opentext™

Community Edition 日本語ガイド

Vertica Analytics Platform

1 はじめに

1.1 本書の目的

本書は、Vertica Analytics Platform（以降、Vertica）の Community Edition を使い、データ分析を実施するためのテスト環境を Enterprise Mode で構築することを目的としております。

Community Edition とは、3 ノードクラスター、データサイズ 1TB まで無料でお試しいただけるエディションとなります。3 ノードクラスター、データサイズ 1TB 以上を使用したテストを実施されたい場合は、弊社問い合わせ先（MFI-jpn_vertica_info@opentext.com）までメールでお問い合わせください。

1.2 記載範囲

本書の記載範囲は、以下の通りです。

- Vertica Community Edition のインストール
- データベースの作成と削除
- データベースの起動と停止
- データベースオブジェクトの作成
- データロード
- データベースデザイナーの実行
- マネージメントコンソール（運用ツール）のインストール

これらの内容は、以下の英語版マニュアルから特に重要な部分を抜粋し、補足情報を加えた上で日本語化したものです。より詳細な情報については、マニュアルをご参照ください。

- Vertica Analytics Platform Version 23.3.x Documentation https://docs.vertica.com/23.3.x/en/
    - Supported Platforms
    - Set up Vertica on-premises
    - Getting Started

1.3 前提

本書では、以下の環境に Vertica Community Edition をインストールすることを前提としております。

- プラットフォーム：x86_64 アーキテクチャマシン 3 台（物理サーバー）
- オペレーティングシステム：Red Hat Enterprise Linux 8.8 64bit
- Vertica Community Edition のバージョン：23.3.0-0

インストール環境は、Red Hat Enterprise Linux 8.8 がインストールされている 3 ノードへのインストールを想定しております。1 ノードや 2 ノードへのインストールも可能です。Red Hat Enterprise Linux から派生するディストリビューションを利用される場合は同様の手順でインストールは可能と考えられますが、弊社では Red Hat Enterprise Linux および CentOS（CentOS は 8.5 まで）を除いて動作確認を行っておりませんのでご注意ください。また、Red Hat Enterprise Linux およびその派生ディストリビューション以外を利用される場合は、一部手順が異なる場合があります。前述の英語版マニュアルを必ずご確認ください。

マネージメントコンソールは前述とは別のサーバーにインストールすることとします。

本書の内容は、予告なく変更される可能性がある点、ご了承のうえご利用ください。

2 Vertica のインストール

2.1 インストールを始める前に

Vertica のインストールを始める前に、使用する環境について次の内容を確認してください。

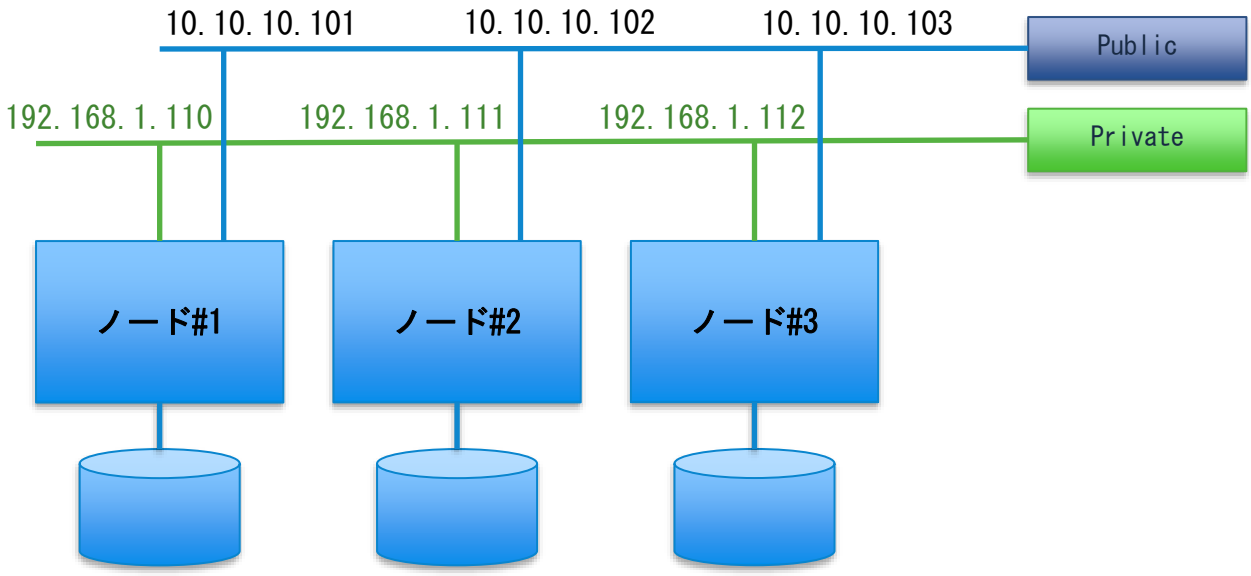
2.1.1 ネットワーク要件

Vertica は、Private と Public の 2 系統のネットワークを使用することを推奨します。

- Private: Vertica のノード内通信用。ネットワーク転送遅延が発生するとパフォーマンスの低下、または Vertica ノードが停止することがあります。
- Public: 外部アプリケーション（BI ツール等）との通信用。

Private は 10 ギガビット・イーサネット以上を推奨します。

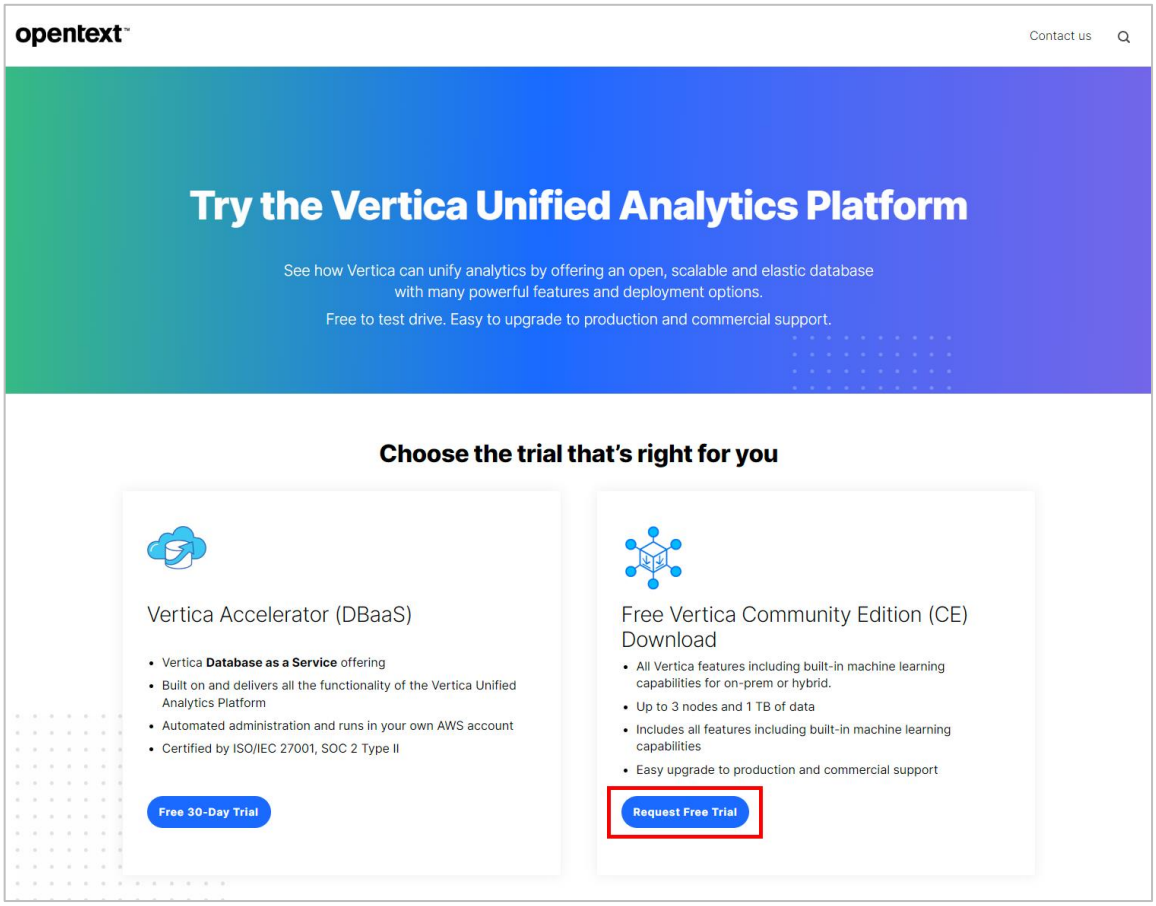
2.1.2 マシン構成例



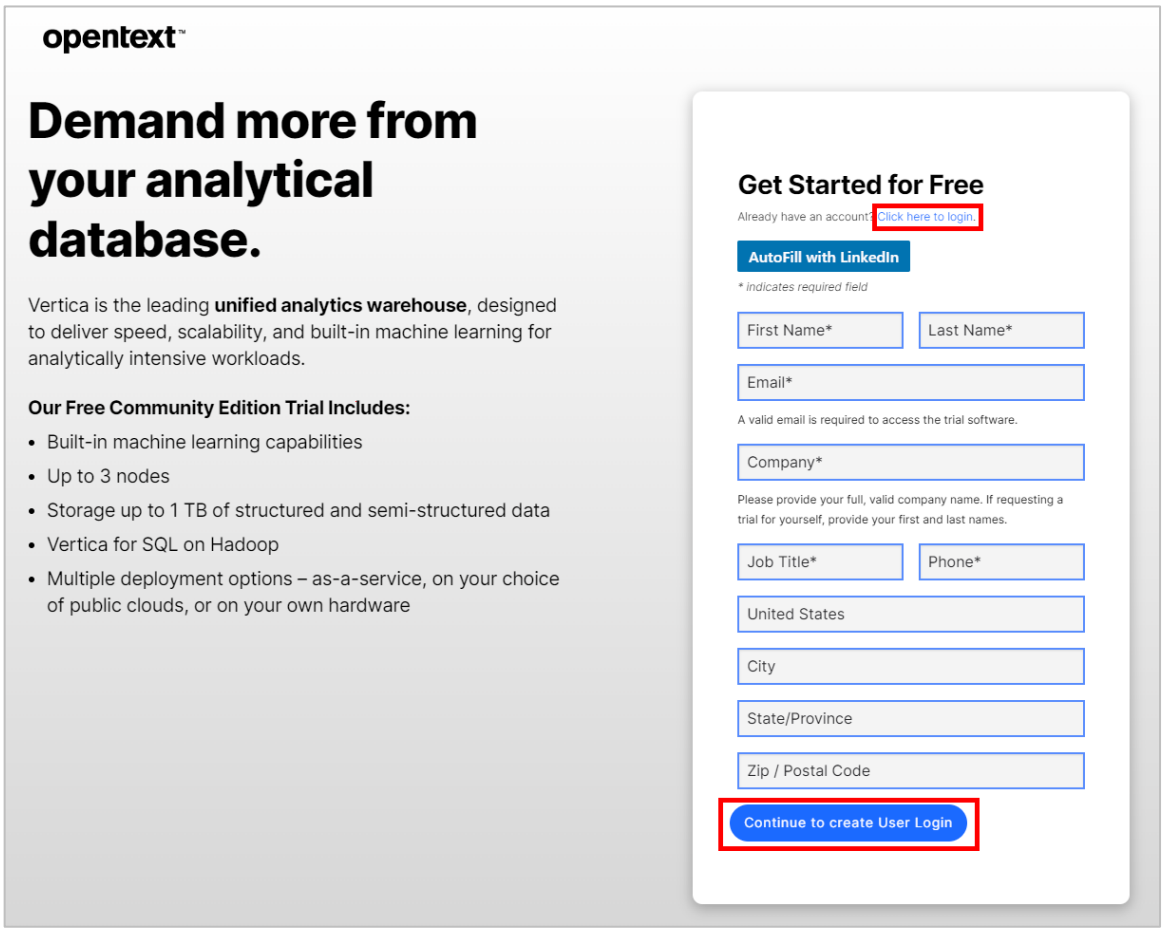
2.2 インストーラーファイルのダウンロードの実施

下記の手順に従って、Vertica のインストーラーファイルなどをダウンロードします。

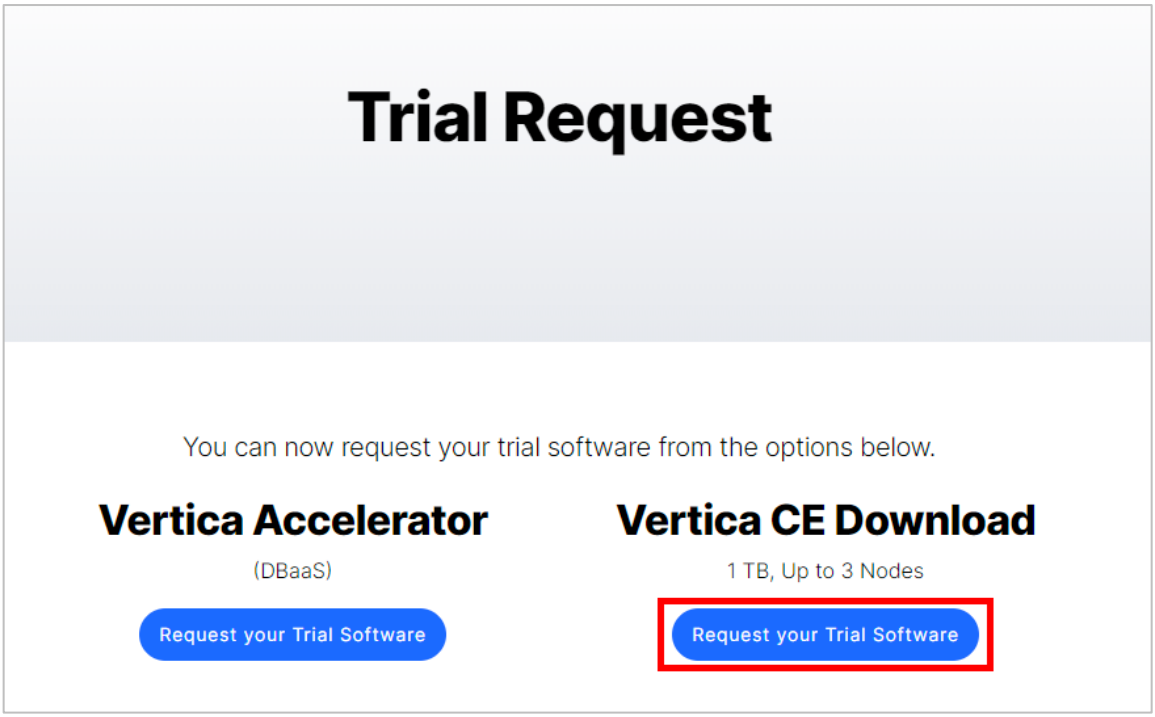
1. Web ブラウザを起動し、https://www.vertica.com/try/ へ接続します。
   「Free Vertica Community Edition (CE) Download」の「Request Free Trial」を選択します。



2. 必要な情報を入力して「Continue to create User Login」を選択し、その後の指示に従って Micro Focus Portal のアカウントを作成します。
既にアカウントを作成されている場合は「Click here to login」を選択してログインしてください。



3. アカウントにログイン後、次の画面で「Vertica CE Download」の「Request your Trial Software」を選択します。
弊社担当者が内容を確認後、ダウンロードを行うサイトの案内をメールにてお知らせいたします。確認が完了するまで数日を要することがありますのでご了承ください。



4. 案内に従いソフトウェアダウンロードページを表示します。「アカウントの検索」を押し、表示されるアカウントを選択します。

その後、「ダウンロード」を選択します。

5. ソフトウェアとバージョンの選択に移ります。以下の内容を選択します。

   製品：Vertica Community Edition
   製品名：Vertica Community Edition Evaluation Sub SW E-LTU
   バージョン：23.3

6. ソフトウェアファイルの選択に移ります。
   次のファイルをダウンロードします。

   vertica-23.3.0-0.x86_64.RHEL6.rpm（Vertica インストーラー）
   vertica-console-23.3.0-0.x86_64.RHEL6.rpm（Management Console インストーラー）

2.3 OS 設定の実施

2.3.1 OS をインストールする前のチェック項目

項目	チェック内容
メモリ	1 論理プロセッサあたり 1GB 以上（8GB 以上を推奨）
SWAP 領域	2GB 以上
ファイルシステム	ext4, ext3, XFS のいずれか ※パフォーマンスの観点から、ext4 または XFS を推奨
LVM	LVM を使用する場合は以下が条件となります。 • LVM のバージョン：2.02.66 以降 • device-mapper のバージョン：1.02.48 以降 ※論理ボリュームを lvcreate コマンドで作成する際、readahead オプションを使用し、Read ahead sector count を 2048KB 以上に設定します。
ディスクブロックサイズ	4,096 Bytes
ディスクスペース	Vertica のカタログとデータ用に使うディスクスペースを用意します。カタログは、Vertica のテーブル情報などのメタ情報が格納されます。OS 領域に配置することもできます。データは、Vertica に格納される実データです。データ領域は OS 領域とは別に配置されるのが一般的です。 データ用のディスクスペースはディスクスペース総容量の 60%以下になるように確保します。（K-Safe=1 を想定） アンチウィルスソフトウェアを導入されている場合、カタログとデータ用のディスクスペース、および Vertica のインストールディレクトリをアンチウィルスソフトウェアのスキャン対象外とします。
シェル	Vertica で使用するシェルスクリプトはすべて BASH で動作することを想定しています。root ユーザおよび Vertica で使用する OS ユーザのデフォルトのシェルは BASH を指定します。

2.3.2 OS インストール後の設定

OS インストール後に各サーバー上で次の設定を実施します。

2.3.2.1 パッケージのインストール

次のパッケージをインストールします。すでにインストールされている場合は最新のバージョンにアップデートします。

- bc chrony cronie dialog gdb libnsl mcelog openssh perl sysstat tzdata which

2.3.2.2 設定ファイルの有効化

本セクションにて OS の設定を行います。設定のいくつかは/etc/rc.local に行うことで OS の再起動時に再設定を行うことを避けることができます。

1. root ユーザーもしくは sudo で次のコマンドを実行し、/etc/rc.local を有効化します。

```
# chmod +x /etc/rc.d/rc.local
# systemctl start rc-local
```

2.3.2.3 Firewall の無効化

1. root ユーザーもしくは sudo で、firewalld を無効化します。

```
# systemctl mask firewalld
# systemctl disable firewalld
# systemctl stop firewalld
```

2.3.2.4 チューニングサービスの無効化

1. root ユーザーもしくは sudo で次のコマンドを実行し、tuned を無効化します。

```
# systemctl tuned stop
# systemctl disable tuned
```

2.3.2.5 ディスクの I/O スケジューラの変更

データ・カタログ用に使用するディスクに対して実施します。

ディスクが SSD の場合「none」に変更します。SSD 以外、または「none」の設定が適していない構成の場合は個々のディスク I/O スケジューラを試用後、設定を確定してください。Vertica ではディスク I/O 性能の測定を行うツールとして vioperf を提供しています。ご活用ください。

https://docs.vertica.com/23.3.x/en/setup/set-up-on-premises/install-using-command-line/validation-scripts/vioperf/

1. ディスクの I/O スケジューラの設定を確認します。

```
# cat /sys/block/sda/queue/scheduler
[none] mq-deadline kyber bfq

# cat /sys/block/sdb/queue/scheduler
[none] mq-deadline kyber bfq
```

   上記は、カタログと OS が「/dev/sda」、データが「/dev/sdb」に格納される場合の例です。「df -h」等で確認し、環境に合わせて変更してください。

2. 手順 1 で、「none」となっていない場合、root ユーザーもしくは sudo で、設定を変更します。

```
# echo none > /sys/block/sda/queue/scheduler
# echo 'echo none > /sys/block/sda/queue/scheduler' >> /etc/rc.local

# echo none > /sys/block/sdb/queue/scheduler
# echo 'echo none > /sys/block/sdb/queue/scheduler' >> /etc/rc.local
```

3. 手順 2 で設定変更した場合、ディスクの I/O スケジューラの設定を確認します。

```
# cat /sys/block/sda/queue/scheduler
[none] mq-deadline kyber bfq

# cat /sys/block/sdb/queue/scheduler
[none] mq-deadline kyber bfq
```

2.3.2.6 ディスクの Readahead の変更

データ・カタログ用に使用するディスクに対して実施します。

1. root ユーザーもしくは sudo で、Readahead の設定を確認します。

```
# /sbin/blockdev --getra /dev/sda
256

# /sbin/blockdev --getra /dev/sdb
256
```

上記は、カタログと OS が「/dev/sda」、データが「/dev/sdb」に格納される場合の例です。「df -h」等で確認し、環境に合わせて変更してください。

2. 手順 1 で「2048 未満」の値が確認できた場合、root ユーザーもしくは sudo で、設定を変更します。最大値は 8192 です。本手順でも必要に応じてディスク I/O 性能の測定ツール vioperf をご利用ください。

```
# /sbin/blockdev --setra 2048 /dev/sda
# echo '/sbin/blockdev --setra 2048 /dev/sda' >> /etc/rc.local

# /sbin/blockdev --setra 8192 /dev/sdb
# echo '/sbin/blockdev --setra 8192 /dev/sdb' >> /etc/rc.local
```

3. 手順 2 で設定変更した場合、root ユーザーもしくは sudo で、Readahead の設定を確認します。

```
# /sbin/blockdev --getra /dev/sda
2048

# /sbin/blockdev --getra /dev/sdb
8192
```

2.3.2.7 Transparent Hugepages の無効化

1. Transparent Hugepages の設定を確認します。

```
# cat /sys/kernel/mm/transparent_hugepage/enabled
[always] madvise never
```

2. 手順 1 で確認した設定が「never」となっていない場合、root ユーザーもしくは sudo で、設定を変更します。

```
# echo never > /sys/kernel/mm/transparent_hugepage/enabled
# echo 'echo never > /sys/kernel/mm/transparent_hugepage/enabled' >>
/etc/rc.local
```

3. 手順 2 で設定変更した場合、Transparent Hugepages の設定を確認します。

```
# cat /sys/kernel/mm/transparent_hugepage/enabled
always madvise [never]
```

なお、使用用途によっては Transparent Hugepages を有効化した方が Vertica のパフォーマンスが良い場合もあります。詳しくは製品ドキュメントを参照してください。

https://docs.vertica.com/23.3.x/en/setup/set-up-on-premises/before-you-install/manually-configured-os-settings/enabling-or-disabling-transparent-hugepages/

2.3.2.8 swappiness の値の変更

カーネルパラメータの swappiness の値を変更します。

1. root ユーザーもしくは sudo で、swappiness の値を確認します。

```
# cat /proc/sys/vm/swappiness
60
```

2. 手順 1 で「0」以外の値が確認できた場合、root ユーザーもしくは sudo で、/etc/sysctl.conf にパラメータを追記します。

```
# vi /etc/sysctl.conf
vm.swappiness=0
```

2.3.2.9 Dirty Data パラメータの変更

カーネルパラメータの dirty_ratio の値を変更します。設定値の推奨は以下のサイトを参照してください。

Tuning Linux Dirty Data Parameters for Vertica

https://www.vertica.com/kb/Tuning-Linux-Dirty-Data-Parameters-for-Vertica/Content/BestPractices/Tuning-Linux-Dirty-Data-Parameters-for-Vertica.htm

1. root ユーザーもしくは sudo で、/etc/sysctl.conf にパラメータを追記します。

```
# vi /etc/sysctl.conf
vm.dirty_ratio=5
vm.dirty_background_ratio=2
kernel.hung_task_panic=0
```

2.3.2.10 Network パラメータの変更

Network 関連のカーネルパラメータの値を変更します。

1. root ユーザーもしくは sudo で、/etc/sysctl.conf にパラメータを追記します。

```
# vi /etc/sysctl.conf
net.core.netdev_max_backlog=100000
net.core.rmem_default=262144
net.core.rmem_max=16777216
net.core.somaxconn=1024
net.core.wmem_default=262144
net.core.wmem_max=16777216
net.ipv4.tcp_mem=16777216 16777216 16777216
net.ipv4.tcp_rmem=8192 262144 8388608
net.ipv4.tcp_wmem=8192 262144 8388608
net.ipv4.udp_mem=16777216 16777216 16777216
net.ipv4.udp_rmem_min=16384
net.ipv4.udp_wmem_min=16384
```

2.3.2.11 ネットワークインターフェースカードの MTU 値およびオフロード値の設定

Private および Public のネットワークで使用しているネットワークインターフェースカードの MTU（Maximum Transmission Unit）の値およびオフロードの使用設定を必要に応じて見直します。

2.3.2.12 SELinux 機能の無効化

1. SELinux の設定を確認します。

```
# getenforce
Enforcing
```

2. 手順 1 で「Disabled」となっていない場合、/etc/selinux/config を編集し、SELinux を無効（SELINUX=disabled）します。

```
# setenforce 0
# sed -i 's/SELINUX=enforcing/SELINUX=disabled/' /etc/selinux/config
# grep 'SELINUX=' /etc/selinux/config
SELINUX=disabled
```

3. 手順 2 で設定変更した場合、任意のユーザーで、SELinux の設定を確認します。

```
# getenforce
Permissive
```

2.3.2.13 Defrag の無効化

1. defrag の設定を確認します。

```
# cat /sys/kernel/mm/transparent_hugepage/defrag
always defer defer+madvise [madvise] never
```

2. 手順 1 で「never」となっていない場合、root ユーザーもしくは sudo で、設定を変更します。

```
# echo never > /sys/kernel/mm/transparent_hugepage/defrag
# echo 'echo never > /sys/kernel/mm/transparent_hugepage/defrag' >>
/etc/rc.local
```

3. 手順 2 で設定変更した場合、任意のユーザーで、defrag の設定を確認します。

```
# cat /sys/kernel/mm/transparent_hugepage/defrag
always defer defer+madvise madvise [never]
```

2.3.2.14 時刻の同期設定

すべてのノードで時刻の同期がされている必要があります。同期は公開非公開を問わず任意の NTP サーバーで行うことができます。同期できる NTP サーバーがない場合、1 ノードをマスターサーバーとして他ノードを同期させる設定を行います。ここでは、後者の設定を行い、時刻の同期を行います。

1. マスターサーバー上で、root ユーザーもしくは sudo で/etc/chrony.conf の設定を変更します。以下の設定を行います。

```
# server xxx.xxx.xxx（すべての server 設定をコメントアウトします）
driftfile /var/lib/chrony/drift
makestep 1.0 3
allow 192.168.1.0/24
local stratum 8
```

```
keyfile /etc/chrony.keys
manual
```

2. マスターサーバー上で、root ユーザーもしくは sudo で、chronyd を開始し、OS 再起動時も自動実行されるように設定変更します。

```
# systemctl start chronyd
# systemctl enable chronyd
```

3. マスターサーバー以外のサーバー上で、root ユーザーもしくは sudo で/etc/chrony.conf の設定を変更します。以下の設定を行います。

```
server 192.168.1.110 iburst
driftfile /var/lib/chrony/drift
allow 192.168.1.110
local stratum 10
keyfile /etc/chrony.keys
```

4. マスターサーバー以外のサーバー上で、root ユーザーもしくは sudo で、chronyd を開始し、OS 再起動時も自動実行されるように設定変更します。

```
# systemctl start chronyd
# systemctl enable chronyd
```

5. マスターサーバー上で、任意のユーザーで、chrony が正常に動作していることを確認します。

```
# chronyc tracking

Reference ID    : 7F7F0101 ()
Stratum         : 8
Ref time (UTC)  : Mon Oct 14 06:02:59 2019
System time     : 0.000000000 seconds fast of NTP time
Last offset     : +0.000000000 seconds
RMS offset      : 0.000000000 seconds
Frequency       : 0.000 ppm slow
Residual freq   : +0.000 ppm
Skew            : 0.000 ppm
Root delay      : 0.000000000 seconds
Root dispersion : 0.000000000 seconds
Update interval : 0.0 seconds
Leap status     : Normal
```

Stratum が 8 になっていることを確認します。

6. マスターサーバー上以外のサーバーで、任意のユーザーで、chrony が正常に動作していることを確認します。

```
# chronyc tracking

Reference ID    : C0A8380B (192.168.1.110)
Stratum         : 9
Ref time (UTC)  : Mon Oct 14 06:07:52 2019
System time     : 0.000003991 seconds fast of NTP time
Last offset     : -0.009239891 seconds
RMS offset      : 0.009239891 seconds
Frequency       : 11.770 ppm fast
```

```
Residual freq   : -5.627 ppm
Skew            : 184.650 ppm
Root delay      : 0.000717187 seconds
Root dispersion : 0.012136783 seconds
Update interval : 64.1 seconds
Leap status     : Normal
```

Stratum が 9 になっていることを確認します。

2.3.2.15 CPU Frequency Scaling の無効化

1. root ユーザーもしくは sudo で、/etc/default/grub に設定を追記します。

```
# vi /etc/default/grub
GRUB_CMDLINE_LINUX="intel_idle.max_cstate=0 processor.max_cstate=1
intel_pstate=disable"
```

GRUB_CMDLINE_LINUX パラメータは存在しています。設定を追記します。

2. root ユーザーもしくは sudo で、OS 起動時の設定ファイルを生成します。

［BIOS を使用している環境の場合］

```
# grub2-mkconfig -o /boot/grub2/grub.cfg
```

［UEFI を使用している環境の場合］

```
# grub2-mkconfig -o /boot/efi/EFI/redhat/grub.cfg
```

2.3.2.16 TZ 環境変数の設定

1. root ユーザーもしくは sudo で、/etc/profile に下記を追加します。

```
# vi /etc/profile
export TZ="Asia/Tokyo"
```

2.3.2.17 LANG 環境変数の設定

LANG 環境変数は「en_US.UTF-8」を推奨しますが、それ以外の設定値でも問題ありません。ただし、すべてのノードで同一の設定を行っている必要があります。

2.3.2.18 Vertica の使用ポートが不使用であることの確認

Vertica で使用するポートが使用されていないことを確認します。

```
# ss -atupn
```

Vertica で使用するポートは次の通りです。

ポート番号	プロトコル	サービス
22	TCP	sshd
4803	TCP	Spread
4803	UDP	Spread
4804	UDP	Spread

ポート番号	プロトコル	サービス
5433	TCP	Vertica
5433	UDP	Vertica
5434	TCP	Vertica
5444	TCP	Management Console
5450	TCP	Management Console
5554	TCP	Node Management Agent
6543	UDP	Spread
8443	TCP	Built-in HTTPS service

2.3.2.19 サーバーの再起動

1. root ユーザーで、各サーバーを再起動します。

```
# reboot
```

2.4 インストールの実施

2.4.1 インストール用モジュールのアップロード

「rpm ファイルのダウンロード」の手順でダウンロードしたファイルを root ユーザーでノード#1 の/tmp ディレクトリにアップロードします。

2.4.2 rpm のインストール

Vertica 23.3 の rpm をインストールします。

1. 「インストール用モジュールのアップロード」の手順で rpm ファイルを格納したディレクトリに移動します。

```
# cd /tmp
```

2. root ユーザーで rpm をインストールします。

```
# dnf install ./vertica-23.3.0-0.x86_64.RHEL6.rpm
```

2.4.3 install_vertica スクリプトを使った Vertica のインストール

install_vertica スクリプトを使って、Vertica 23.3 をインストールします。

1. root ユーザーで install_vertica スクリプトを使って、クラスターを構築します。

   スクリプト実行中に、Linux OS 上に DB 管理者ユーザー（dbadmin）、DB 管理者ユーザーグループ（verticadba）が作成されます。異なるユーザーまたはグループを使用したい場合は--dba-user または--dba-group オプションを指定してください。

本書は、物理サーバーへのインストールを前提としておりますが、仮想サーバーにインストールされる場合は、下記オプションに加え、「--point-to-point」オプションを付与ください。詳細につきましては、下記マニュアルページを参照ください。

https://docs.vertica.com/23.3.x/en/setup/set-up-on-premises/install-using-command-line/install-with-installation-script/install-vertica-options/

```
# /opt/vertica/sbin/install_vertica --hosts [node#1 の Private IP],[node#2 の Private IP],[node#3 の Private IP] --rpm /tmp/vertica-23.3.0-0.x86_64.RHEL6.rpm --ssh-password [root のパスワード] --dba-user-password [DB 管理者（dbadmin）のパスワード] --license CE --accept-eula --failure-threshold FAIL
```

2.4.4 Vertica インストール後の設定

Vertica をインストール後、各サーバー上で下記の設定を実施します。

2.4.4.1 LANG 環境変数の設定

1. dbadmin ユーザーで、必要に応じて、/etc/profile、/home/dbadmin/.bashrc、あるいは、/home/dbadmin/.bash_profile を編集し、下記のように LANG 環境変数が UTF-8 となるように設定を追記します。

```
$ vi /home/dbadmin/.bashrc
export LANG=en_US.UTF-8
```

3 データベースの作成

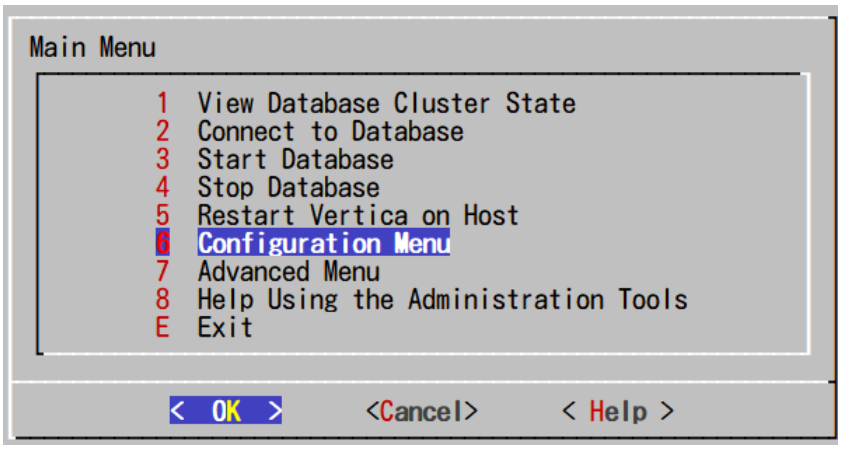
3.1 サンプルデータベースの作成

Administration Tools を使って、データベースを作成します。

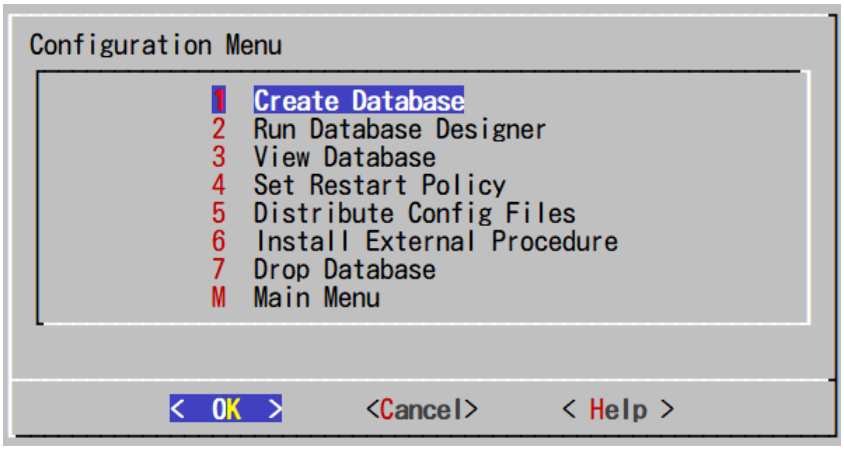
1. dbadmin ユーザーで、ノード#1 にログインします。
2. Administration Tools を起動します。

```
$ /opt/vertica/bin/admintools
```

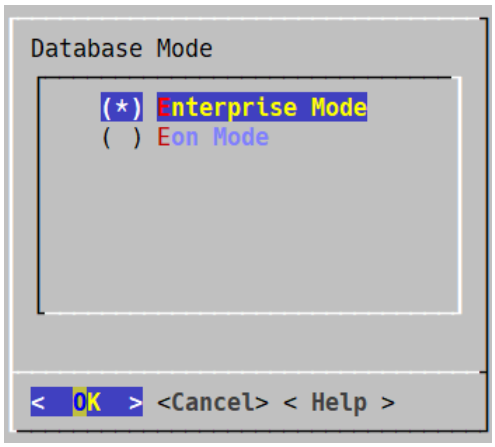
3. 「6 Configuration Menu」を選択します。



4. 「1 Create Database」を選択します。

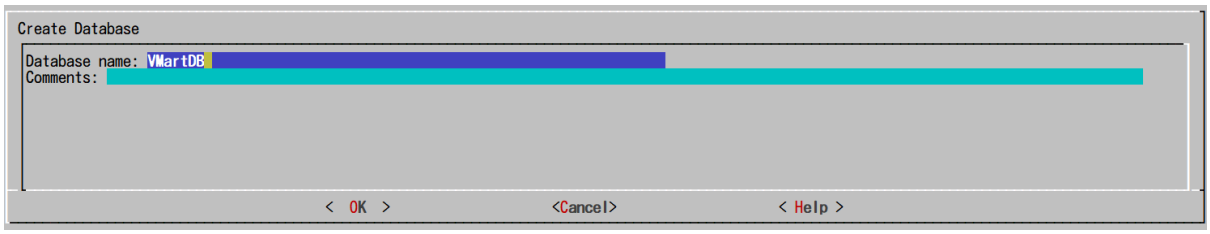


5. 「Enterprise Mode」を選択します。

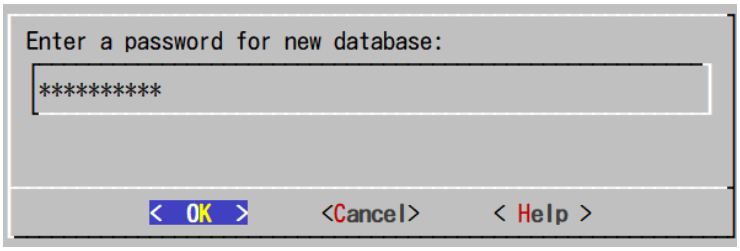


6. 任意のデータベース名を入力します。

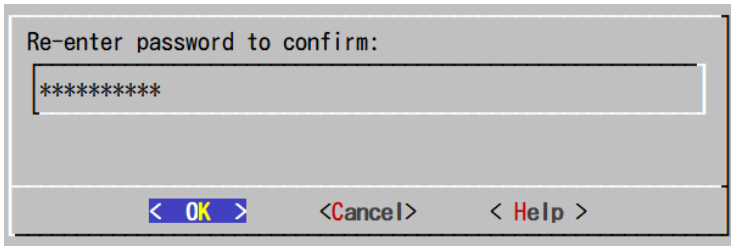
   本例では、「VMartDB」という名前のデータベースを作成しています。



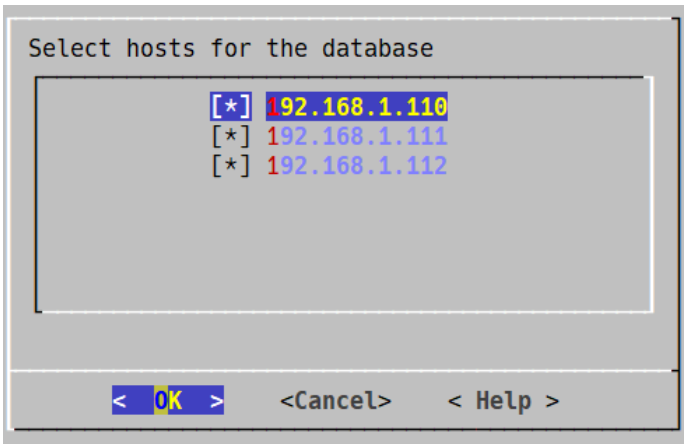
7. 任意のパスワードを入力します。



8. 再度、パスワードを入力します。



9. 作成するデータベースを構成するサーバーを選択します。

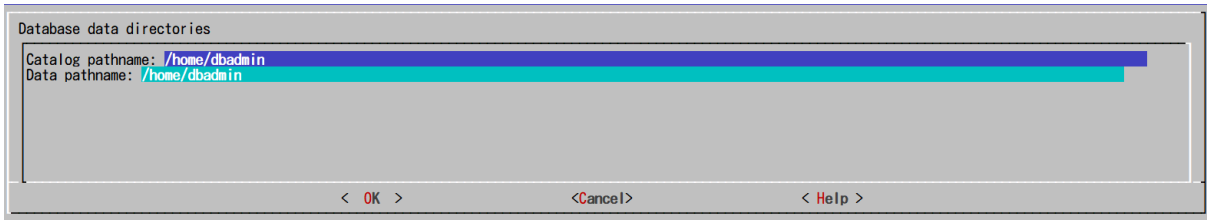


10. カタログファイルとデータファイルを格納するディレクトリを指定します。

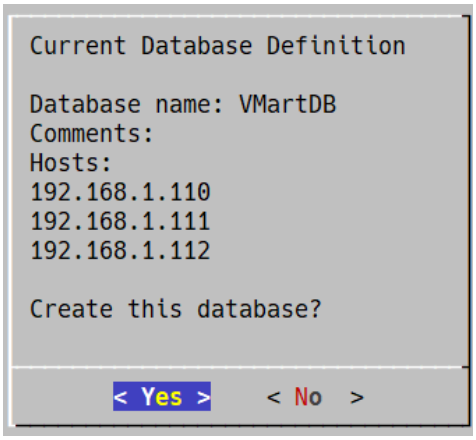
下記の例では、「/home/dbadmin」を指定しています。任意のディレクトリを指定することができますが、データベース作成前に、該当のディレクトリを作成する必要があります。

（作成例：/data にデータファイルを格納する場合）

```
# sudo mkdir /data
# sudo chown dbadmin:verticadba /data
```



11. 内容を確認し、「Yes」を選択します。

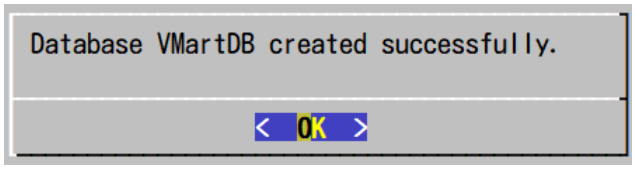


データベースの作成が行われます。

```
*** Creating database: VMartDB ***
	Creating database VMartDB
	Starting bootstrap node v_vmartdb_node0001 (192.168.1.110)
	Starting nodes:
		v_vmartdb_node0001 (192.168.1.110)
	Starting Vertica on all nodes. Please wait, databases with a large catalog may take a while
to initialize.
	Creating database nodes
	Creating node v_vmartdb_node0002 (host 192.168.1.111)
	Creating node v_vmartdb_node0003 (host 192.168.1.112)
	Generating new configuration information
	Stopping single node db before adding additional nodes.
		Database shutdown complete
	Starting all nodes
Start hosts = ['192.168.1.110', '192.168.1.111', '192.168.1.112']
	Starting nodes:
		v_vmartdb_node0001 (192.168.1.110)
		v_vmartdb_node0002 (192.168.1.111)
		v_vmartdb_node0003 (192.168.1.112)
	Starting Vertica on all nodes. Please wait, databases with a large catalog may take a while
to initialize.
	Node Status: v_vmartdb_node0001: (DOWN) v_vmartdb_node0002: (DOWN)
v_vmartdb_node0003: (DOWN)
	Node Status: v_vmartdb_node0001: (UP) v_vmartdb_node0002: (UP)
v_vmartdb_node0003: (UP)
Installing ComplexTypes package
	Success: package ComplexTypes installed
Installing DelimitedExport package
	Success: package DelimitedExport installed
Installing JsonExport package
	Success: package JsonExport installed
Installing MachineLearning package
	Success: package MachineLearning installed
Installing OrcExport package
	Success: package OrcExport installed
Installing ParquetExport package
	Success: package ParquetExport installed
Installing VFunctions package
	Success: package VFunctions installed
Installing approximate package
	Success: package approximate installed
Installing flextable package
	Success: package flextable installed
Installing kafka package
	Success: package kafka installed
Installing logsearch package
	Success: package logsearch installed
Installing place package
	Success: package place installed
Installing txtindex package
	Success: package txtindex installed
Installing voltagesecure package
```

```
        Success: package voltagesecure installed
Database creation SQL tasks completed successfully.
```

12. データベース作成が完了したら、OK を押して完了します。



3.2 サンプルデータの作成

製品で用意されているスクリプトを使って、サンプルデータを作成します。

1. dbadmin ユーザーで、ノード#1 にログインします。
2. サンプルデータ作成用のスクリプトが配置されているディレクトリに移動します。

```
$ cd /opt/vertica/examples/VMart_Schema/
```

3. サンプルデータを作成します。

```
$ ./vmart_gen
Using default parameters
datadirectory = ./
numfiles = 1
seed = 20177
null = ''
timefile = Time.txt
numfactsalesrows = 5000000
numfactorderrows = 300000
numprodkeys = 60000
numstorekeys = 250
numpromokeys = 1000
numvendkeys = 50
numcustkeys = 50000
numempkeys = 10000
numwarehousekeys = 100
numshippingkeys = 100
numonlinepagekeys = 1000
numcallcenterkeys = 200
numfactonlinesalesrows = 5000000
numinventoryfactrows = 300000
gen_load_script = false
years = 2003 to 2007
Data Generated successfully !
```

3.3 サンプルスキーマの作成

製品で用意されているスクリプトを使って、サンプルスキーマを作成します。

1. dbadmin ユーザーで、ノード#1 にログインします。

2. サンプルデータ作成用のスクリプトが配置されているディレクトリに移動します。

```
$ cd /opt/vertica/examples/VMart_Schema/
```

3. サンプルスキーマを作成します。

   vsql は文字ベースで対話型のフロントエンドユーティリティで、これを使用することにより、SQL 文を入力して結果を確認することができます。詳細は「vsql の使用方法」を参照してください。

```
$ /opt/vertica/bin/vsql -f vmart_define_schema.sql -w [指定したパスワード]
```

3.4 サンプルデータのロード

製品で用意されているスクリプトを使って、サンプルデータをロードします。

1. dbadmin ユーザーで、ノード#1 にログインします。
2. サンプルデータ作成用のスクリプトが配置されているディレクトリに移動します。

```
$ cd /opt/vertica/examples/VMart_Schema/
```

3. サンプルデータをロードします。

```
$ /opt/vertica/bin/vsql -f vmart_load_data.sql -w [指定したパスワード]
```

4 データベースの起動と停止

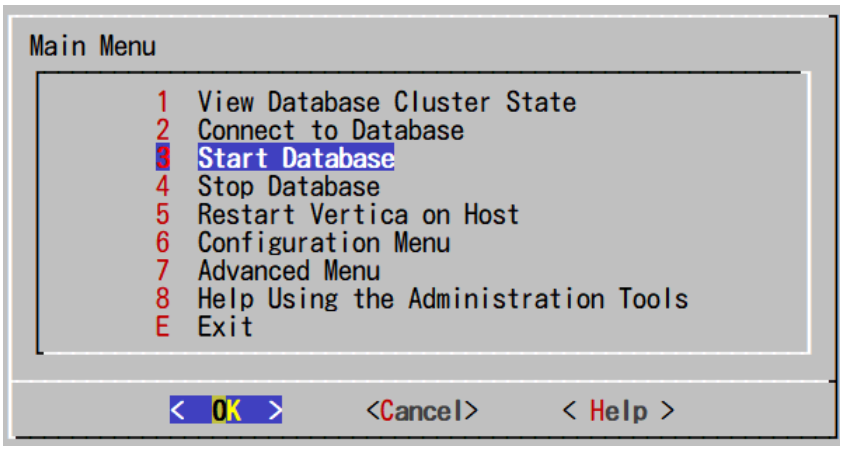
4.1 データベースの起動

Administration Tools を使って、データベースを起動します。

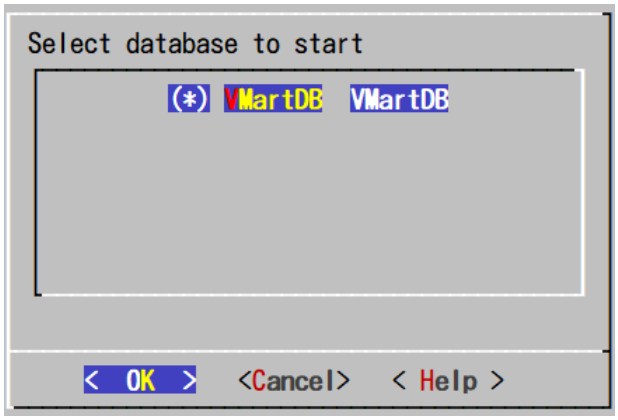
1. dbadmin ユーザーで、ノード#1 にログインします。
2. Administration Tools を起動します。

```
$ /opt/vertica/bin/admintools
```

3. 「3 Start Database」を選択します。

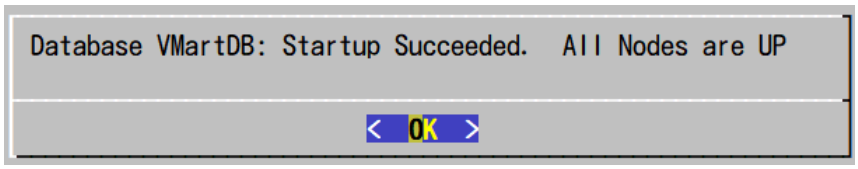


4. 起動するデータベースを選択します。



5. データベースのパスワードを入力します。

6. データベースの起動が完了したら、OK を押して完了します。



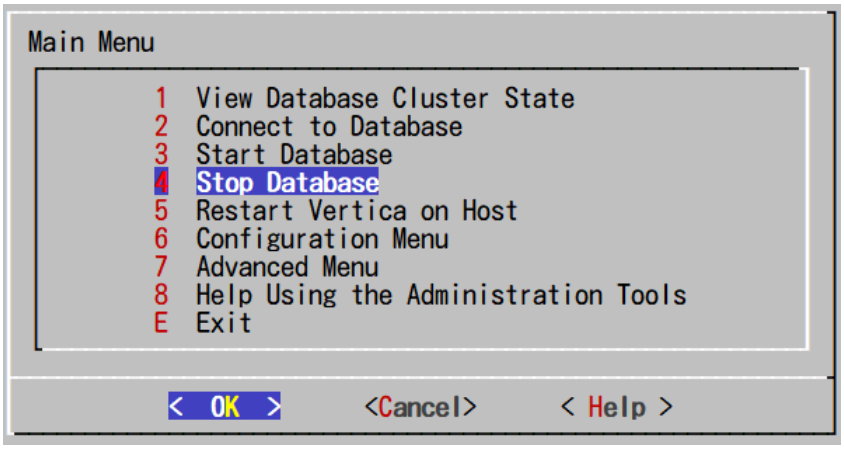
4.2 データベースの停止

Administration Tools を使って、データベースを停止します。

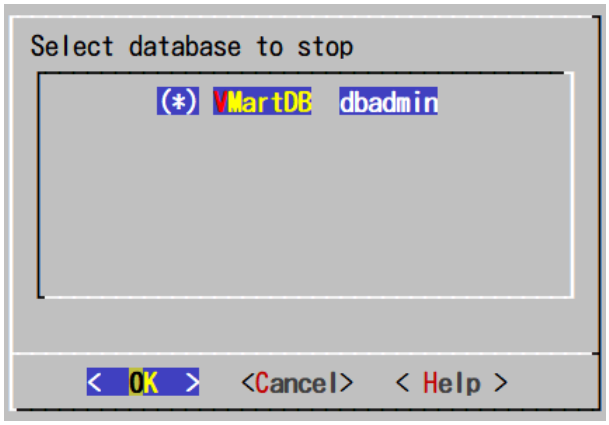
1. dbadmin ユーザーで、ノード#1 にログインします。
2. Administration Tools を起動します。

```
$ /opt/vertica/bin/admintools
```

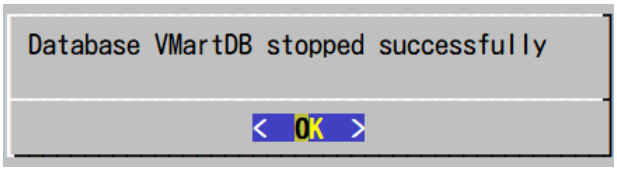
3. 「4 Stop Database」を選択します。



4. 停止するデータベースを選択します。



5. データベースのパスワードを入力します。
6. データベースの停止が完了したら、OK を押して完了します。



5 データベースの削除

5.1 データベースの停止

データベースが起動している場合、データベースの削除を実施する前に、データベースを停止する必要があります。データベースの停止の手順を参照し、データベースを停止します。

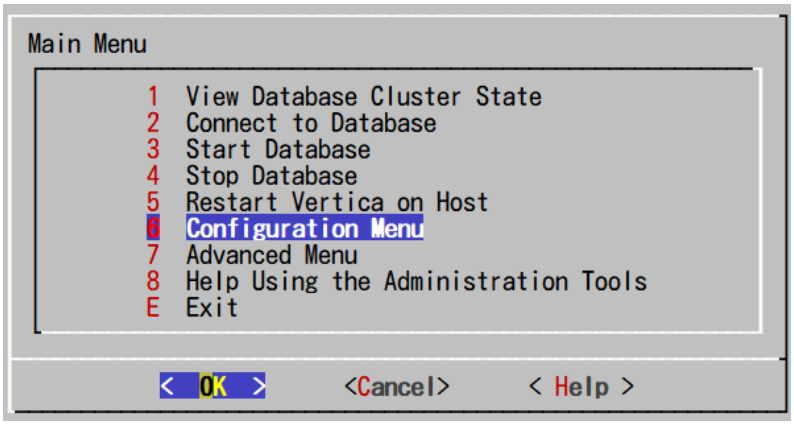
5.2 データベースの削除

Administration Tools を使って、データベースを削除します。

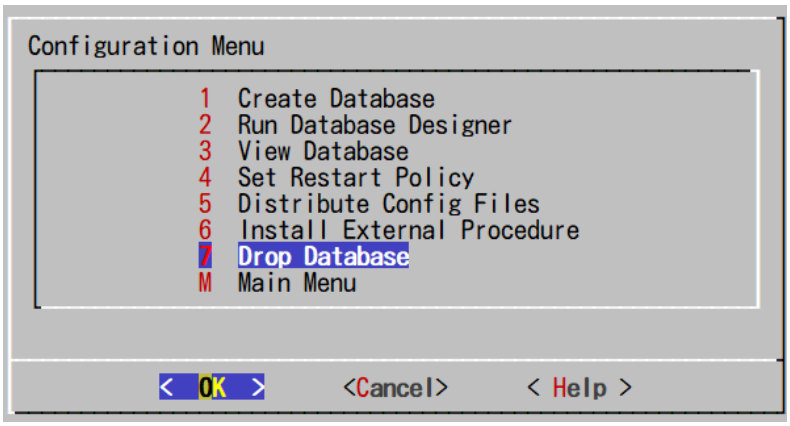
1. dbadmin ユーザーで、ノード#1 にログインします。
2. Administration Tools を起動します。

```
$ /opt/vertica/bin/admintools
```

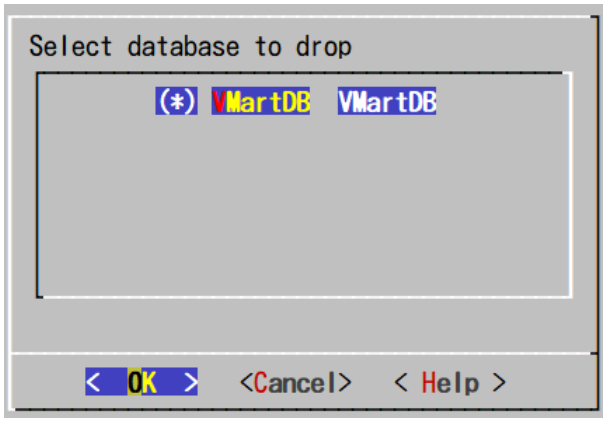
3. 「6 Configuration Menu」を選択します。



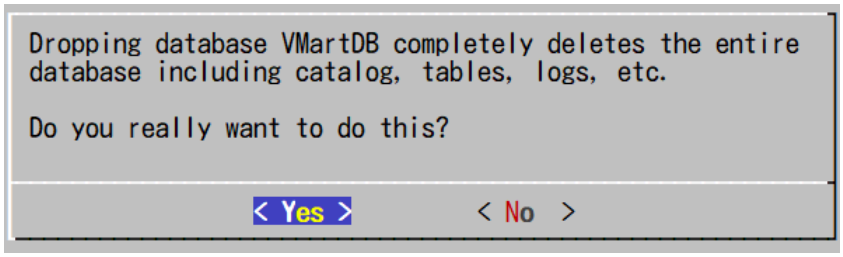
4. 「7 Drop Database」を選択します。



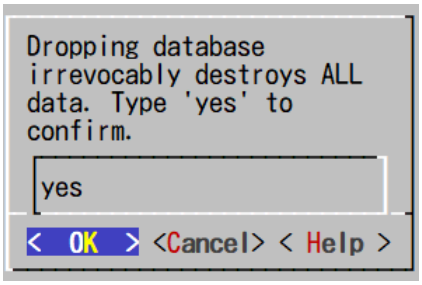
5. 削除するデータベースを選択します。



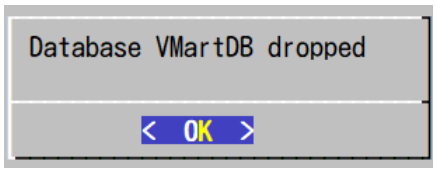
6. 確認画面で、「Yes」を選択します。



7. 本当に該当のデータベースを削除したい場合は、「yes」と入力します。



8. データベースの削除が完了したら、OK を押して完了します。



6 データベースデザイナーの実行

データベースデザイナーは、Vertica 上のデータの最適化を自動で実行するツールです。データベース管理者が、クエリの実行速度を向上させたい、あるいは、ストレージ容量をより節約したい場合等に実行します。テーブルを作成し、データロード後に実行することができます。通常、データベース全体の最適化を実施したい場合、Comprehensive モードで実行します。Comprehensive モードでデータベースデザイナーを実行後に、特定のクエリに対して追加で最適化を実行したい場合は、Incremental モードでデータベースデザイナーを実行します。

6.1 Comprehensive モードでのデータベースデザイナーの実行

Administration Tools を使って、データベースデザイナーを Comprehensive モードで実行します。

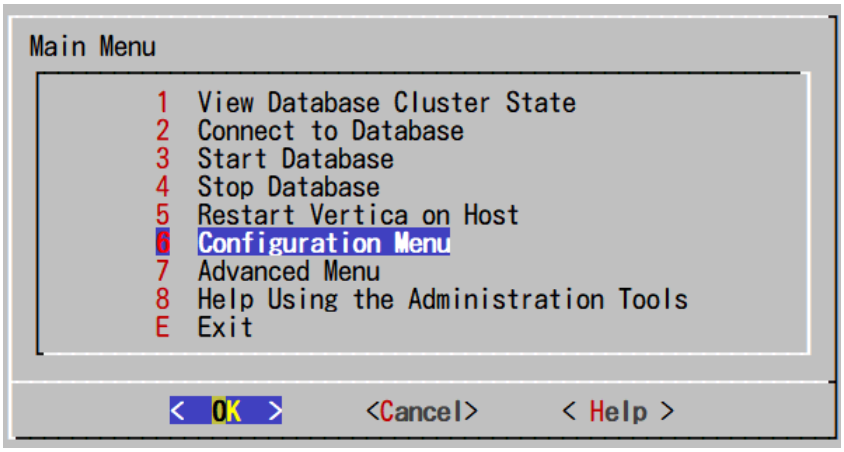
1. dbadmin ユーザーで、ノード#1 にログインします。
2. データベースデザイナーのログファイル等の出力ディレクトリを作成し、作成したディレクトリに移動します。

```
$ mkdir -p /home/dbadmin/DBD/comp
$ cd /home/dbadmin/DBD/comp
```

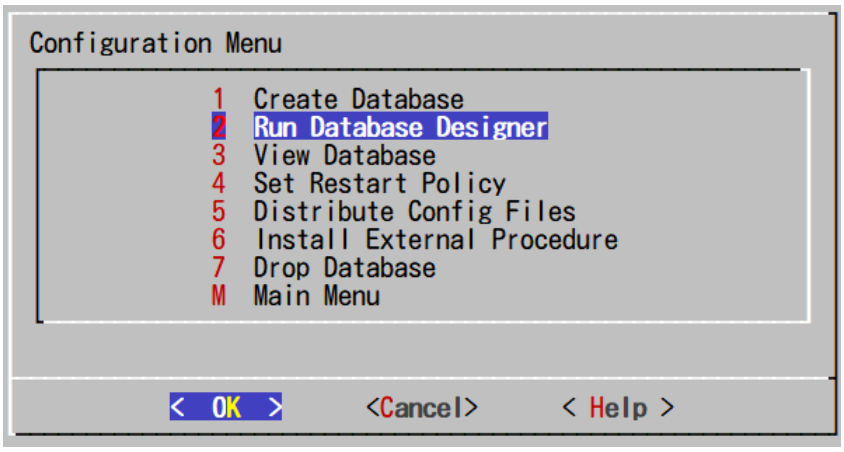
3. Administration Tools を起動します。

```
$ /opt/vertica/bin/admintools
```

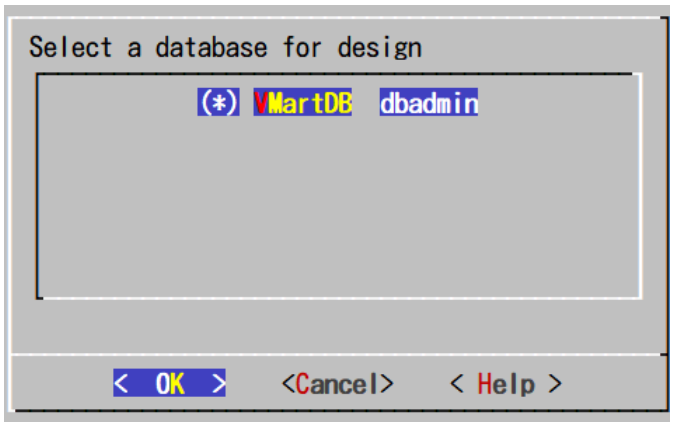
4. 「6 Configuration Menu」を選択します。



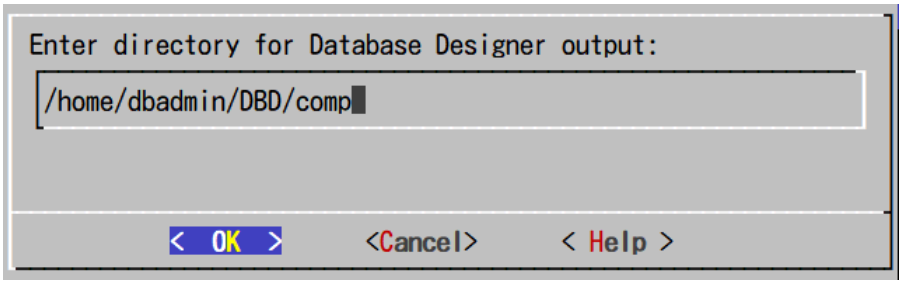
5. 「2 Run Database Designer」を選択します。



6. データベースデザイナーを実行するデータベースを選択します。

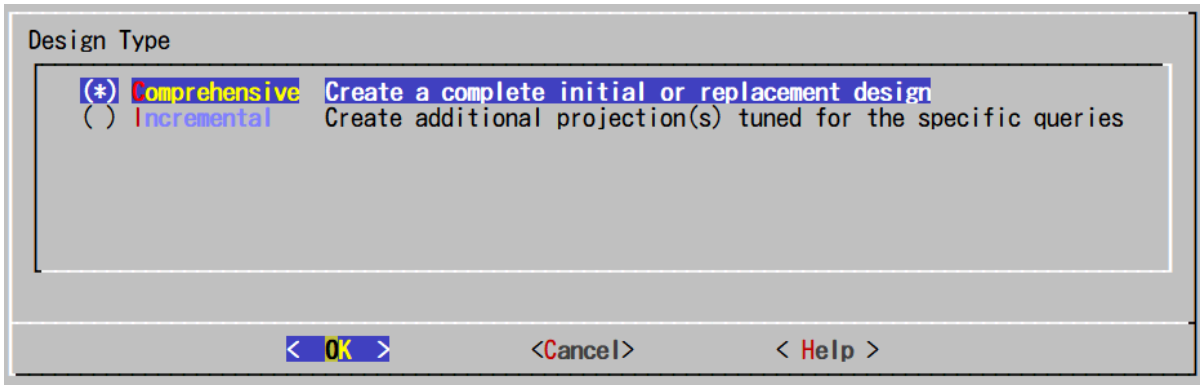


7. データベースのパスワードを入力します。
8. データベースデザイナーのアウトプットの出力先ディレクトリを指定します。

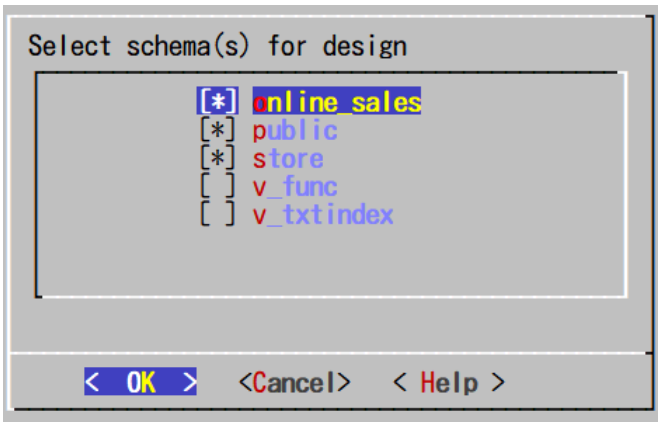


9. 任意のデザイン名を入力します。

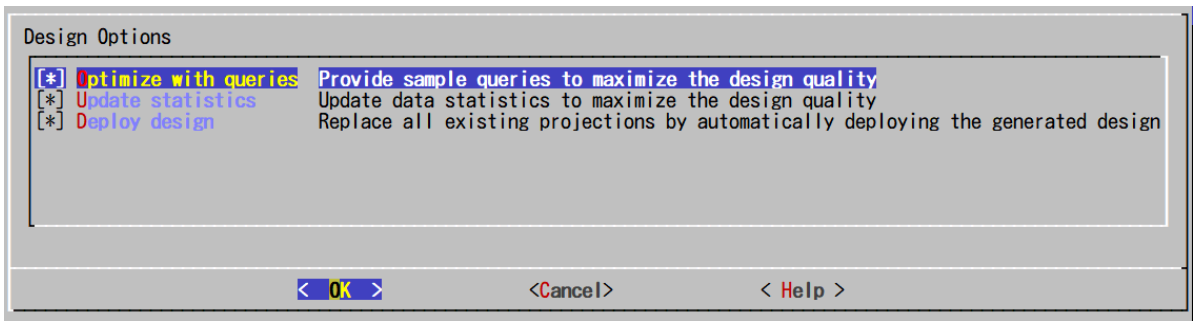
10. Design Type に「Comprehensive」を選択します。



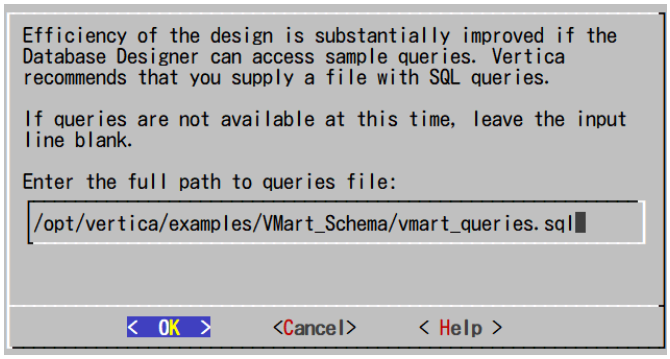
11. データベースデザイナーを実行する対象のスキーマを選択します。（サンプルスキーマを使用している場合は、「online_sales」「public」「store」を選択します。）



12. 全てのオプションをチェックします。（最適化するクエリが存在しない場合は、「Optimize with queries」のチェックを外します。）



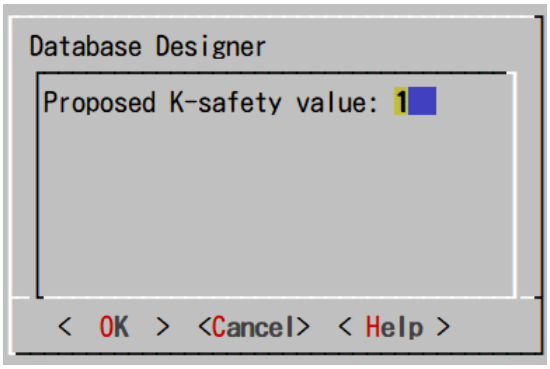
13. 最適化するクエリが存在する場合は、クエリファイルを指定します。（サンプルスキーマを使用している場合は、「/opt/vertica/examples/VMart_Schema/vmart_queries.sql」を指定します。）存在しない場合は空白のままにします。



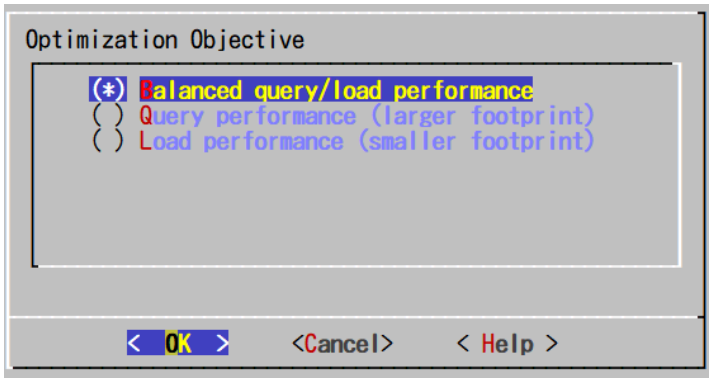
14. K-safety の値を「1」と指定します。

    K-safety は Vertica の高可用性を担保するためのパラメータになります。詳細は、以下を参照してください。

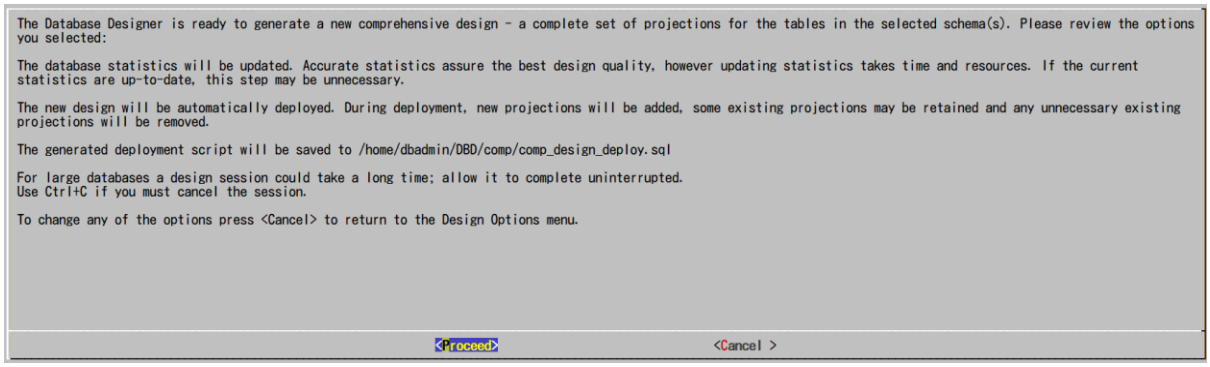
    https://docs.vertica.com/23.3.x/en/architecture/enterprise-concepts/k-safety-an-enterprise-db/



15. 「Balanced query/load performance」を選択します。



16. 「Proceed」を選択し、データベースデザイナーの実行を開始します。



17. データベースデザイナーの実行が終了したら、Enter を押して完了します。

```
Database Designer started.

   For large databases a design session could take a long time; allow it to complete uninterrupted.
   Use Ctrl+C if you must cancel the session.

   Setting up design session...

   Examining table data...

   Loading queries from '/opt/vertica/examples/VMart_Schema/vmart_queries.sql'.
      Processed 9 SQL statement(s), all accepted and considered in the design.

   Creating design and deploying projections...
      [100%] Design in progress...

   Query optimization results...
        Query 1 optimization ratio or status is 1
        Query 2 optimization ratio or status is 1
        Query 3 optimization ratio or status is 1
        Query 4 optimization ratio or status is 1
```

```
        Query 5 optimization ratio or status is 1
        Query 6 optimization ratio or status is 1
        Query 7 optimization ratio or status is 1
        Query 8 optimization ratio or status is 1
        Query 9 optimization ratio or status is 1

Deploying and generating deployment script...
      [100%] Deploying/Dropping projections...

   Completed 52 of 52 projections.
Design script is located in /home/dbadmin/DBD/comp/comp_design_design.sql
Deployment script is located in /home/dbadmin/DBD/comp/comp_design_deploy.sql

Database Designer finished.
Press <Enter> to return to the Administration Tools menu.
```

6.2 Incremental モードでのデータベースデザイナーの実行

Administration Tools を使って、データベースデザイナーを Incremental モードで実行します。

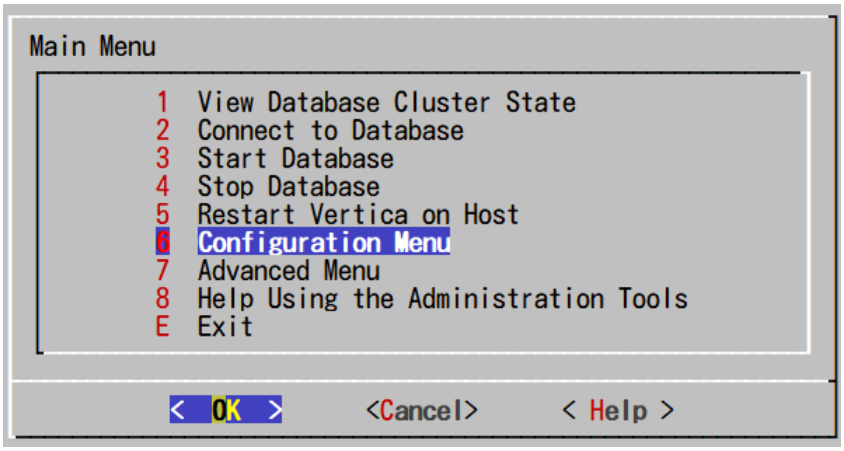
1. dbadmin ユーザーで、ノード#1 にログインします。
2. データベースデザイナーのログファイル等の出力ディレクトリを作成し、作成したディレクトリに移動します。

```
$ mkdir -p /home/dbadmin/DBD/incr
$ cd /home/dbadmin/DBD/incr
```

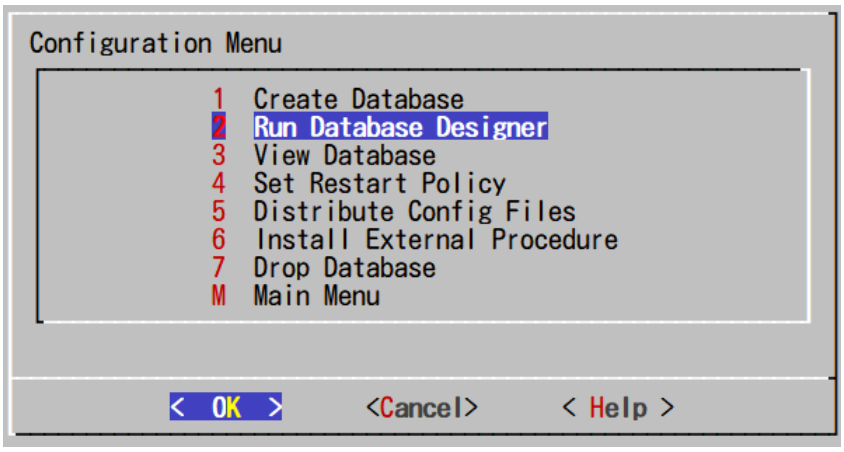
3. Administration Tools を起動します。

```
$ /opt/vertica/bin/admintools
```

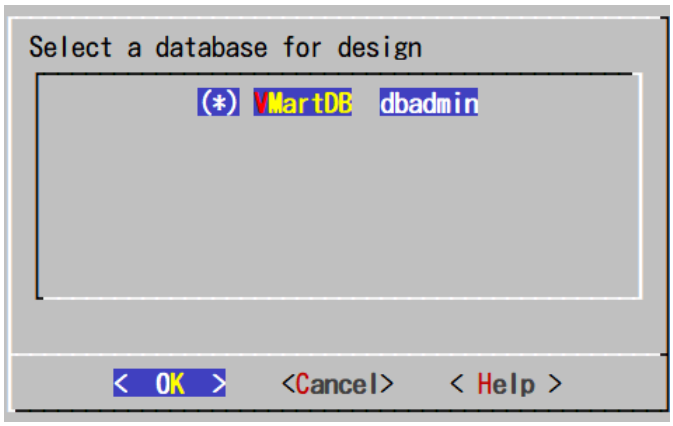
4. 「6 Configuration Menu」を選択します。



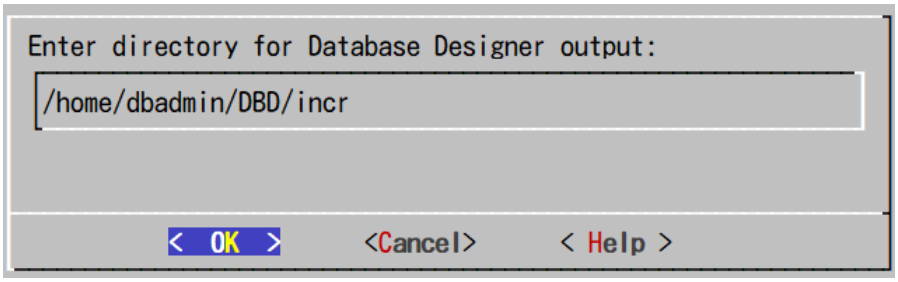
5. 「2 Run Database Designer」を選択します。



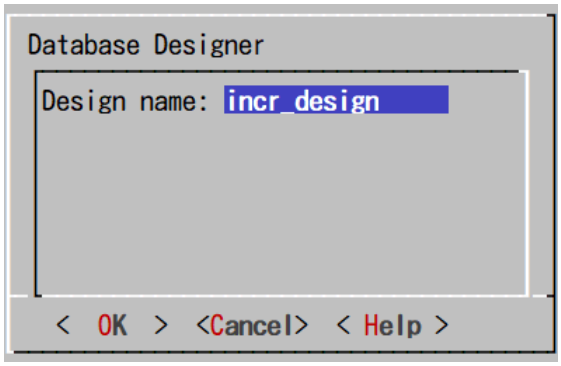
6. データベースデザイナーを実行するデータベースを選択します。



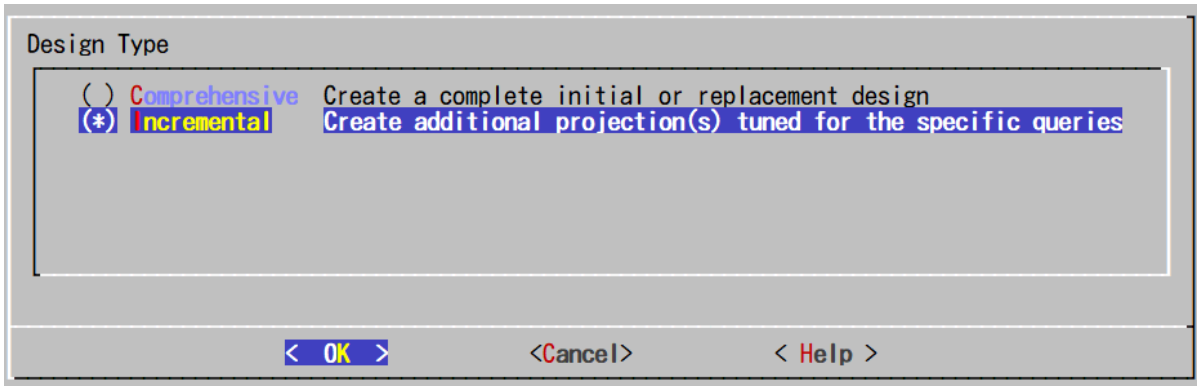
7. データベースのパスワードを入力します。
8. データベースデザイナーのアウトプットの出力先ディレクトリを指定します。



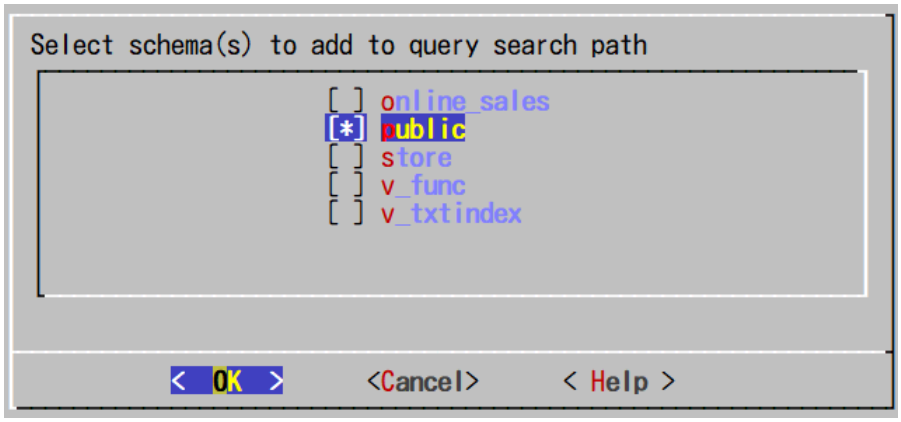
9. 任意のデザイン名を入力します。



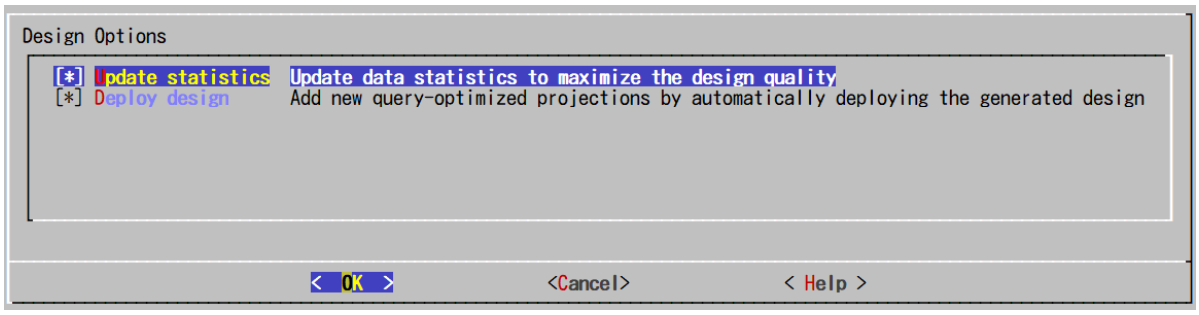
10. Design Type に「Incremental」を選択します。



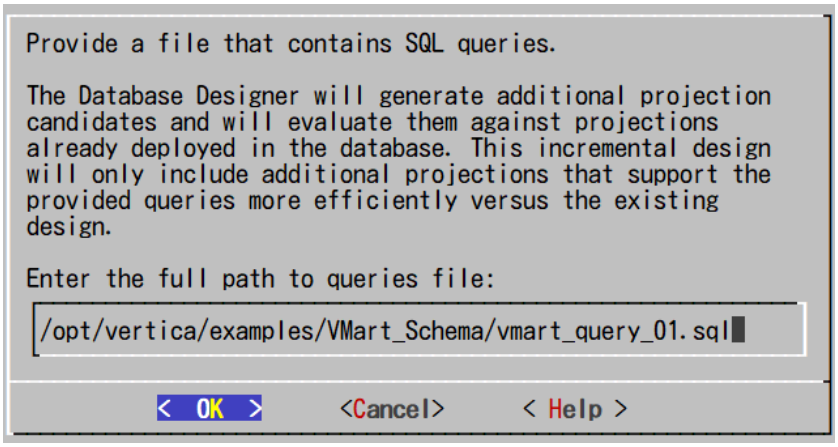
11. データベースデザイナーを実行する対象のスキーマを選択します。（最適化したいクエリが参照するスキーマを選択します。）



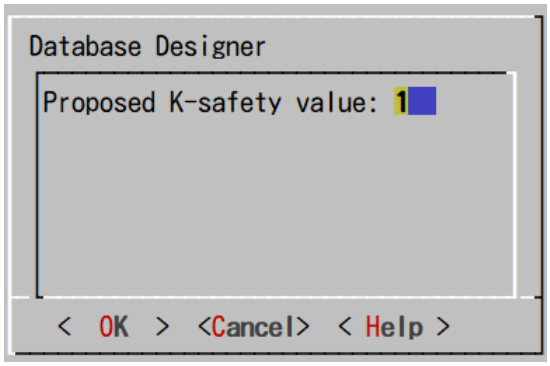
12. 全てのオプションをチェックします。



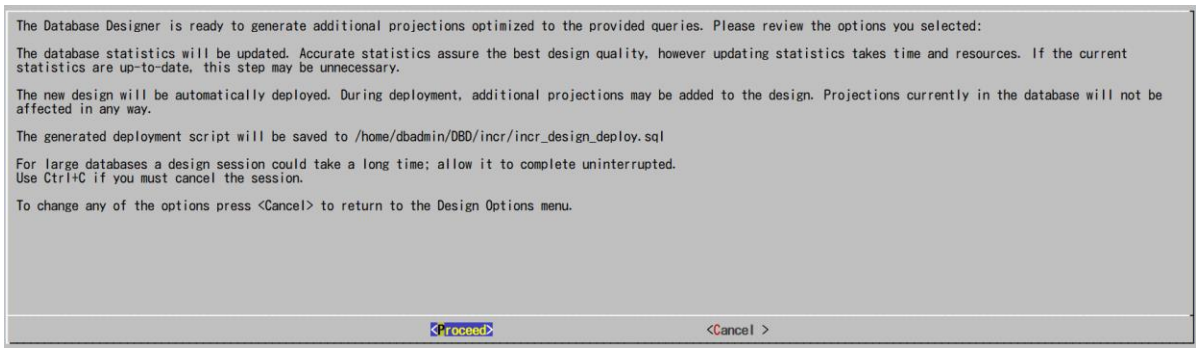
13. 最適化したいクエリファイルを指定します。



14. K-safety の値を「1」と指定します。



15. 「Proceed」を選択し、データベースデザイナーの実行を開始します。



16. データベースデザイナーの実行が終了したら、Enter を押して完了します。

```
Database Designer started.

   For large databases a design session could take a long time; allow it to complete uninterrupted.
   Use Ctrl+C if you must cancel the session.

   Setting up design session...
   Examining table data...

   Loading queries from '/opt/vertica/examples/VMart_Schema/vmart_query_01.sql'.
      Processed 1 SQL statement(s), all accepted and considered in the design.

   Creating design and deploying projections...
      [100%] Generating deployment script...

   Query optimization results...
       Query 1 optimization ratio or status is 1

Deploying and generating deployment script...
      [100%] Deploying/Dropping projections...

   Completed 2 of 2 projections.
Design script is located in /home/dbadmin/DBD/incr/incr_design_query_design.sql
Deployment script is located in /home/dbadmin/DBD/incr/incr_design_deploy.sql

Database Designer finished.
Press <Enter> to return to the Administration Tools menu.
```

7 マネージメントコンソールのインストール

マネージメントコンソール（以降、MC）は、Web ベースの GUI の管理ツールであり、システムの状況をグラフィカルに表示したり、あるいは、データベースの停止・起動等のデータベースの管理を行うことができます。ここでは、MC を Vertica が既にインストールされているノードとは別のノード上にインストールする方法をご紹介します。MC を Vertica が既にインストールされているノードと同じノードにインストールする場合、Resource Pool の設定などを変更する必要性があります。

7.1 MC ソフトウェアのインストールの実施

1. root ユーザーで、ノードにログインします。
2. 「インストールの実施 - インストール用モジュールのアップロード」の手順で rpm ファイルをアップロードしたディレクトリに移動します。

```
# cd /tmp
```

3. rpm をインストールします。

```
# dnf install ./vertica-console-23.3.0-0.x86_64.RHEL6.rpm
```

7.2 MC の初期設定

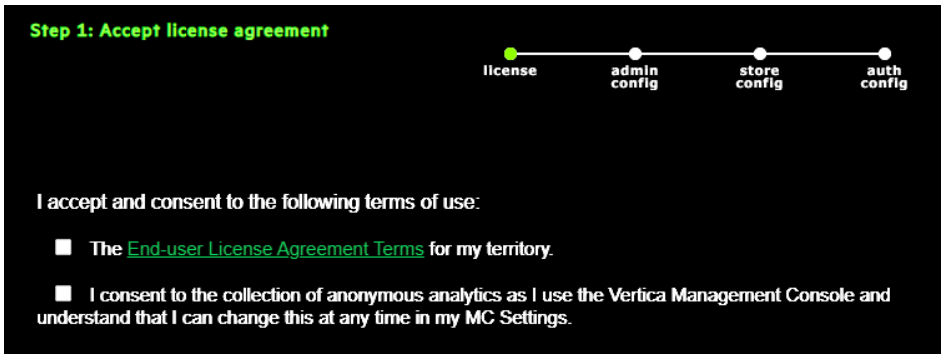
rpm をインストール後、MC の初期設定を実施します。

1. ブラウザから、URL「https://[ノードの IP アドレス]:5450/webui」に接続します。
   23.3 の MC がサポートしているブラウザは以下の通りです。

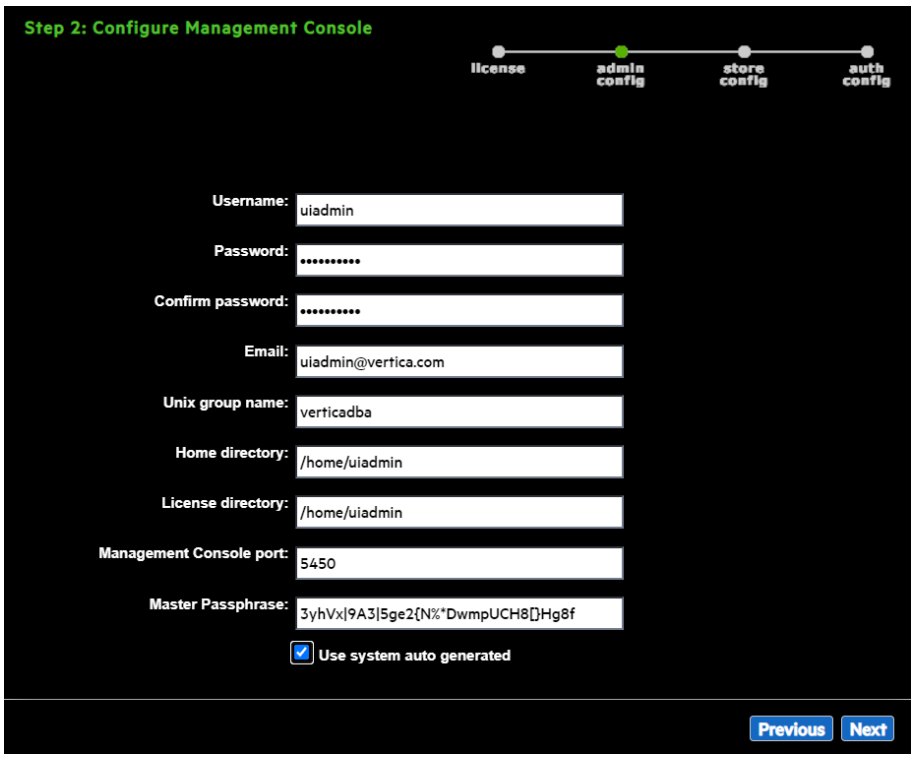
   - Chrome
   - Firefox
   - Microsoft Edge

   MC はインストール直後では自己署名証明書を使用しています。そのため、ブラウザによってはセキュリティの警告が表示されることがあります。

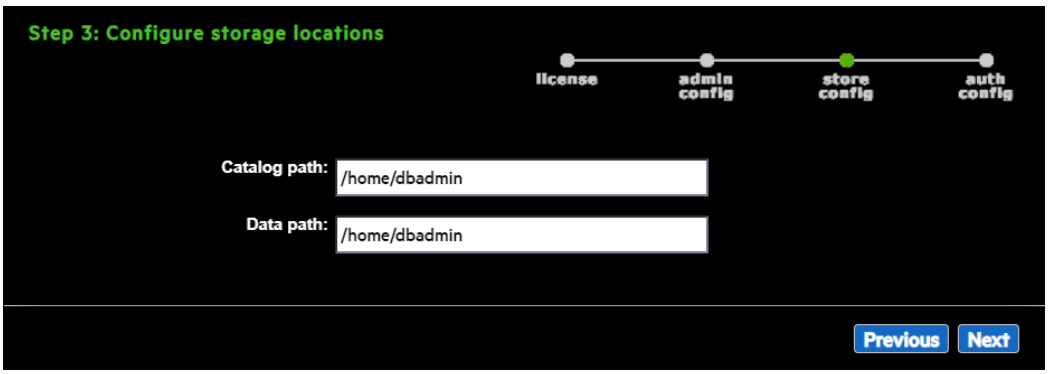
2. License agreement にチェックを入れ、「Next」をクリックします。2 つ目のチェックは MC のユーザー使用統計情報の収集と提供を許可するかの確認になります。



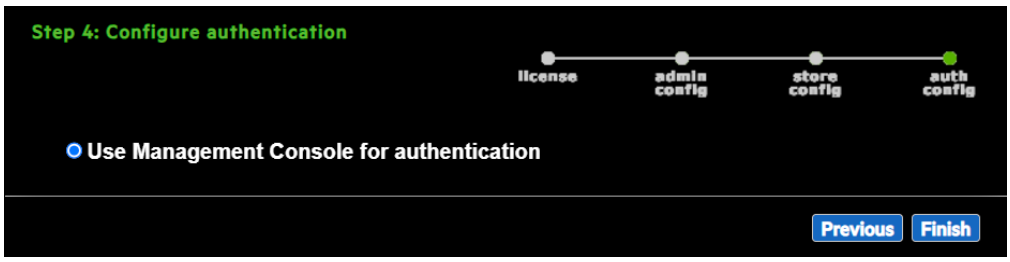
3. MC の管理者ユーザーの情報を入力し、「Next」をクリックします。この情報を基に OS ユーザーが作成されます。また、MC のログインユーザーとしても登録されます。



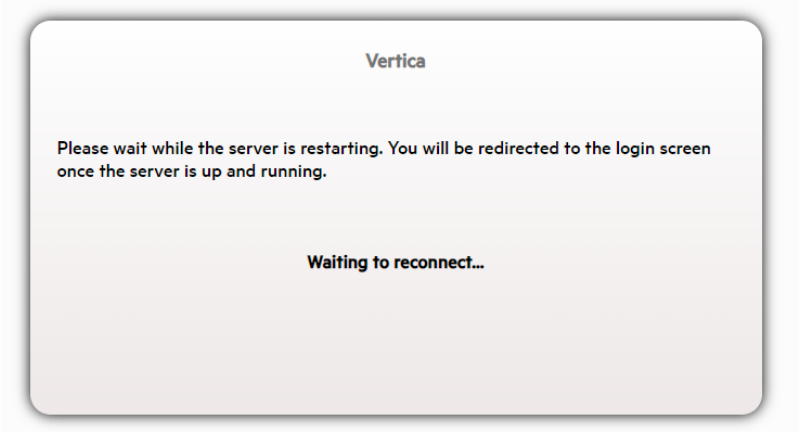
4. MC からデータベースを作成する場合のデフォルトのカタログ・データ格納ディレクトリを入力し、「Next」をクリックします。ディレクトリは作成時に変更できます。



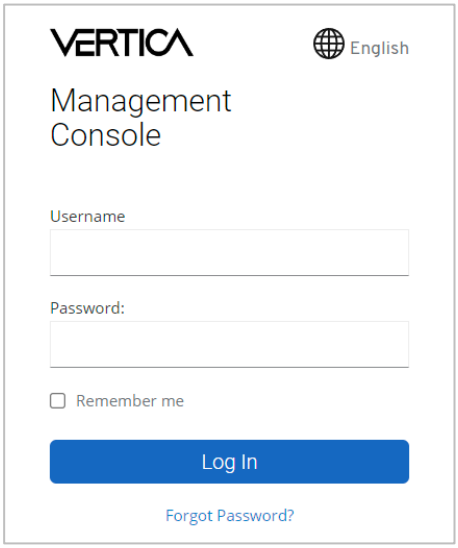
5. MC にログインするためのユーザー認証方式として「User Management Console for authentication」を選択し、「Finish」をクリックします。



6. 設定が保存され、MC サービスが再起動されます。



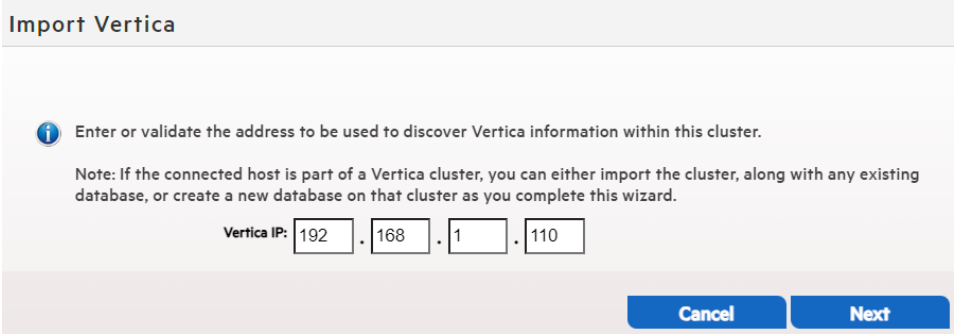
7. 再起動が終わると、ログイン画面に変わります。管理者ユーザーの情報を入力し、「Log in」をクリックします。



8. Email gateway が設定されていない旨を表すウィンドウと新機能を紹介するウィンドウが表示されます。必要に応じて参照してください。「OK」をクリックして、ウィンドウを閉じます。

9. 既存のクラスターを認識させるため、「Import database」をクリックします。

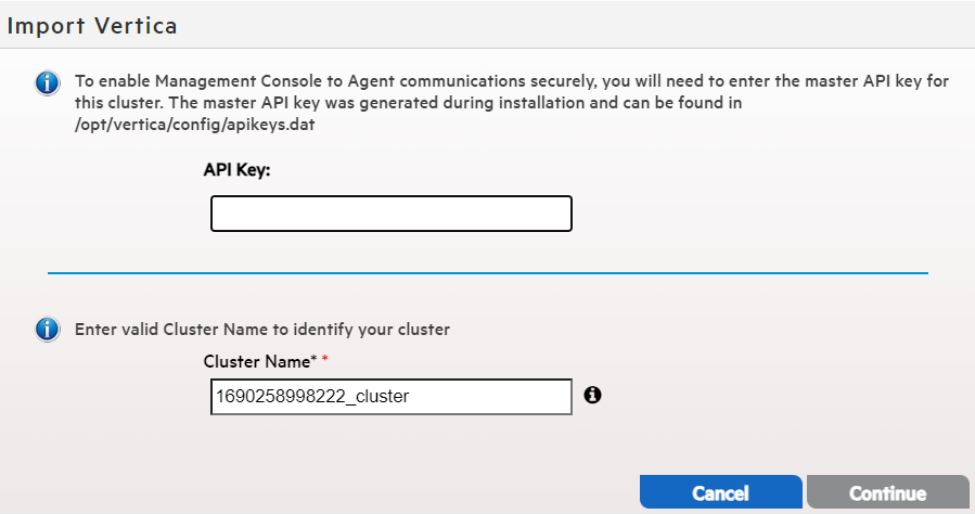
10. ノード#1 の Private ネットワークの IP を入力し、「Next>>」をクリックします。もし、MC をインストールしたノードがノード#1 と Private ネットワークで通信できない場合、Public ネットワークの IP を入力します。



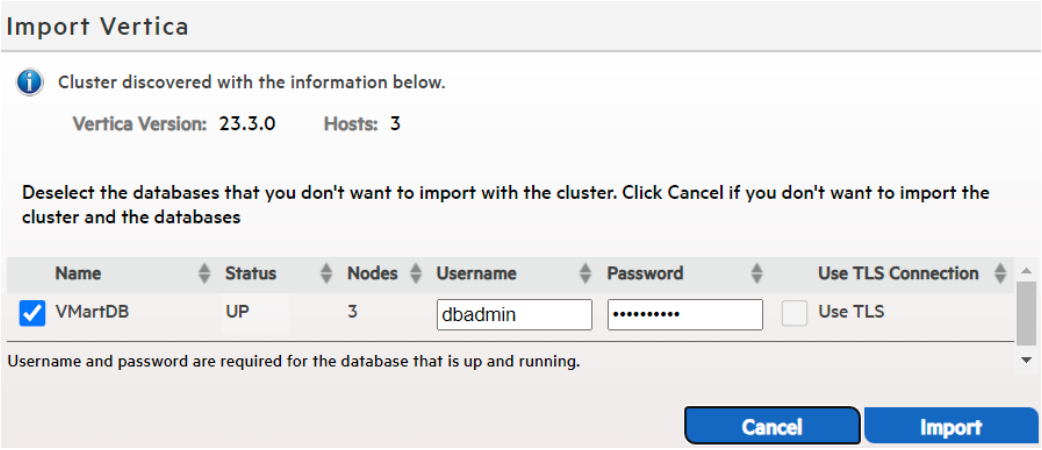
11. dbadmin ユーザーで、ノード#1 にログインします。/opt/vertica/config/apikeys.dat の"apikey"の値を確認します。

```
$ cat /opt/vertica/config/apikeys.dat
[
   {
      "apikey": "N5EbanYyoBDQdnTT/5ES7G4w0469r4s/H0gVNA",
      "app": "vertica",
      "level": "admin",
      "requestor": "master"
   }
]
```

12. 確認した API Key を入力し、任意の Cluster の名前を入力します。「Continue」をクリックします。



13. 作成したデータベースにチェックが入っていることを確認し、Username に「dbadmin」、Password にパスワードを入力し、「Import」をクリックします。



14. クラスターの認識が行われます。完了後、「Done」をクリックします。認識されると各種情報の表示が行えるようになります。

8 vsql の使用方法

vsql は文字ベース・対話型のフロントエンドユーティリティで、これを使用することにより、SQL 文を入力して結果を確認することができます。また、メタコマンドや、スクリプトを作成して様々なタスクの自動化を簡単にするシェルのような機能が多数用意されています。本章で、vsql を使用するためのいくつかの一般的なコマンドを記載します。

8.1 vsql の起動

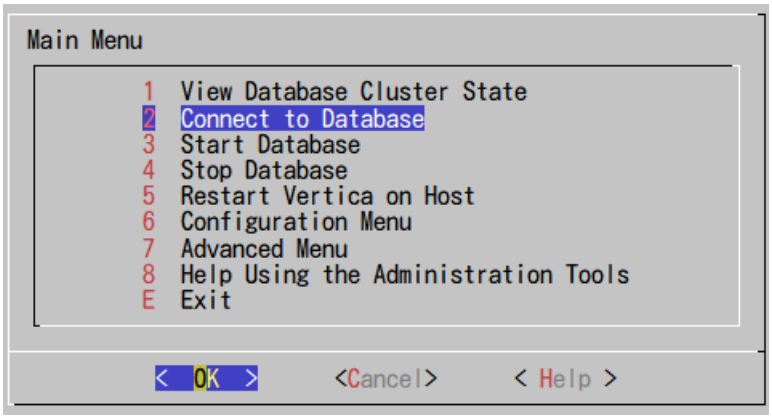
vsql は Administration Tools から起動するか、あるいは、コマンドプロンプトからクエリやスクリプトが直接実行できます。

8.1.1 Administration Tools からの起動

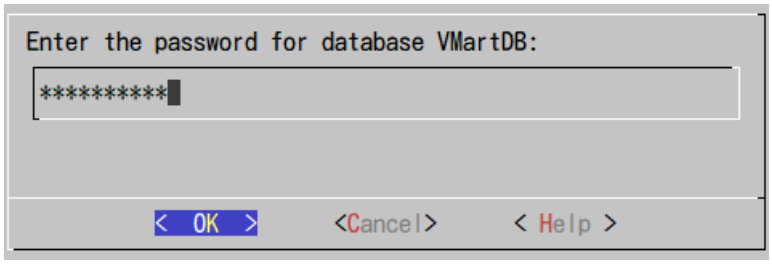
1. dbadmin ユーザーで、ノード#1 にログインします。Administration Tools を起動します。

```
$ /opt/vertica/bin/admintools
```

2. Main Menu 上で、「2 Connect to Database」を選択し、OK をクリックします。



3. データベースのパスワードを入力します。



8.1.2 Linux のコマンドプロンプトからの起動

1. dbadmin ユーザーで、いずれかのノードにログインします。vsql --[vsql command]と入力します。例えば、「vsql --help」と実行すると、vsql の引数一覧が確認できます。

8.2 メタコマンド

データベースに接続した後に使用できるメタコマンドは次の通りです。

コマンド	内容
¥?	ヘルプ
¥h	ヘルプ
¥q	vsql の終了
¥dt	ユーザーテーブル一覧
¥dj	プロジェクション一覧
¥dS	システムテーブル一覧
¥d <table name>	ユーザーテーブルの詳細
¥dn	スキーマ一覧
¥timing	タイミングの切り替え。実行時間を確認したい場合に設定。
¥i	SQL スクリプトファイルの実行

9 最後に

この文書につきまして、フィードバック等ございましたら、以下までメールでお問い合わせください。

- 宛先メールアドレス：MFI-jpn_vertica_info@opentext.com
- メール件名：Vertica Community Edition 日本語ガイドに関するお問い合わせ

英語での記載とはなりますが、下記弊社公式サイトにて多くの有用なドキュメントが公開されておりますので、そちらもあわせてご活用ください。

https://www.vertica.com/

Copyright © 2023 Open Text. All rights reserved. Trademarks owned by Open Text