。
- ※ オプションである Oracle ASM オペレータ (ASM の OSOPER) グループには、Oracle Grid Infrastructure のインストール・ユーザーの所属に関わらず、すべてのノードに共通して存在する任意の OS グループを入力できます。

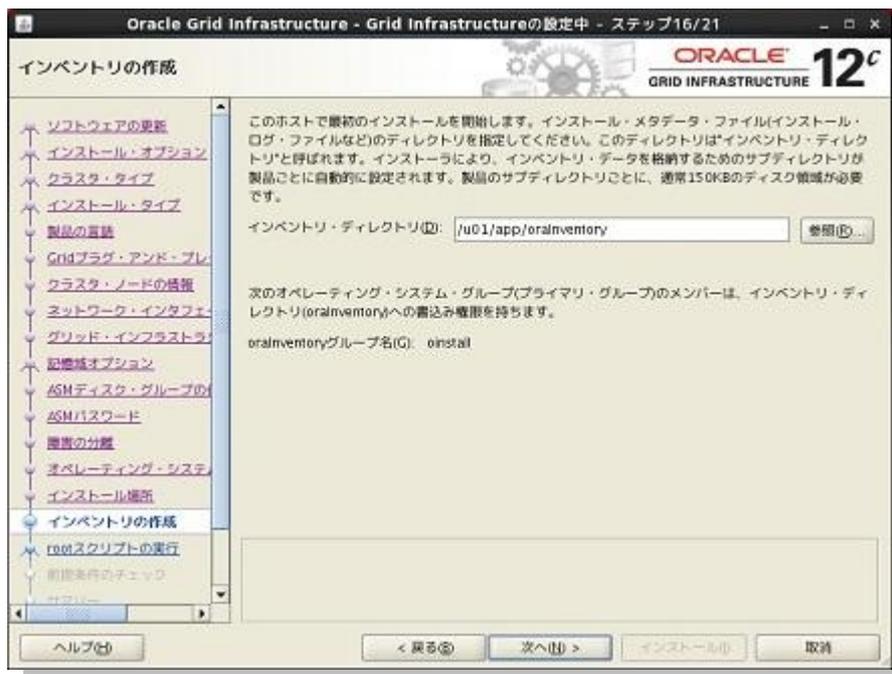
## 17. インストール場所の指定

Oracle ベースと Oracle Grid Infrastructure のホーム・ディレクトリとなるソフトウェアの場所を指定します。ここでは Oracle ベースが「/u01/app/grid」、ソフトウェアの場所が「/u01/app/12.1.0/grid」と入力して「次へ」をクリックします。



## 18. インベントリの作成

インベントリ・ディレクトリを設定します。ここでは「/u01/app/orainventory」が設定されていることを確認して「次へ」をクリックします。



## 19. root スクリプトの実行構成

ソフトウェアのインストールおよび構成中に、root ユーザーとして実行する必要があるスクリプト操作があります。root ユーザーのパスワードを入力するか、sudo を設定して使用することによって、手動ではなく OUI によりスクリプトを実行することができます。

ここでは root ユーザーのパスワードを入力しておき、すべてのノードでのスクリプト実行を OUI で行うものとしします。

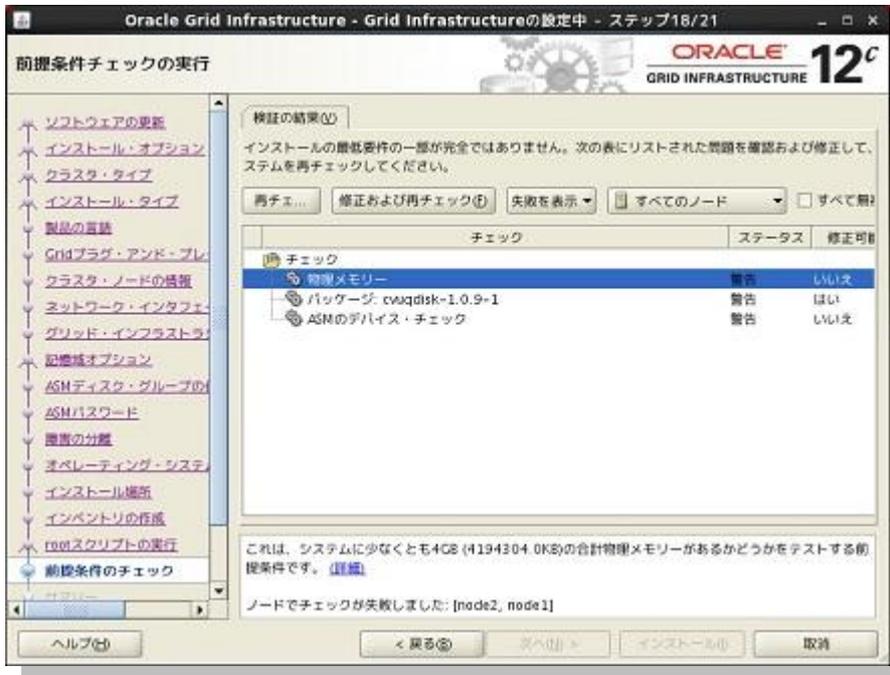
「構成スクリプトを自動的に実行」をチェック (☑) して、「root ユーザーの資格証明を使用」を選択してパスワードを入力します。入力後、「次へ」をクリックします。



## 20. 前提条件のチェック

インストール実行前に前提条件のチェックが実行されます。

すべての項目に対してチェックが成功した場合は自動的にサマリー画面に遷移します。いくつかの項目のチェックに失敗した場合は、以下の画面のように表示されますので適宜修正を実施します。



今回は cvuqdisk パッケージが未インストールであるため（「5.7 cvuqdisk パッケージのインストール」を参照）この項目に対するチェックが失敗しています。「修正および再チェック」をクリックして OUI によって生成されている修正スクリプトを確認し、パッケージのインストールを実施します。

表示されている修正スクリプト (runfixup.sh) を root ユーザーで実行します。スクリプトを実行後は「OK」をクリックして前提条件のチェックを再度実行します。



node1 における修正スクリプトの実行例を以下に記載します。修正スクリプトは、修正が必要なすべてのノードでそれぞれ実行します。今回は ndoe1 および node2 でそれぞれ実行します。

```
# /tmp/CVU_12.1.0.1.0_grid/runfixup.sh
```

<実行例>

```
[root@node1 ~]# /tmp/CVU_12.1.0.1.0_grid/runfixup.sh  
All Fix-up operations were completed successfully.
```

本ガイドでは検証環境用途での環境構築を想定しているため、メモリ要件に関する警告が表示されていますが、そのままインストールを実行します。「すべてを無視」にチェック (☑) をして「次へ」をクリックします。

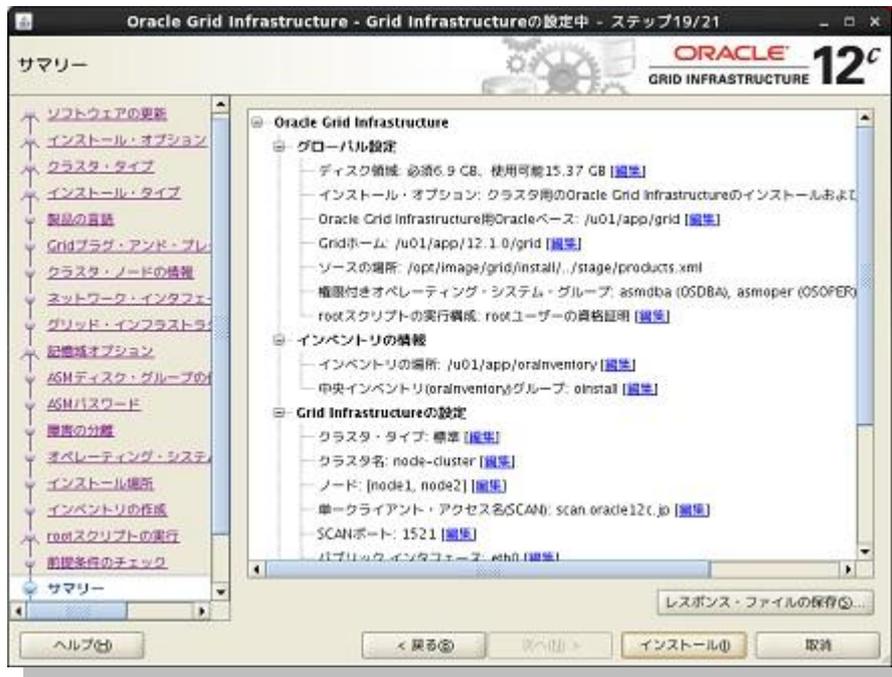


続行を確認するメッセージが表示されたら「はい」をクリックして続行します。



## 21. サマリー

サマリー画面の表示を確認の上、「インストール」をクリックしてインストールを開始します。

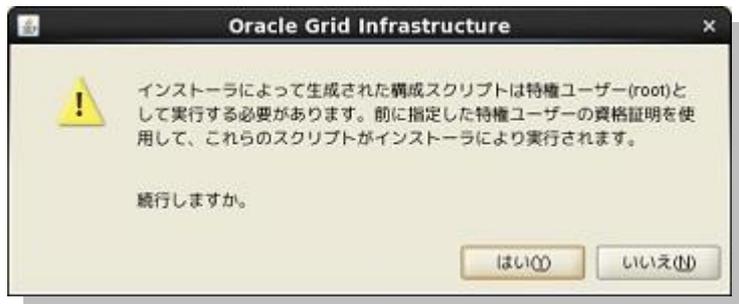


## 22. 製品のインストール

製品のインストールが実行されます。



インストールが進むと、OUI により構成スクリプト (orainstRoot.sh と root.sh) の実行について確認が表示されます。「はい」をクリックして続行します。



### 23. 終了

次の画面が表示されれば Oracle Grid Infrastructure のインストールおよび構成は完了です。「閉じる」をクリックして、OUI を終了します。



## 24. インストール後の確認

Oracle Grid Infrastructure インストール後の確認として、Oracle Clusterware のリソースの稼働状況を確認します。確認には、クラスタ制御ユーティリティである CRSCCTL コマンドを使用します。CRSCCTL コマンドは Oracle Grid Infrastructure をインストールしたユーザー (ここでは grid ユーザー) で実行します。

実行するコマンドと出力例を記載します。

```
$ /u01/app/12.1.0/grid/bin/crsctl status resource -t
```

<実行例>

```
[grid@node1 ~]$ /u01/app/12.1.0/grid/bin/crsctl status resource -t
-----
Name          Target State      Server      State details
-----
Local Resources
-----
ora.ASMNET1LSNR_ASM.lsnr
                ONLINE ONLINE      node1      STABLE
                ONLINE ONLINE      node2      STABLE
ora.ASMNET2LSNR_ASM.lsnr
                ONLINE ONLINE      node1      STABLE
                ONLINE ONLINE      node2      STABLE
ora.DATA.dg
                ONLINE ONLINE      node1      STABLE
                ONLINE ONLINE      node2      STABLE
ora.LISTENER.lsnr
                ONLINE ONLINE      node1      STABLE
                ONLINE ONLINE      node2      STABLE
ora.net1.network
                ONLINE ONLINE      node1      STABLE
                ONLINE ONLINE      node2      STABLE
ora.ons
                ONLINE ONLINE      node1      STABLE
                ONLINE ONLINE      node2      STABLE
ora.proxy_advm
                ONLINE ONLINE      node1      STABLE
                ONLINE ONLINE      node2      STABLE
```

<次ページへ続く>

&lt;前ページより続き&gt;

-----  
Cluster Resources  
-----

ora.LISTENER_SCAN1.lsnr	1	ONLINE	ONLINE	node2	STABLE
ora.LISTENER_SCAN2.lsnr	1	ONLINE	ONLINE	node1	STABLE
ora.LISTENER_SCAN3.lsnr	1	ONLINE	ONLINE	node1	STABLE
ora.MGMTLSNR	1	ONLINE	ONLINE	node1	169.254.119.2 192.168.100.101 192.168.200.101, STABLE
ora.asm	1	ONLINE	ONLINE	node1	STABLE
	2	ONLINE	ONLINE	node2	STABLE
	3	OFFLINE	OFFLINE		STABLE
ora.cvu	1	ONLINE	ONLINE	node1	STABLE
ora.mgmtdb	1	ONLINE	ONLINE	node1	Open, STABLE
ora.node1.vip	1	ONLINE	ONLINE	node1	STABLE
ora.node2.vip	1	ONLINE	ONLINE	node2	STABLE
ora.oc4j	1	ONLINE	ONLINE	node1	STABLE
ora.scan1.vip	1	ONLINE	ONLINE	node2	STABLE
ora.scan2.vip	1	ONLINE	ONLINE	node1	STABLE
ora.scan3.vip	1	ONLINE	ONLINE	node1	STABLE

- ※ 今回の環境では、ora.asm の 3 以外のすべてのリソースが ONLINE であることを確認します。
- ※ 今回の環境では、管理リポジトリ用のデータベースを構成していないため、リソース ora.mgmt.db および ora.MGMTLSNR は表示されません。
- ※ ADVM や ACFS を利用しない場合には、リソース ora.proxy\_advm を停止しても構いません。詳細は、「Appendix 3. Oracle ASM プロキシ・インスタンスの管理」を参照してください。

## 7. Oracle Database のインストールと RAC データベースの作成

ここでは、Oracle Database のインストールについて説明し、続いて Database Configuration Assistant (DBCA) を使用したデータベースの作成について説明します。

### 7.1. Oracle Database のインストール

#### 1. OUI の起動

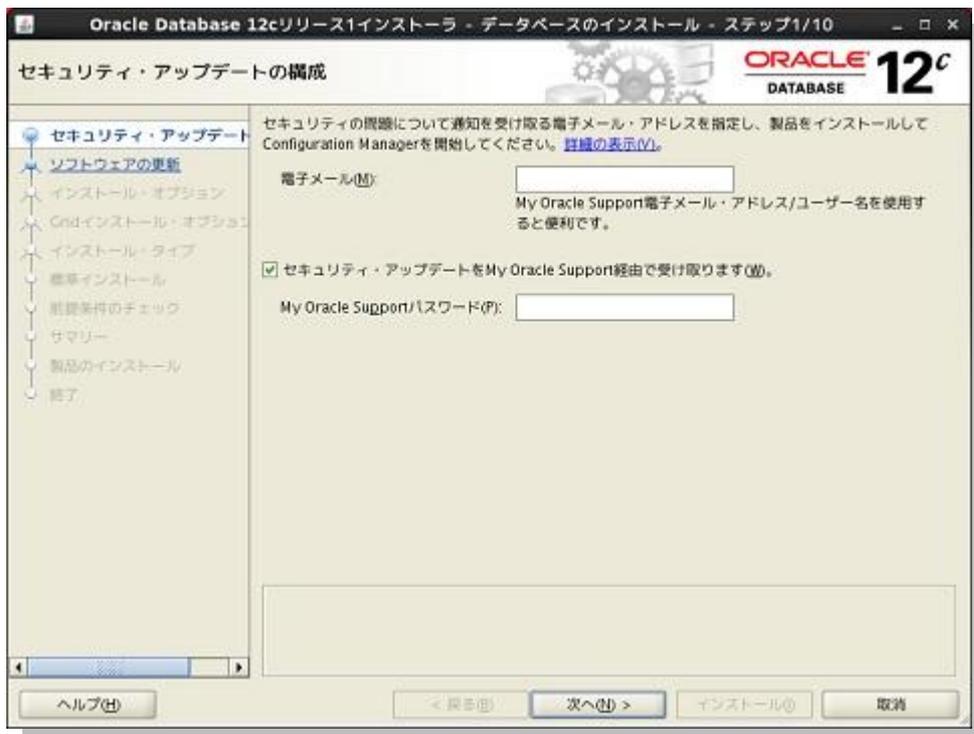
インストールを行う OS ユーザー (ここでは oracle ユーザー) で OUI を起動します。OUI を起動するため、Oracle VM VirtualBox 画面の端末から、新しい端末を開き oracle ユーザーで次のコマンドを実行してください。

```
$ /opt/image/database/runInstaller
```



#### 2. セキュリティ・アップデートの構成

セキュリティに関する更新を電子メールや My Oracle Support (MOS) 経由で受け取る設定ができます。ここでは、そのまま「次へ」をクリックします。

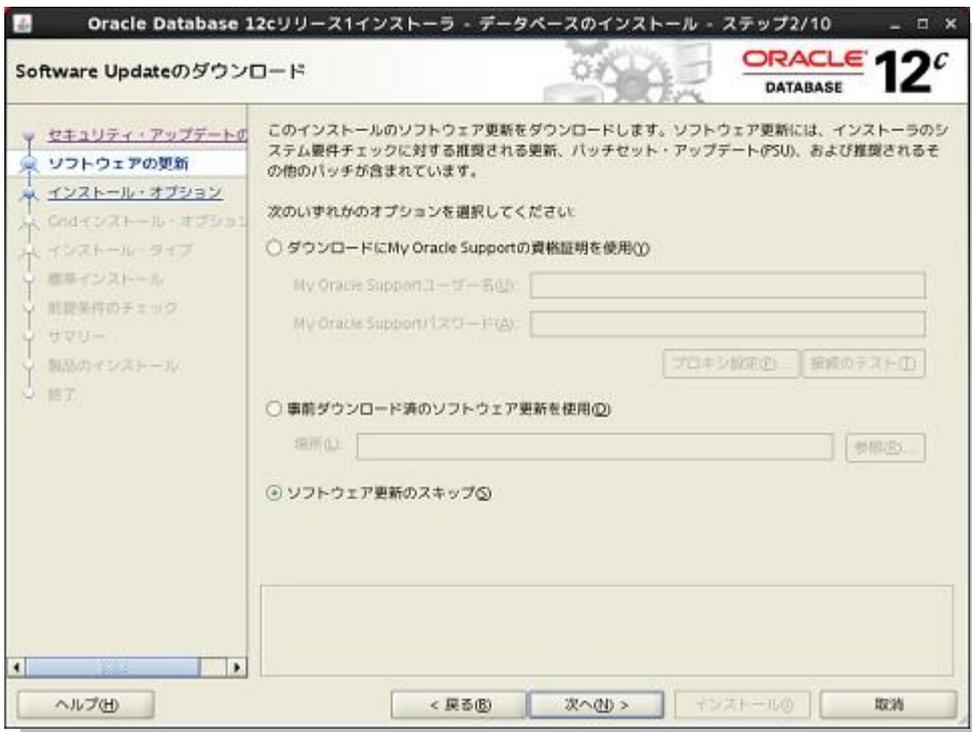


電子メール・アドレスの登録は任意なので、ここでは「はい」を選択してインストールを続けます。



### 3. Software Update のダウンロード

インストール中に最新のパッチなどの更新をダウンロードして適用するためのオプションとして、ソフトウェア更新のダウンロードオプションが提供されています。ここでは更新のダウンロードや適用は行わないものとしますので、「ソフトウェア更新のスキップ」を選択して「次へ」をクリックします。



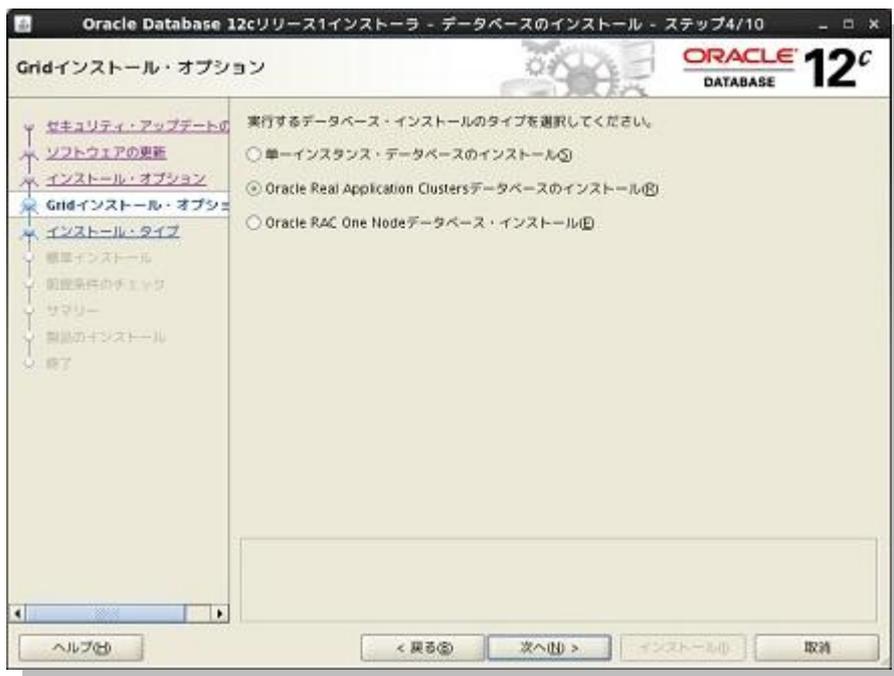
#### 4. インストール・オプションの選択

インストールのオプションを選択します。ここでは、データベースの構成はインストール後に DBCA を用いて実施するものとしますので「データベース・ソフトウェアのみインストール」を選択して、「次へ」をクリックします。



#### 5. Grid インストール・オプション

実行するインストールのタイプを選択します。「Oracle Real Application Clusters データベースのインストール」を選択して、「次へ」をクリックします。



## 6. ノード・リストの選択

インストールを行うノードを選択します。ここでは node1 および node2 がチェック (☑) されていることを確認して、「SSH 構成」をクリックします。SSH の構成に必要な情報を入力するフィールドが表示されるので、oracle ユーザーのパスワードを入力して「設定」をクリックします。



※ Oracle Database のインストールに、Oracle Grid Infrastructure のインストールと同じ OS ユーザーを使用している場合には、ここで改めて SSH 接続を構成する必要はありません。

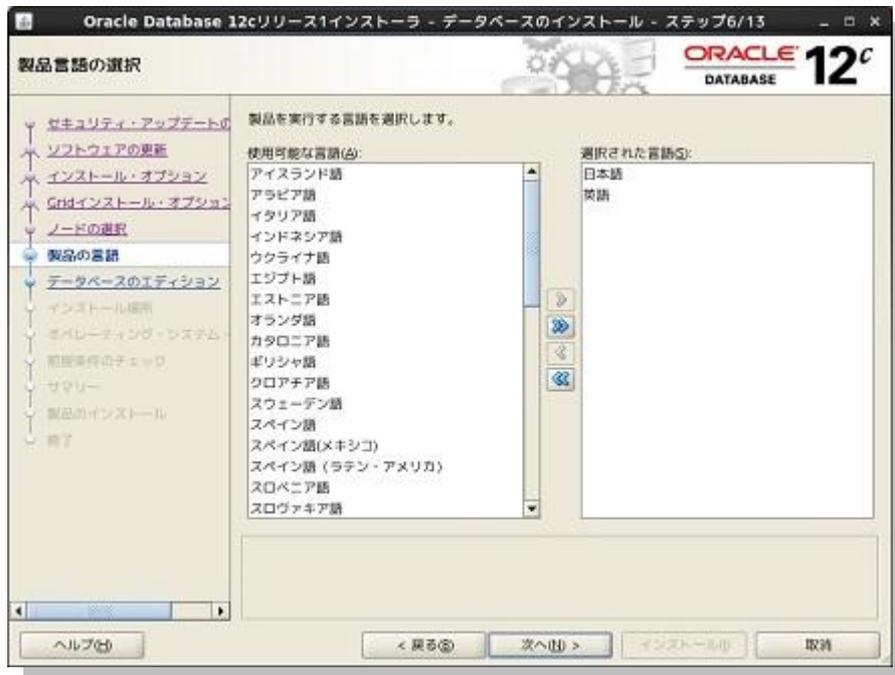
OUI により SSH 接続の確立が試行されます。次の画面が表示されたら「OK」をクリックします。



SSH 接続の確立が完了したので、「次へ」をクリックします。

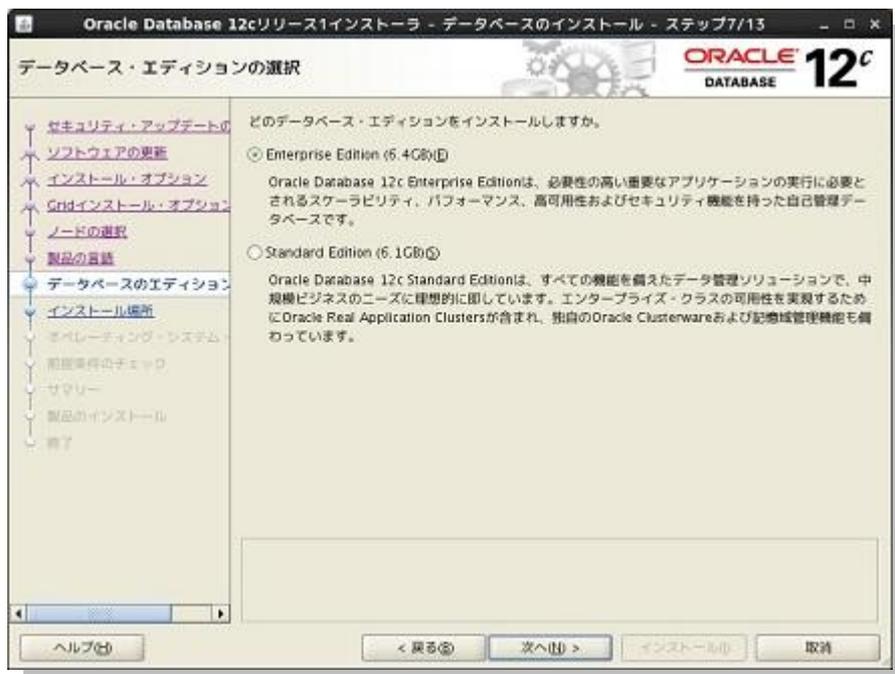
## 7. 製品言語の選択

製品を実行する言語を選択します。ここでは、製品を実行する言語として「日本語」と「英語」が選択されていることを確認して「次へ」をクリックします。



## 8. データベース・エディションの選択

インストールするソフトウェアのデータベース・エディションを選択します。ここでは「Enterprise Edition」を選択して「次へ」をクリックします。



## 9. インストール場所の指定

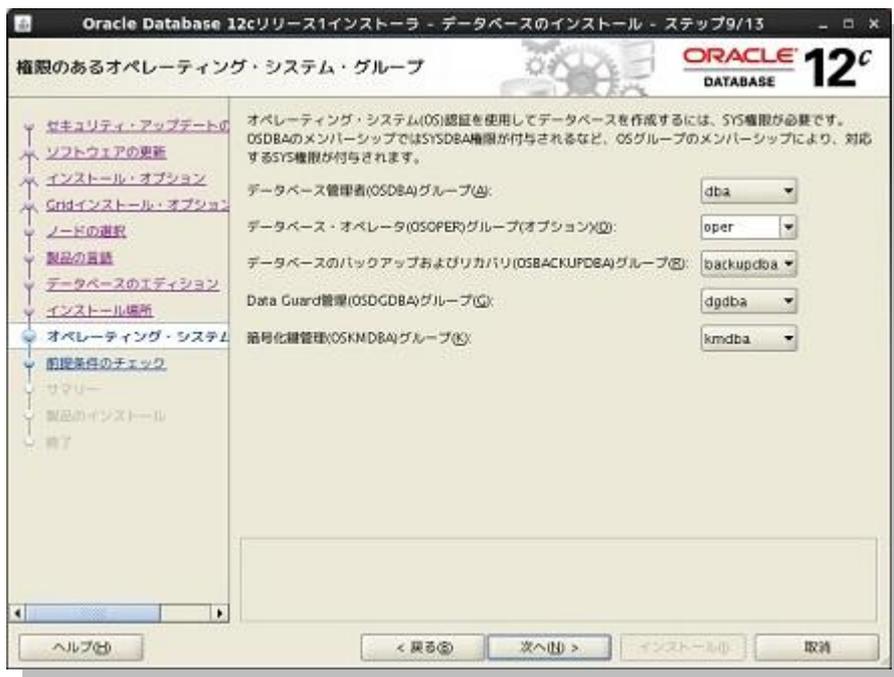
Oracle ベースと Oracle Database のホーム・ディレクトリとなるソフトウェアの場所を指定します。ここでは Oracle ベースが「/u01/app/oracle」、ソフトウェアの場所が「/u01/app/oracle/product/12.1.0/dbhome\_1」であることを確認して「次へ」をクリックします。



## 10. 権限付きオペレーティング・システム・グループ

データベースに対する OS 認証に使用する OS グループを設定します。ここではデフォルトの設定のまま、次のように設定するものとします。

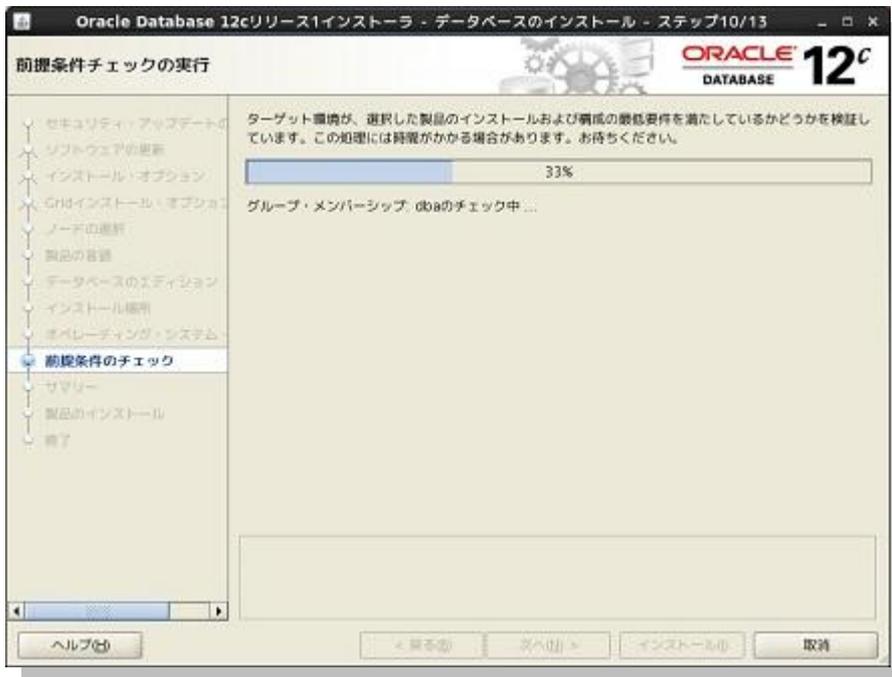
- データベース管理者 (OSDBA) グループに「dba」
- データベース・オペレータ (OSOPER) グループには「oper」
- データベースのバックアップおよびリカバリ (OSBACKUPDBA) グループには「backupdba」
- Data Guard 管理 (OSDBDBA) グループには「dgdba」
- 暗号化鍵管理 (OSKMDBA) グループには「kmdba」



※ データベース管理者 (OSDBA) グループなどとしてプルダウンより選択できる OS グループは Oracle Database のインストール・ユーザー (ここでは oracle ユーザー) が所属している OS グループです。オプションであるデータベース・オペレータ (OSOPER) グループには、Oracle Database のインストール・ユーザーの所属に関わらず、すべてのノードに共通して存在する任意の OS グループを入力できます。

## 11. 前提条件チェックの実行

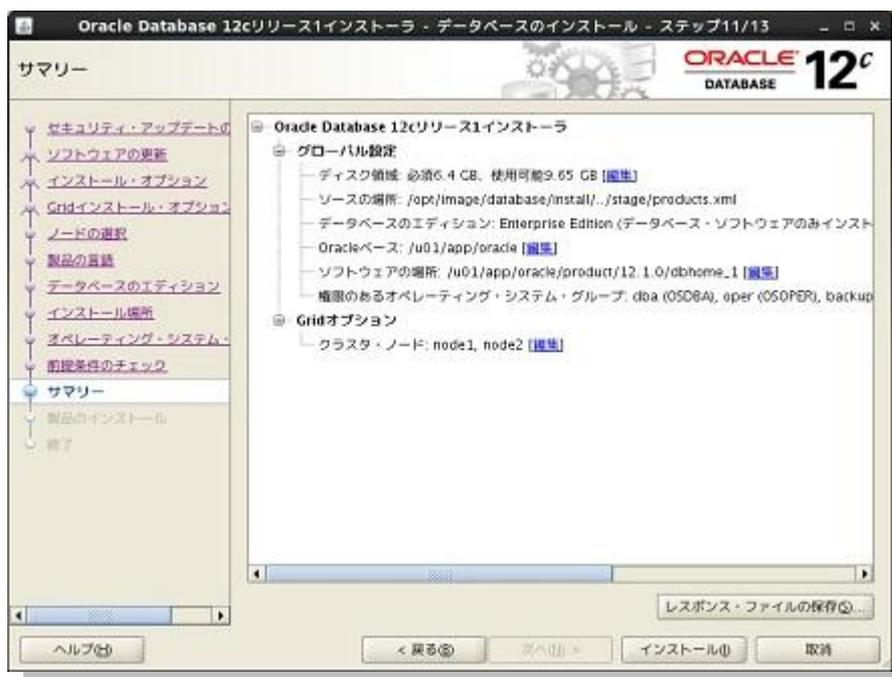
インストール実行前に前提条件のチェックが実行されます。



すべての項目に対してチェックが成功した場合は自動的にサマリー画面に遷移します。いくつかの項目のチェックに失敗した場合には、結果が表示されますので適宜修正を実施します。

## 12. サマリー

サマリー画面の表示を確認の上、「インストール」をクリックしてインストールを開始します。



## 13. 製品のインストール

製品のインストールが実行されます。

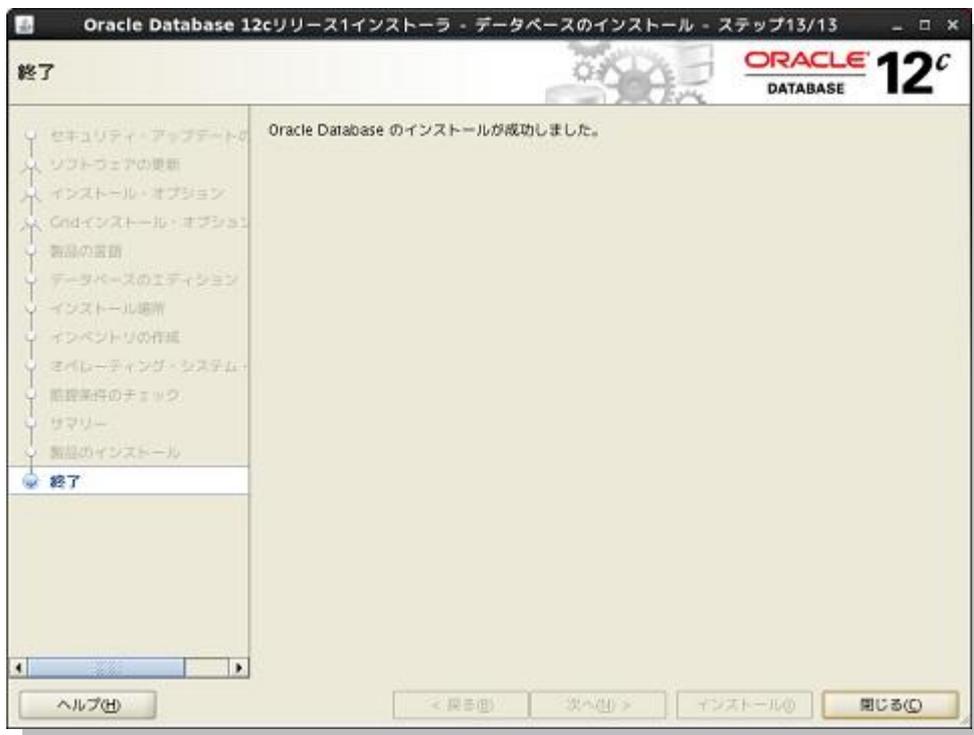


インストールが進むと、OUIにより構成スクリプト (root.sh) の実行が指示されます。すべてのノードにおいて root ユーザーで構成スクリプトを実行します。実行が完了したら、「OK」をクリックします。



## 14. 終了

次の画面が表示されれば Oracle Database のインストールは完了です。「閉じる」をクリックして、OUI を終了します。



## 7.2. ASMCA を利用した新規 ASM ディスク・グループの作成

RAC データベースの作成に必須ではありませんが、DBCA を利用した RAC データベース作成の前に、データベースの高速リカバリ領域 (Fast Recovery Area) 用の ASM ディスク・グループを新規に作成します。作成には Automatic Storage Management Configuration Assistant (ASMCA) を利用します。

### 1. ASMCA の起動

Oracle Grid Infrastructure のインストールを実行したユーザー (ここでは grid ユーザー) で、Oracle VM VirtualBox 画面上の端末から、次のコマンドを実行して ASMCA を起動します。

```
$ su - grid
```

```
$ /u01/app/12.1.0/grid/bin/asmca
```



### 2. 高速リカバリ領域用の ASM ディスク・グループの作成

「ディスク・グループ」タブで ASM ディスク・グループの一覧を確認し、「作成」をクリックします。



ASM ディスク・グループの新規作成に必要な情報を入力します。ここでは、ディスク・グループ名に「FRA」と入力し、冗長性は「通常」のまま作成するものとします。メンバー・ディスクの選択としてASM ディスク・グループの構成に使用するディスクを選択し、「OK」をクリックします。



ASM ディスク・グループの作成が実行されます。



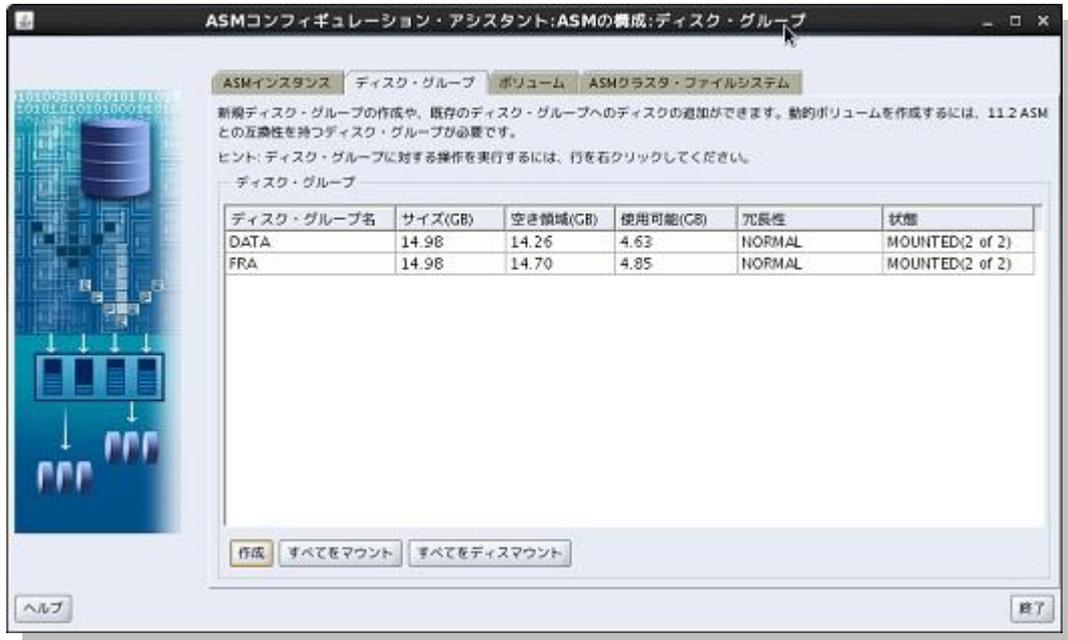
作成が正常に完了後、次のメッセージが表示されます。確認の上「OK」をクリックします。



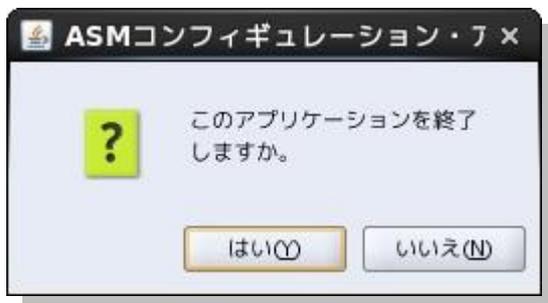
### 3. 作成後の確認

再度 ASM ディスク・グループの一覧を確認し、作成した ASM ディスク・グループがすべてのノードでマウントされていることを「状態」で確認します。

ここまでで ASMCA での操作は完了です。「終了」をクリックします。



終了を確認するメッセージが表示されたら「はい」をクリックします。



### 7.3. DBCA を利用した RAC データベースの作成

#### 1. DBCA の起動

Oracle Database のインストールを実行したユーザー（ここでは oracle ユーザー）で、Oracle VM VirtualBox 画面上の端末から、次のコマンドを実行して DBCA を起動します。

```
$ /u01/app/oracle/product/12.1.0/dbhome_1/bin/dbca
```



#### 2. データベース操作

実行するデータベースの操作を選択します。「データベースの作成」を選択して「次へ」をクリックします。



### 3. 作成モード

作成のモードを選択します。ここでは、「**拡張モード**」を選択して「次へ」をクリックします。



### 4. データベース・テンプレート

データベースのテンプレートを選択します。ここではデータベース・タイプに「**Oracle Real Application Clusters (RAC) データベース**」を、構成タイプに「**汎用またはトランザクション処理**」を選択して「次へ」をクリックします。



## 5. データベース識別情報

データベースの構成に必要な情報を入力します。ここでは、グローバル・データベース名に「orcl」と入力します。また、コンテナ・データベースとして作成を選択()して、「1 つ以上の PDB を含むコンテナ・データベースの作成」を選択します。PDB の数は「2」とし、名前接頭辞には「pdb」を入力します。入力後、「次へ」をクリックします。



※ 「コンテナ・データベースとして作成」を選択()すると、マルチテナント・アーキテクチャに対応した CDB としてデータベースを作成します。従来のアーキテクチャに対応した non-CDB としてデータベースを作成する場合には選択()せずに次の画面へ進みます。

※ CDB としてデータベースを作成する場合、PDB を併せて作成することができます。複数の PDB を作成することも可能ですが、その場合は PDB 名前接頭辞を指定します。作成される PDB には、PDB 名前接頭辞に数字を加えた名前が適用されます。

例：作成する PDB の数を「2」、PDB 名前接頭辞に「pdb」と入力した場合は、PDB として「pdb1」と「pdb2」が作成されます。

## 6. データベース配置

データベースを配置するサーバー・プールについて設定します。「このデータベースのサーバー・プールの新規作成」を選択して、サーバー・プール名に「srvpool1」を入力してカーディナリティに「2」を入力します。入力後、「次へ」をクリックします。



## 7. 管理オプション

データベースの管理オプションを選択します。ここでは「Enterprise Manager (EM) Database Express の構成」と「クラスタ検証ユーティリティ (CVU) ・チェックを定期的に行う」にチェック (☑) がされていることを確認して「次へ」をクリックします。



## 8. データベース資格証明

データベースの資格証明を設定します。ここでは「別の管理パスワードを使用」を選択し、ユーザーごとにパスワードを設定した後、「次へ」をクリックします。



## 9. 記憶域の場所

データベース・ファイルを格納する記憶域について設定をします。ここでは、記憶域のタイプに「**自動ストレージ管理 (ASM)**」が選択されていることを確認します。「**すべてのデータベース・ファイルに対して共通の位置を使用**」が選択されていること、およびデータベース・ファイルの位置に「**+DATA**」が設定されていることを確認の上、「**Oracle Managed Files の使用**」を設定します。

リカバリ関連ファイルについては、記憶域のタイプに「**自動ストレージ管理 (ASM)**」が選択されていることを確認します。また、「**高速リカバリ領域の指定**」が選択されていること、および高速リカバリ領域に「**+FRA**」が設定されていることを確認します。今回の構成ではサイズを「**4800**」MB に設定します。

アーカイブの有効化は任意ですが、ここでは「**アーカイブ有効化**」を選択()して「**次へ**」をクリックします。



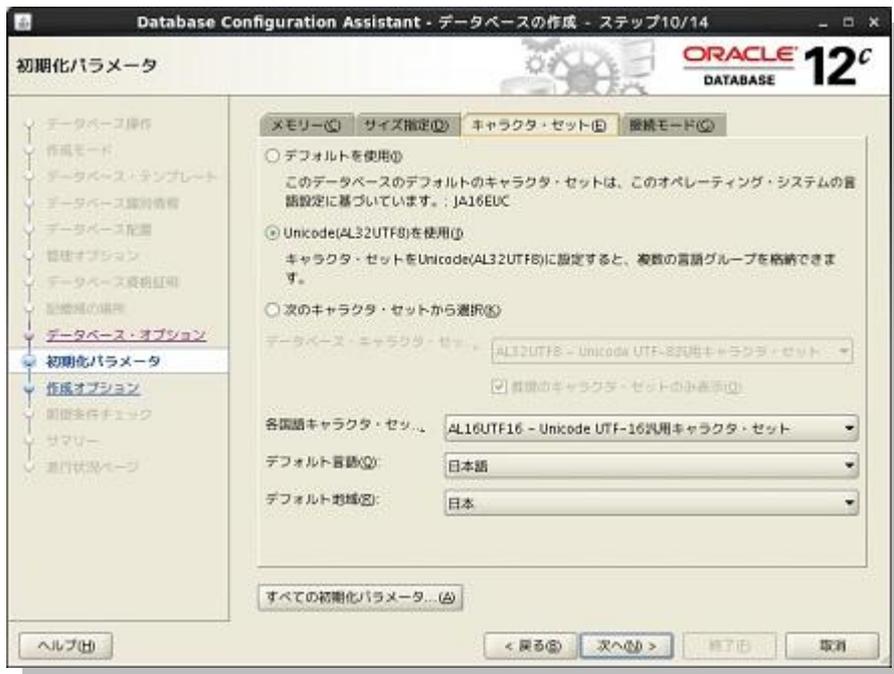
## 10. データベース・オプション

Database Vault と Label Security の設定を実施します。ここでは、特に設定変更はせずに、「次へ」をクリックします。



## 11. 初期化パラメータ

初期化パラメータに関する設定を実施します。ここでは、「キャラクタ・セット」タブをクリックして、データベース・キャラクタ・セットに「Unicode (AL32UTF8) を使用」を選択し、「次へ」をクリックします。



## 12. 作成オプション

データベースの作成オプションを選択します。ここでは、「データベースの作成」にチェック (☑) が付いていることを確認して「次へ」をクリックします。



データベース作成に関するサマリーが表示されます。内容を確認して「終了」をクリックします。



データベースの作成が実行されます。



データベースのユーザーに関してアカウントのアンロックを実施したい場合は「パスワード管理」より操作します。特に必要なければ「終了」をクリックしてデータベースの作成は完了です。



続いて「閉じる」をクリックして DBCA を終了します。

## 8. インストール後の確認と設定

最後にインストール後の確認および設定として、次の内容を実施します。

- 8.1 環境変数の設定
- 8.2 SCAN を使用した SQL\*Plus による接続
- 8.3 Oracle Enterprise Manager Database Express への接続

### 8.1 環境変数の設定

OS にログインした上で、Oracle Grid Infrastructure や Oracle Database に対して SRVCTL や CRSCTL といったユーティリティや SQL\*Plus を使用して管理操作を実行するため、環境変数を設定します。本ガイドでは、Oracle Grid Infrastructure と Oracle Database のインストールに異なる OS ユーザーを使用しているため、環境変数の設定はユーザーごとに実施します。(「5.11 環境変数とリソース制限の設定」を参照)

ここでは、環境変数の設定を永続的に行う方法として、ユーザーのプロファイル・ファイル内に設定を記述する例を紹介します。

- Oracle Grid Infrastructure 所有ユーザー (grid) 用の環境変数

<設定例>

```
[grid@node1 ~]# vi .bash_profile
# .bash_profile

# Get the aliases and functions
if [ -f ~/.bashrc ]; then
    . ~/.bashrc
fi

# User specific environment and startup programs

PATH=$PATH:$HOME/bin
export PATH

<以下を追記>
export TMPDIR=$HOME/tmp
export TEMP=$HOME/tmp
export ORACLE_BASE=/u01/app/grid
export ORACLE_HOME=/u01/app/12.1.0/grid
export PATH=$ORACLE_HOME/bin:$ORACLE_HOME/jdk/bin:${PATH}
export LD_LIBRARY_PATH=$ORACLE_HOME/lib
export NLS_LANG=JAPANESE_JAPAN.UTF8
export LANG=ja_JP.UTF-8
```

環境変数 ORACLE\_SID を設定する場合には、本ガイドでは次の記述を追加します。

export ORACLE\_SID=+ASM1 ← node1 では「+ASM1」、node2 では「+ASM2」と設定します。

Oracle Flex ASM 構成の場合は必ずしも上記のようなノード名と SID 名の組み合わせとは限りません。

- Oracle Database 所有ユーザー (oracle) 用の環境変数

<設定例>

```
[oracle@node1 ~]# vi /home/oracle/.bash_profile
# .bash_profile

# Get the aliases and functions
if [ -f ~/.bashrc ]; then
    . ~/.bashrc
fi

# User specific environment and startup programs

PATH=$PATH:$HOME/bin
export PATH

<以下を追記>
export TMPDIR=$HOME/tmp
export TEMP=$HOME/tmp
export ORACLE_BASE=/u01/app/oracle
export ORACLE_HOME=/u01/app/oracle/product/12.1.0/dbhome_1
export PATH=$ORACLE_HOME/bin:$ORACLE_HOME/jdk/bin:${PATH}
export LD_LIBRARY_PATH=$ORACLE_HOME/lib
export NLS_LANG=JAPANESE_JAPAN.UTF8
export LANG=ja_JP.UTF-8
```

本ガイドでは、環境変数 ORACLE\_SID についても次のように記述を追加します。

```
export ORACLE_SID=orcl_1 ← node1 では「orcl_1」、node2 では「orcl_2」と設定します。
```

ポリシーベース管理型の RAC データベースの場合は、必ずしも上記のようなノード名と SID 名の組み合わせとは限りません。

本ガイドのように、ポリシー管理型の RAC データベースを作成した場合には、ノードごとにアンダースコアを含む SID 名を設定します。管理者管理型の RAC データベースを作成した場合には SID 名にアンダースコアは含まれませんので、アンダースコアを含まず、次のように SID 名を設定します。

```
export ORACLE_SID=orcl1 ← node1 では「orcl1」、node2 では「orcl2」と設定します。
```

## 8.2 SCAN を使用した SQL\*Plus からの接続

SCAN を使用した RAC データベースへの接続方法には、設定ファイルである tnsnames.ora に記述して接続する方法と、簡易接続ネーミング・メソッド (EZCONNECT) を使用する方法があります。ここでは、簡易接続ネーミング・メソッドを使用した接続方法を紹介합니다。

SCAN を使用した SQL\*Plus からの接続には、oracle ユーザーで次のコマンドを実行します。

```
$ sqlplus <USERNAME>/<PASSWORD>@<SCAN_NAME>:<PORT>/<SERVICE_NAME>
```

<実行例>

```
[oracle@node1 ~]$ sqlplus system/Welcome1@scan.oracle12c.jp:1521/orcl
SQL*Plus: Release 12.1.0.1.0 Production on 月 9月 30 10:27:45 2013
Copyright (c) 1982, 2013, Oracle. All rights reserved.

Oracle Database 12c Enterprise Edition Release 12.1.0.1.0 - 64bit Production
With the Partitioning, Real Application Clusters, Automatic Storage
Management, OLAP,
Advanced Analytics and Real Application Testing options
に接続されました。
```

また、プラガブル・データベースの稼働状況を確認します。

接続しているインスタンスについて確認する場合は次のコマンドを続けて実行します。作成した PDB が READ WRITE モードでオープンしていることを確認します。

```
SQL> SELECT INSTANCE_NAME, STATUS FROM V$INSTANCE;
```

```
SQL> show pdbs
```

<実行例>

```
SQL> SELECT INSTANCE_NAME, STATUS FROM V$INSTANCE;

INSTANCE_NAME          STATUS
-----
orcl_1                 OPEN

SQL> show pdbs

  CON_ID  CON_NAME          OPEN MODE          RESTRICTED
-----
      2  PDB$SEED          READ ONLY          NO
      3  PDB1              READ WRITE         NO
      4  PDB2              READ WRITE         NO
```

また、すべてのインスタンスについて確認する場合は次のコマンドを実行します。

```
SQL> SELECT NAME, OPEN_MODE, INST_ID FROM GV$PDBS;
```

<実行例>

```
SQL> SELECT NAME, OPEN_MODE, INST_ID FROM GV$PDBS;
```

NAME	OPEN_MODE	INST_ID
PDB\$SEED	READ ONLY	1
PDB1	READ WRITE	1
PDB2	READ WRITE	1
PDB\$SEED	READ ONLY	2
PDB1	READ WRITE	2
PDB2	READ WRITE	2

### 8.3 Oracle Enterprise Manager Database Express への接続

Oracle Enterprise Manager Database Express (EM Express) を構成した場合には、次の URL で接続することができます。ホスト名の代わりに SCAN 名を使用してアクセスすることも可能です。

`https://<HOSTNAME_OR_IPAddress>:5500/em`

アクセスはホスト OS からでも、ゲスト OS からでもかまいません。画面の表示には Flash プラグインのインストールが必要です。EM Express を使用する OS へ事前に Flash プラグインのインストールを行います。

本ガイドの構成では、ゲスト OS 上からブラウザを起動して、EM Express にアクセスするものとします。

ブラウザを起動して、次の URL を使用してアクセスします。

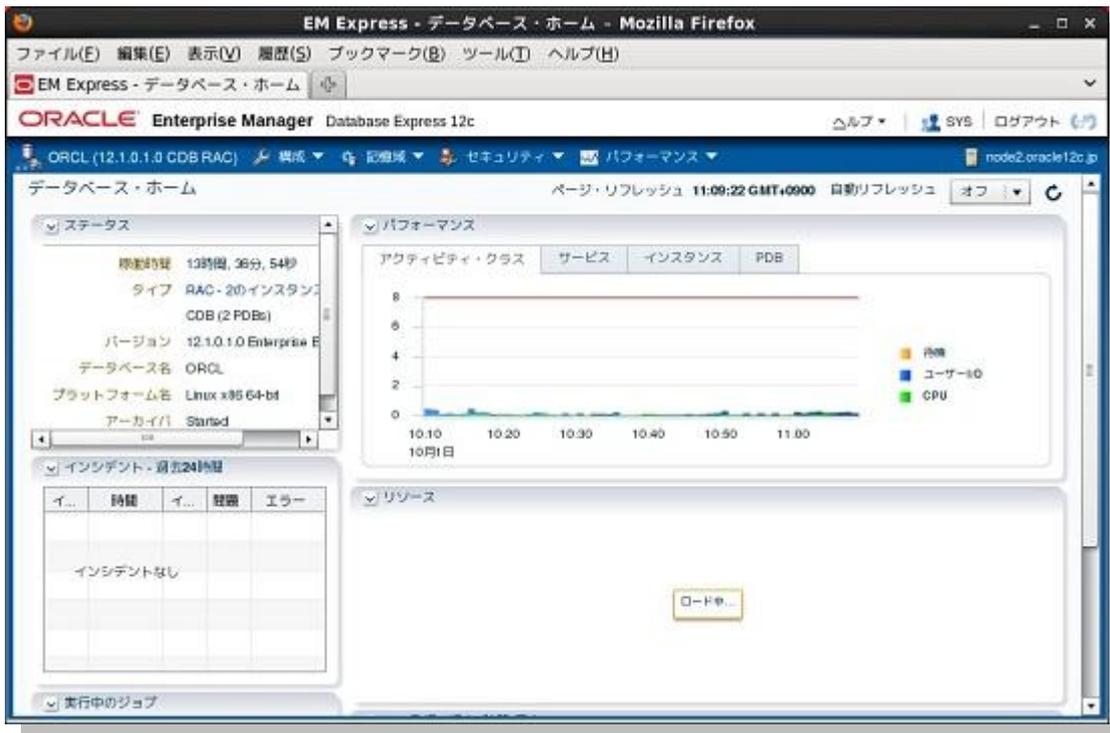
`https://scan.oracle12c.jp:5500/em`

EM Express のログイン画面が表示されたら、構成したデータベースに対するユーザーとパスワードを入力してログインします。ここでは、ユーザー名に「**sys**」、パスワードに Oracle Database インストール時に設定したユーザーのパスワードを入力して、接続モードに「**SYSDBA 権限**」を選択して「**ログイン**」をクリックします。



ログイン画面の表示に際し、セキュリティの警告が表示された場合には、セキュリティ例外を承認するか、セキュリティの証明書をインストールするなどの対処を実施します。

ログイン後の画面例は以下です。



※ EM Express については、既知の問題として次の点があります。

マルチテナント・コンテナ・データベースを使用している場合、EM Express ホームページ画面の「リソース」と「SQL 監視」のチャートをロードしている間に次のエラーが表示される場合があります。



サポート契約をお持ちの方は、この問題に対応するパッチの適用により問題の解決が可能です。

## Appendix 1. DNS サーバーの準備

DNS サーバーや、DNS サーバーとして利用できる他のサーバーがある場合は、環境構築に使用する仮想マシンや SCAN 用の名前解決を追加で設定して利用することができます。環境の構築に利用できる DNS サーバーがない場合は、仮想マシンを作成して DNS サーバーとして利用する方法があります。ここでは、後者の場合について準備の手順を紹介します。

前者の場合 (利用可能な DNS サーバーがある場合) には、既存の設定に SCAN 用の記述を追加します。

<記述例>

```
scan.oracle12c.jp IN A 192.168.56.201
                  IN A 192.168.56.202
                  IN A 192.168.56.203
```

通常、DNS サーバーの設定には DNS サーバーや名前解決のためのライブラリなどを提供する BIND を用いる方法がありますが、本ガイドでは検証環境用途での環境構築を想定しているため、より設定が容易な dnsmasq を用いて簡易的な DNS サーバーとして利用するものとします。

手順の説明には、仮想マシンの作成から dnsmasq を用いた設定までを含みます。また、本ガイドでの構成として Oracle Linux 6 を使用した場合を前提としています。

- Oracle VM VirtualBox コンソールを使用して、仮想マシンを作成します。

仮想マシンの作成については「3.4 仮想マシンの作成」を参照してください。今回は DNS サーバーとしての用途でのみ使用しますので、メモリには 512MB、ディスクは 8GB を割り当てて作成するものとします。作成後は、次の記述を参考に Oracle VM VirtualBox コンソール画面よりネットワークの設定を変更します。

- アダプター1：割り当てを「**ホストオンリーアダプター**」に変更します。

- 作成した仮想マシンを起動して Oracle Linux 6 をインストールします。

インストール方法は「4. Oracle Linux 6 のインストールと再起動後における設定」を参照してください。DNS サーバーとして利用するため Oracle Linux 6 のインストール時には、Oracle Validated RPM パッケージの追加インストールを選択する必要はありません。

また、本ガイドにおける仮想マシンのネットワーク設定を次に記述します。

- インターフェースの設定として eth0 には IP アドレスを「**192.168.56.254**」と設定するものとします。
- ホスト名には、「**dns.oracle12c.jp**」を設定します。

- Oracle Linux のインストールが完了したら、DNS サーバーを設定します。まずは、DNS サーバーの構築に必要なパッケージがインストールされているかを確認します。

次のパッケージがインストールされていることを確認してください。パッケージ名に加えて、括弧内には参考として Oracle Linux 6 でのパッケージのバージョンを記載します。

- dnsmasq (dnsmasq-2.48-13. el6.x86\_64.rpm)

確認は、次のコマンドで実施できます。

```
# rpm -qa dnsmasq
```

6. 名前解決のための記述を追記します。

/etc 配下の hosts ファイルに設定を記述します。dnsmasq を使用する場合は、host ファイルを利用した名前解決が可能です。

```
# cd /etc
```

```
# vi hosts
```

<実行例>

```
# cd /etc
# vi hosts

<以下を追記>
192.168.56.101      node1.oracle12c.jp  node1
192.168.56.102      node2.oracle12c.jp  node2
192.168.56.111      node1-vip.oracle12c.jp  node1-vip
192.168.56.112      node2-vip.oracle12c.jp  node2-vip
192.168.56.201      scan.oracle12c.jp   scan
192.168.56.202      scan.oracle12c.jp   scan
192.168.56.203      scan.oracle12c.jp   scan
192.168.56.254      dns.oracle12c.jp    dns
```

7. dnsmasq を起動します。

次のコマンドを実行して、サービスを起動します。

```
# /sbin/service dnsmasq start
```

<実行例>

```
# /sbin/service dnsmasq start
dnsmasq を起動中: [ OK ]
```

dnsmasq サービスが無効化されている場合は有効化します。

```
# chkconfig --list dnsmasq
```

```
# chkconfig dnsmasq on
```

## &lt;実行例&gt;

```
# chkconfig --list dnsmasq
dnsmasq      0:off  1:off  2:off  3:off  4:off  5:off  6:off

<無効化 (off) に設定されている場合には次のコマンドで有効化>
# chkconfig dnsmasq on

# chkconfig --list dnsmasq
dnsmasq      0:off  1:off  2:on   3:on   4:on   5:on   6:off
```

## 8. 確認

DNS サーバー以外のノードから、DNS サーバーによる名前解決ができることを確認します。次のコマンドを実行して、SCAN として利用するホスト名への問い合わせに対して、SCAN 用の IP アドレスが 3 つ返ることを確認します。ここでは、scan.oracle12c.jp の名前解決ができることを確認する例とします。

```
# nslookup scan.oracle12c.jp
```

## &lt;実行例&gt;

```
# nslookup scan.oracle12c.jp

Server:      192.168.56.254
Address:     192.168.56.254#53

Name:   scan.oracle12c.jp
Address: 192.168.56.201
Name:   scan.oracle12c.jp
Address: 192.168.56.202
Name:   scan.oracle12c.jp
Address: 192.168.56.203
```

## Appendix 2. CDB と PDB の基本操作

作成したデータベースでマルチテナント・コンテナ・データベース (CDB) およびプラグブル・データベース (PDB) の基本操作を確認する手順を記載します。ここでは PDB を作成し、接続する方法を紹介します。

### 1. リスナーおよびデータベースの起動確認

リスナーとデータベースが起動していることを確認します。

```
$ lsnrctl status
```

```
$ sqlplus / as sysdba
```

```
SQL> SELECT INSTANCE_NAME, STATUS FROM V$INSTANCE;
```

```
SQL> show pdbs
```

<実行例>

```
SQL> SELECT INSTANCE_NAME, STATUS FROM V$INSTANCE;

INSTANCE_NAME          STATUS
-----
orcl_1                 OPEN

SQL> show pdbs

   CON_ID CON_NAME                                OPEN MODE    RESTRICTED
-----
      2 PDB$SEED                                READ ONLY    NO
      3 PDB1                                  READ WRITE   NO
      4 PDB2                                  READ WRITE   NO
```

すべてのインスタンスについて確認する場合は次のコマンドを実行します。

```
SQL> SELECT NAME, OPEN_MODE, INST_ID FROM GV$PDBS;
```

<実行例>

```
SQL> SELECT NAME, OPEN_MODE, INST_ID FROM GV$PDBS;

NAME                OPEN_MODE          INST_ID
-----
PDB$SEED            READ ONLY          1
PDB1                 READ WRITE         1
PDB2                 READ WRITE         1
PDB$SEED            READ ONLY          2
PDB1                 READ WRITE         2
PDB2                 READ WRITE         2
```

## 2. PDB の新規作成

ここでは PDB3 を作成するものとします。次の構文を参考に、プラグブル・データベースを作成します。

```
SQL> CREATE PLUGGABLE DATABASE <NEW_PDB_NAME> ADMIN USER <USERNAME>  
IDENTIFIED BY <PASSWORD>;
```

<実行例>

```
SQL> CREATE PLUGGABLE DATABASE pdb3 ADMIN USER admin IDENTIFIED BY Welcome1;  
プラグブル・データベースが作成されました。  
SQL>
```

作成後の状態を確認しておきます。

```
SQL> SELECT NAME, OPEN_MODE, INST_ID FROM GV$PDBS;
```

<実行例>

```
SQL> SELECT NAME, OPEN_MODE, INST_ID FROM GV$PDBS;
```

NAME	OPEN_MODE	INST_ID
-----	-----	-----
PDB\$SEED	READ ONLY	1
PDB1	READ WRITE	1
PDB2	READ WRITE	1
PDB3	MOUNTED	1
PDB\$SEED	READ ONLY	2
PDB1	READ WRITE	2
PDB2	READ WRITE	2
PDB3	MOUNTED	2

作成した PDB をすべてのインスタンスでオープンします。

```
SQL> show pdbs
```

```
SQL> ALTER PLUGGABLE DATABASE <PDB_NAME> OPEN INSTANCES = ALL;
```

```
SQL> show pdbs
```

## &lt;実行例&gt;

```
SQL> show pdbs
```

CON_ID	CON_NAME	OPEN MODE	RESTRICTED
2	PDB\$SEED	READ ONLY	NO
3	PDB1	READ WRITE	NO
4	PDB2	READ WRITE	NO
5	PDB3	MOUNTED	

```
SQL> ALTER PLUGGABLE DATABASE pdb3 OPEN;
```

プラグブル・データベースが変更されました。

```
SQL> show pdbs
```

CON_ID	CON_NAME	OPEN MODE	RESTRICTED
2	PDB\$SEED	READ ONLY	NO
3	PDB1	READ WRITE	NO
4	PDB2	READ WRITE	NO
5	PDB3	READ WRITE	NO

すべてのインスタンスでの状態を確認しておきます。

```
SQL> SELECT NAME, OPEN_MODE, INST_ID FROM GV$PDBS;
```

## &lt;実行例&gt;

```
SQL> SELECT NAME, OPEN_MODE, INST_ID FROM GV$PDBS;
```

NAME	OPEN_MODE	INST_ID
PDB\$SEED	READ ONLY	1
PDB1	READ WRITE	1
PDB2	READ WRITE	1
PDB3	READ WRITE	1
PDB\$SEED	READ ONLY	2
PDB1	READ WRITE	2
PDB2	READ WRITE	2
PDB3	READ WRITE	2

### 3. ルートから PDB へ接続を切り替える

ルート (CDB\$ROOT) に接続している状態から、特定の PDB へ接続を切り替えます。接続の切り替えには SQL\*Plus で再接続する方法もありますがここでは ALTER SESSION 文を使用する方法を紹介します。

```
SQL> show con_name
```

```
SQL> ALTER SESSION SET CONTAINER = pdb3;
```

```
SQL> show user
```

```
SQL> show con_name
```

```
SQL> exit
```

<実行例>

```
SQL> show con_name

CON_NAME
-----
CDB$ROOT

SQL> ALTER SESSION SET CONTAINER = pdb3;

セッションが変更されました。

SQL> show user

ユーザーは"SYS"です。

SQL> show con_name

CON_NAME
-----
PDB3

SQL> exit
```

#### 4. PDB 用のサービスを作成

PDB に接続するためのサービスを作成します。PDB に接続するためのサービスは、PDB 名と同じ名前のサービスがデフォルトで作成されています。ここでは新規にサービスを作成する手順を記載します。

サービスの管理にはサーバー管理ユーティリティ (SRVCTL) の使用が推奨です。oracle ユーザーで次のコマンドを実行します。

```
$ srvctl add service -db <DB_NAME> -service <SERVICE_NAME> -serverpool  
  <SERVERPOOL_NAME> -pdb <PDB_NAME>;
```

```
$ srvctl start service -db <DB_NAME> -service <SERVICE_NAME>
```

<実行例>

```
$ srvctl add service -db orcl -service pdb3_srv -serverpool srvpool1 -pdb pdb3  
$ srvctl start service -db orcl -service pdb3_srv
```

作成後に接続を確認します。作成したサービスを使用して PDB へ接続することができます。

```
SQL> connect <USERNAME>/<PASSWORD>@<SCAN_NAME>:<PORT>/<SERVICE_NAME>
```

```
SQL> show user
```

```
SQL> show con_name
```

<実行例>

```
SQL> connect system/Welcome1@scan.oracle12c.jp:1521/pdb3_srv  
接続されました。  
  
SQL> show user  
ユーザーは"SYSTEM"です。  
  
SQL> show con_name  
  
CON_NAME  
-----  
PDB3
```

## Appendix 3. Oracle ASM プロキシ・インスタンスの管理

Oracle ASM プロキシ・インスタンスは、Oracle Flex ASM を使用する環境でも ACFS や ADVM 機能を使用するための新しいタイプのインスタンスです。

ACFS / ADVM カーネル・モジュールに対して ASM サービスを提供しますがメモリ使用量として 1GB 程度が必要になるため、今回のような検証環境においては ACFS / ADVM を使用しない場合、停止することも可能です。

Oracle ASM プロキシ・インスタンスは Oracle Clusterware にリソースとして登録、管理されます。リソース ora.proxy\_advm を停止する場合には、Oracle Grid Infrastructure をインストールした OS ユーザーで次の手順を実行します。

### 1. リソースの稼働状況の確認

```
$ crsctl status resource ora.proxy_advm -t
```

<実行例>

```
[grid@node1 ~]$ crsctl status resource ora.proxy_advm -t
-----
Name          Target State          Server          State details
-----
Local Resources
-----
ora.proxy_advm
              ONLINE ONLINE         node1           STABLE
              ONLINE ONLINE         node2           STABLE
-----
```

### 2. リソースの停止

```
$ srvctl stop asm -proxy
```

<実行例>

```
[grid@node1 ~]$ srvctl stop asm -proxy
```

念のため、実行後に再度リソースの稼働状況を確認して OFFLINE になっていることを確認します。

```
$ crsctl status resource ora.proxy_advm -t
```



日本オラクル株式会社

〒107-0061

東京都港区北青山 2-5-8

オラクル青山センター

### 無断転載を禁ず

このドキュメントは単に情報として提供され、内容は予告なしに変更される場合があります。このドキュメントに誤りが無いことの保証や、商品性又は特定目的への適合性の黙示的な保証や条件を含め明示的又は黙示的な保証や条件は一切無いものとします。日本オラクル株式会社は、このドキュメントについていかなる責任も負いません。また、このドキュメントによって直接又は間接にいかなる契約上の義務も負うものではありません。このドキュメントを形式、手段(電子的又は機械的)、目的に関係なく、日本オラクル株式会社の書面による事前の承諾なく、複製又は転載することはできません。

Oracle と Java は、Oracle Corporation 及びその子会社、関連会社の米国及びその他の国における登録商標です。文中の社名、商品名等は各社の商標または登録商標である場合があります。

本資料に記載されているシステム名、製品名等には、必ずしも商品表示((R)、TM)を付記していません。



Oracle is committed to developing practices and products that help protect the environment