

Vertica Community Edition 日本語ガイド Vertica Analytics Platform

March, 2021

Contents

はじめに.....	3
本書の目的	3
記載範囲	3
前提	3
Vertica のインストール.....	4
インストールを始める前に	4
ネットワーク要件	4
マシン構成例	4
rpm ファイルのダウンロードの実施.....	4
OS 設定の実施	8
OS をインストールする前のチェック項目	8
OS インストール後の設定	8
インストールの実施	15
インストール用モジュールのアップロード	15
rpm のインストール	16
install_vertica スクリプトを使っての Vertica のインストール	16
Vertica インストール後の設定	16
データベースの作成.....	17
サンプルデータベースの作成	17
サンプルデータの作成	21
サンプルスキーマの作成	22
サンプルデータのロード	23
データベースの起動と停止.....	24
データベースの起動	24
データベースの停止	25
データベースの削除.....	26
データベースの停止	26
データベースの削除	26
データベースデザイナーの実行.....	28
Comprehensive モードでのデータベースデザイナーの実行.....	28
Incremental モードでのデータベースデザイナーの実行.....	33
マネージメントコンソールのインストール.....	38
MC ソフトウェアのインストールの実施.....	38
MC の初期設定	38
最後に.....	47
Appendix I: vsql.....	48
vsql の起動	48
Administration Tools からの起動.....	48
Linux のコマンドプロンプトからの起動	48

はじめに

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（jpn_vertica_info@microfocus.com）までメールでお問い合わせください。

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
※格納データの最適配置をするためのもので、必須ではありません。
- 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 10.1.x Documentation
<https://www.vertica.com/docs/10.1.x/HTML/Content/Home.htm>
 - Vertica 10.1.x Supported Platforms
 - Installing Vertica
 - Getting Started

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：CentOS 8.3 64bit
- Vertica Community Edition のバージョン：10.1.0-0

インストール環境は、CentOS 8.3 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。CentOS 以外のオペレーティングシステムを利用される場合、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。

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。

Vertica のインストール

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。

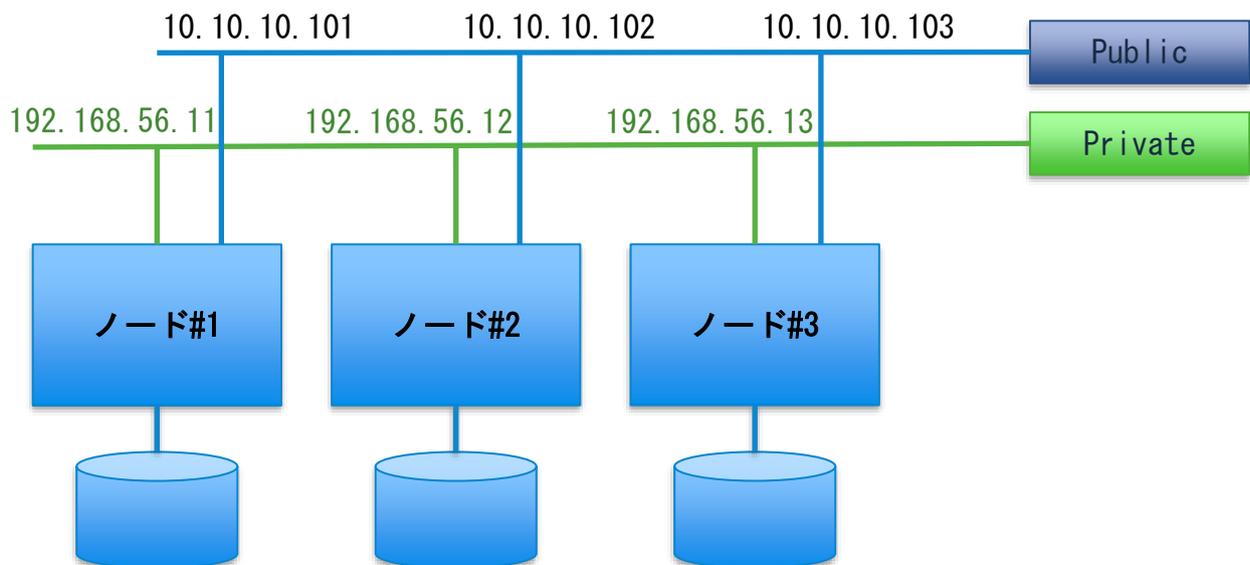
ネットワーク要件

Vertica は、Private と Public の 2 系統を推奨します。

- 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 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。

また、10 ギガビット・イーサネット以上を推奨します。

マシン構成例



rpm ファイルのダウンロードの実施

下記の手順に従って、Community Edition ユーザーとして登録し、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/register/> へ接続します。

2. 下記の画面に必要な情報を英語で入力し、「Sign-up to Start Your Free Trial Today」をクリックします。

VERTICA Product Solutions Partners Resources About Support Try Vertica

Looking for a Big Data Analytics solution? Check out the Vertica Community Edition Free Trial.

Accelerate your time to insights using Community Edition. Vertica is the industry's most unified analytics warehouse that enables your organization to derive maximum value from a big data analytics and machine learning solution. Store and analyze up to 1 TB of data for free with no time limit. Install Vertica CE on-premises, as a VM, on Apache Hadoop, in your choice of clouds (AWS, Azure, Google), or in hybrid environments.

- Install Vertica CE on up to 3 nodes
- Store and analyze up to 1 TB of structured and semi-structured data
- Use Vertica for SQL on Apache Hadoop for data exploration as part of Vertica CE free trial
- Enjoy no time limits or license requirements

The Vertica Analytics Platform is consciously designed with speed, scalability, simplicity, and openness at its core and architected to handle the largest analytical workloads via a distributed compressed columnar architecture. Fueled by petabyte-sized volumes of Big Data found in both small and large, data-driven organizations, the Vertica Big Data Analytics Platform is a SQL analytics solution built from the ground up to handle massive volumes of data and delivers blazing fast big data analytics.

The platform is available in the broadest range of deployment and consumption models, including on premise, on Hadoop, and in the clouds, or a hybrid model. Whether you're just starting out or have been an innovator in big data analytics for a long time, Vertica delivers the proven performance and scale for today's data-driven world.



Complete this Form to Start Your Free Trial Today

[Already registered for Vertica Community Edition?](#)
[Are you a returning customer?](#)

Basic Information

<input type="text" value="First Name*"/>	<input type="text" value="Last Name*"/>
<input type="text" value="Company*"/>	<input type="text" value="Job Title"/>
<input type="text" value="Phone*"/>	<input type="text" value="City*"/>
<input type="text" value="Select a Country*"/>	<input type="text" value="State / Province"/>
<input type="text" value="Zip / Postal Code*"/>	

Get started by downloading and installing the Vertica Server.

Account Credentials

Password Strength

A strong password is required. Suggestions for a strong password:
Use both uppercase and lowercase letters
Include one or more number(s)
Include special characters, such as @, #, \$
Avoid using any personal information, common words or phrases

I'd like to receive emails from Micro Focus to stay up-to-date on products, services, news, events and promotions*
 Yes No

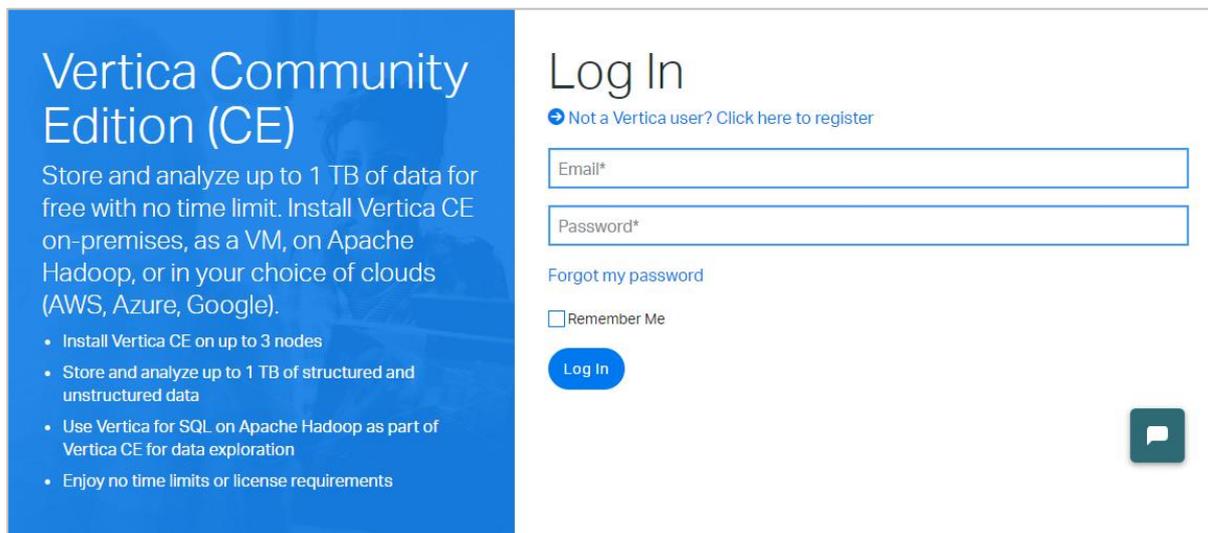
I'm not a robot 
[Privacy - Terms](#)

[Micro Focus Privacy Statement](#)

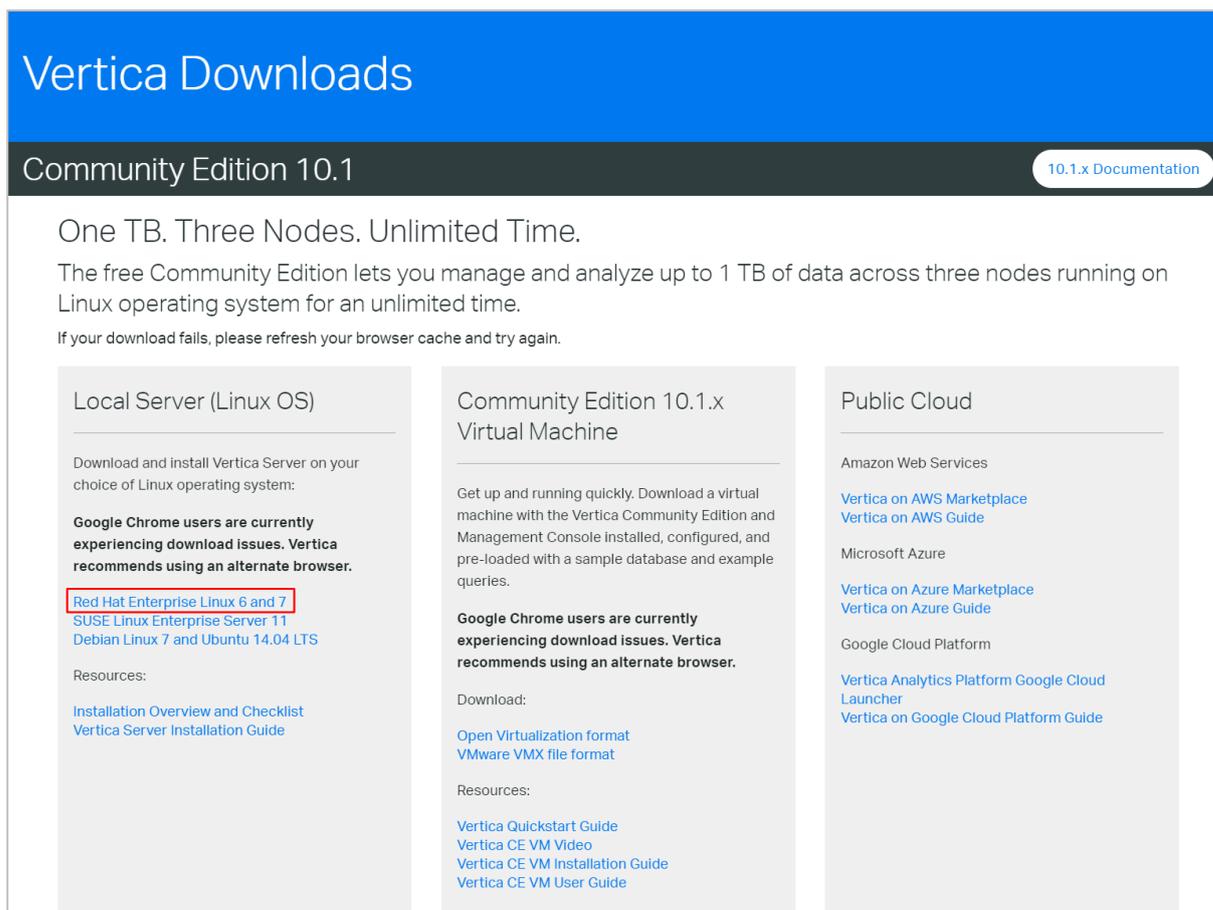
You are free to have your details removed from our data base at any time. Visit <http://go.microfocus.com/company/subscription/> at any time to manage your preferences.

By clicking the Register button, you agree to the [End User Agreement](#)

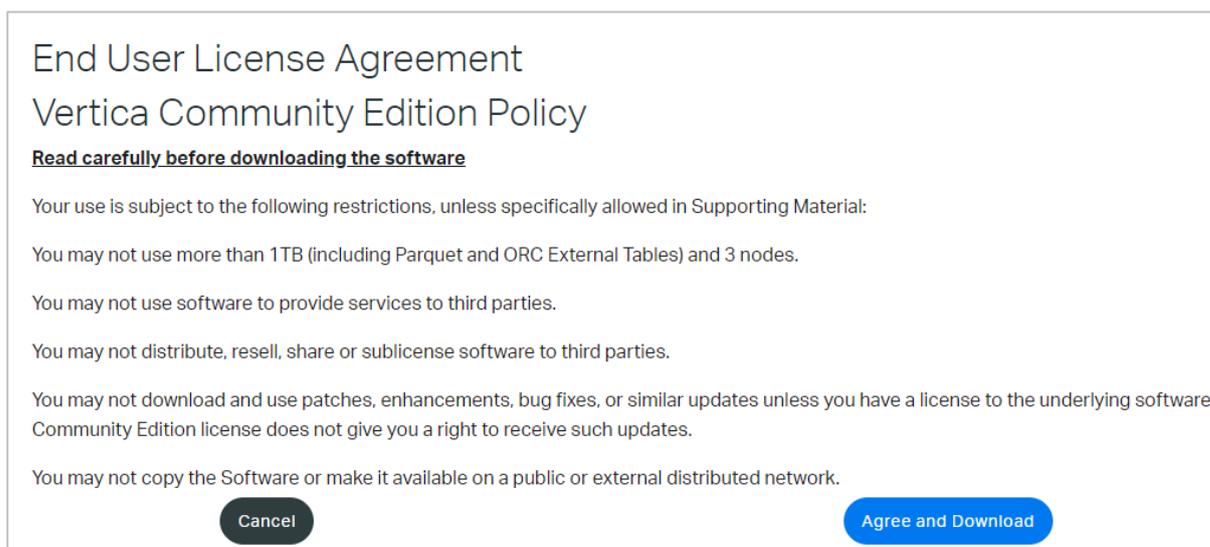
3. <https://www.vertica.com/download/vertica/community-edition/> へ接続し、ログイン情報を入力します。



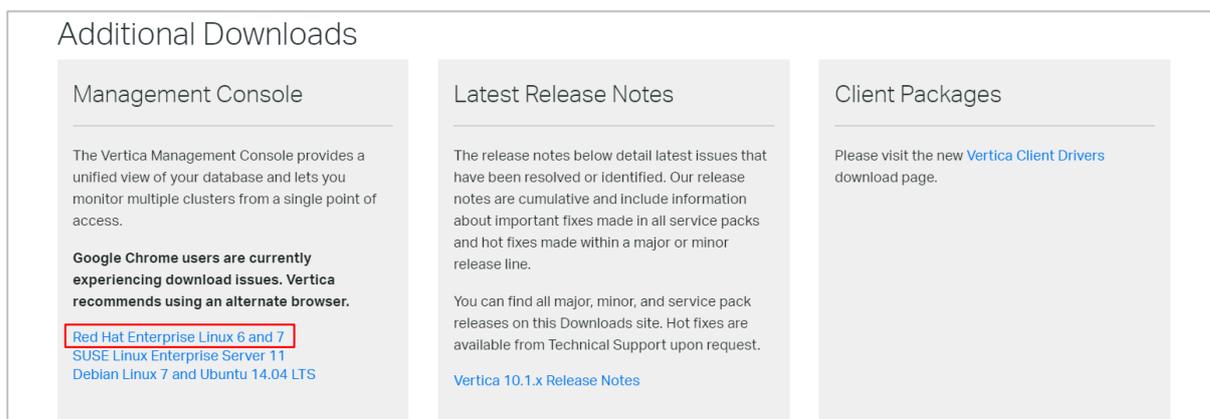
4. サーバー用のモジュールをダウンロードするために、「Local Server (Linux OS)」から「Red Hat Enterprise Linux 6 and 7」をクリックします。



5. ソフトウェア使用許諾契約の内容を確認し、「Agree and Download」をクリックします。



6. ファイルの保存場所を指定し、ダウンロードします。
7. 画面上部のメニューより「Product」→「Downloads」→「Community Edition」を選択し、次は、マネージメントコンソール用のモジュールをダウンロードするために、「Management Console」から「Red Hat Enterprise Linux 6 and 7」をクリックします。



8. ファイルの保存場所を指定し、ダウンロードします。
9. その他、必要なモジュールがある場合は、継続してダウンロードを実施します。

OS 設定の実施

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 (8~12GB 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	4096 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)
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。

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。

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。

- Perl 5, chrony, gdb, mcelog, sysstat, tzdata, openssh, which, dialog, bc, libnsl

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
```

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 に行くことで 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
```

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。

```
# service tuned stop  
# chkconfig tuned off
```

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、環境に合わせて赤字部分は変更してください。

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。

```
# /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
```

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 HDD の場合「bfq」、SSD の場合「mq-deadline」に変更します。

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  
[mq-deadline] kyber bfq none
```

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

※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、環境に合わせて赤字部分は変更してください。

2. 手順 1 で、ディスクが HDD の場合「bfq」、SSD の場合「mq-deadline」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。

[HDD の場合]

```
# echo bfq > /sys/block/sda/queue/scheduler  
# echo 'echo bfq > /sys/block/sda/queue/scheduler' >> /etc/rc.local  
# echo bfq > /sys/block/sdb/queue/scheduler  
# echo 'echo bfq > /sys/block/sdb/queue/scheduler' >> /etc/rc.local
```

[SSD の場合]

```
# echo mq-deadline > /sys/block/sda/queue/scheduler  
# echo 'echo mq-deadline > /sys/block/sda/queue/scheduler' >>  
/etc/rc.local  
# echo mq-deadline > /sys/block/sdb/queue/scheduler  
# echo 'echo mq-deadline > /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  
mq-deadline kyber [bfq] none
```

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

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
```

- 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]
```

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。

- 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
```

- 手順 1 で「1」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=1
```

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>

- 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
```

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。

- 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
```

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。

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
```

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+madvice [madvice] 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+madvice madvice [never]
```

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 192.168.56.0/24
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 192.168.56.11 iburst
driftfile /var/lib/chrony/drift
allow 192.168.56.11
local stratum 10
keyfile /etc/chrony.keys
```

4. 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
```

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 (192.168.56.11)
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。

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="resume=UUID=12345678-1234-1234-1234-1234567890ab
rhgb quiet 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。赤字の設定を追記します。

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/centos/grub.cfg
```

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"
```

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。

Vertica の使用ポートが不使用であることの確認

1. 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	プロトコル	サービス
22	TCP	sshd
5433	TCP	Vertica
5434	TCP	Vertica
5433	UDP	Vertica
5444	TCP	Management Console
5450	TCP	Management Console
4803	TCP	Spread
4803	UDP	Spread
4804	UDP	Spread
6543	UDP	Spread

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
```

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

「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。

rpm のインストール

Vertica 10.1.0 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-10.1.0-0.x86_64.RHEL6.rpm
```

install_vertica スクリプトを使っての Vertica のインストール

install_vertica スクリプトを使って、Vertica 10.1.0 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`」オプションを付与してください。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://www.vertica.com/docs/10.1.x/HTML/Content/Authoring/InstallationGuide/InstallingVertica/InstallVerticaScript.htm>

```
# /opt/vertica/sbin/install_vertica --hosts [node#1 の Private IP],[node#2 の Private IP],[node#3 の Private IP] --rpm /tmp/vertica-10.1.0-0.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
```

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。

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
```

データベースの作成

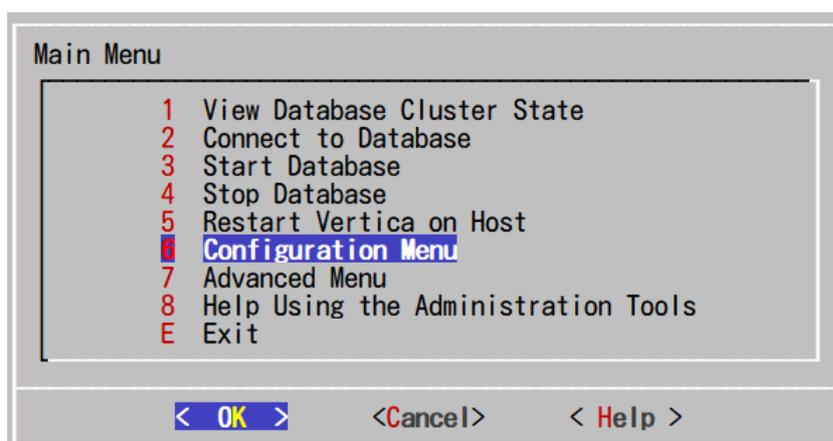
サンプルデータベースの作成

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

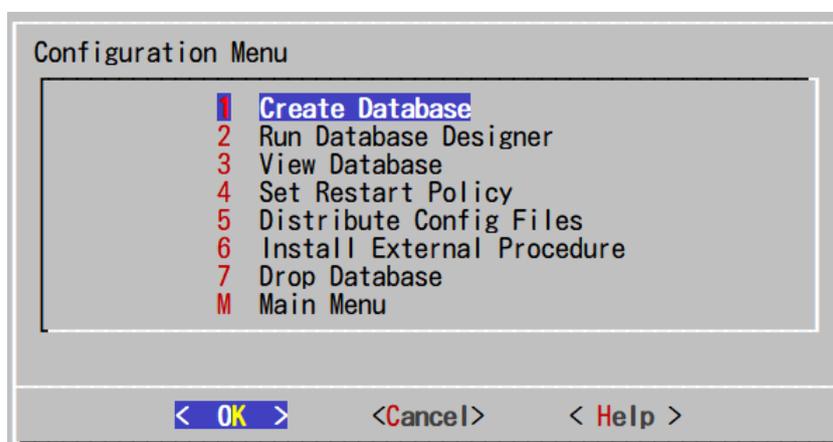
1. dbadmin ユーザーで、ノード#1 にログインします。
2. Administration Tools を起動します。

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

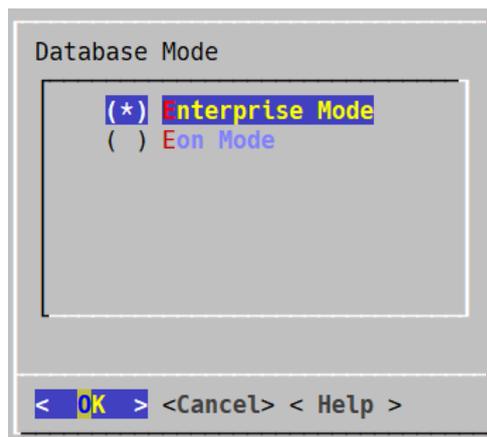
3. 「6 Configuration Menu」 を選択します。



4. 「1 Create Database」 を選択します。

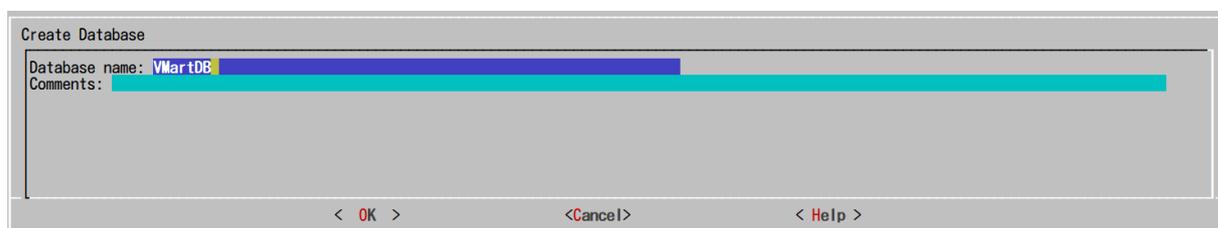


5. 「Enterprise Mode」を選択します。

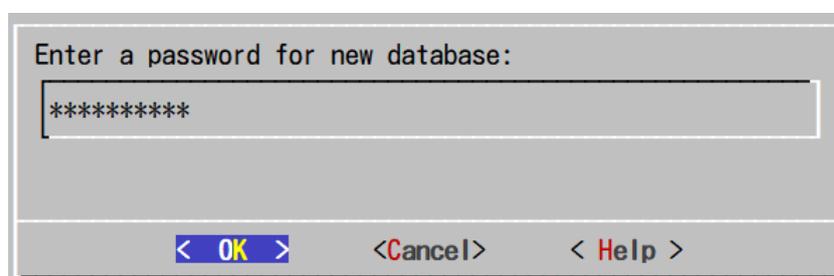


6. 任意のデータベース名を入力します。

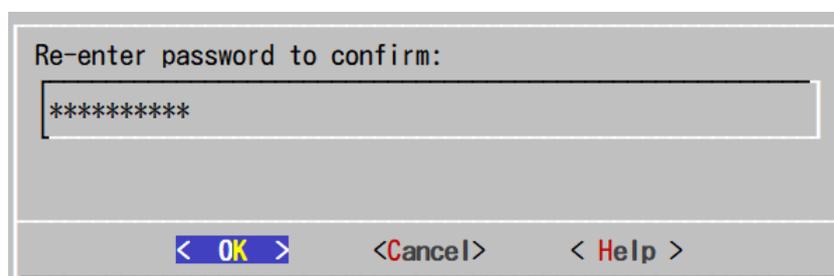
※本例では、「VMartDB」という名前のデータベースを作成しています。



7. 任意のパスワードを入力します。

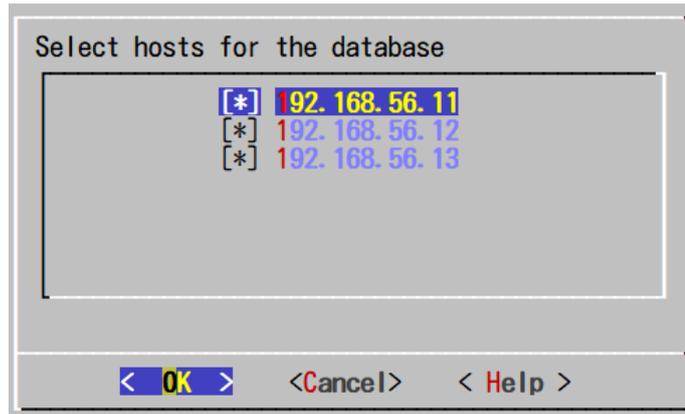


8. 再度、パスワードを入力します。



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。

※下記は、3ノード構成の実行例となります。2ノード構成の場合は、2つのサーバーの表示、1ノード構成の場合は、1つのサーバーの表示となります。

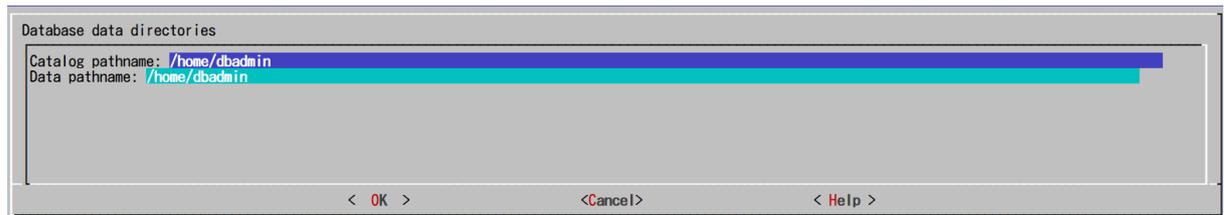


10. カタログファイルとデータファイルを格納するディレクトリを指定します。

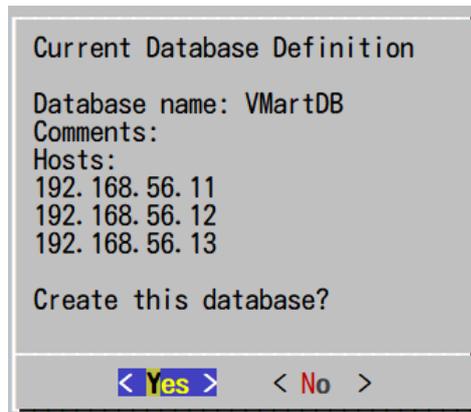
下記の例では、「/home/dbadmin」を指定しています。任意のディレクトリを指定することができますが、データベース作成前に、該当のディレクトリを作成する必要があります。

(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。

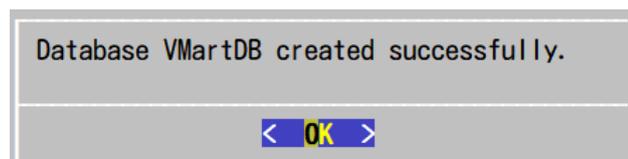


データベースの作成が行われます。

```
*** Creating database: VMartDB ***
  Creating database VMartDB
  Starting bootstrap node v_vmartdb_node0001 (192.168.56.11)
  Starting nodes:
    v_vmartdb_node0001 (192.168.56.11)
  Starting Vertica on all nodes. Please wait, databases with a
  large catalog may take a while to initialize.
```

```
Node Status: v_vmartdb_node0001: (DOWN)
Node Status: v_vmartdb_node0001: (UP)
Creating database nodes
Creating node v_vmartdb_node0002 (host 192.168.56.12)
Creating node v_vmartdb_node0003 (host 192.168.56.13)
Generating new configuration information
Starting all nodes
Start hosts = ['192.168.56.12', '192.168.56.13']
Starting nodes:
    v_vmartdb_node0002 (192.168.56.12)
    v_vmartdb_node0003 (192.168.56.13)
Starting Vertica on all nodes. Please wait, databases with a
large catalog may take a while to initialize.
Node Status: v_vmartdb_node0001: (UP) v_vmartdb_node0002:
(DOWN) v_vmartdb_node0003: (DOWN)
Node Status: v_vmartdb_node0001: (UP) v_vmartdb_node0002: (UP)
v_vmartdb_node0003: (UP)
Installing AWS package
    Success: package AWS installed
Installing ComplexTypes package
    Success: package ComplexTypes installed
Installing MachineLearning package
    Success: package MachineLearning 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。



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 !
```

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。

※vsq1 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「Appendix I : vsq1」を参照してください。

```
$ /opt/vertica/bin/vs1 -f vmart_define_schema.sql -w [指定したパスワード]
```

```
CREATE SCHEMA
CREATE SCHEMA
CREATE TABLE
ALTER TABLE
CREATE TABLE
CREATE TABLE
ALTER TABLE
CREATE TABLE
CREATE TABLE
CREATE TABLE
ALTER TABLE
```

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]
```

データベースの起動と停止

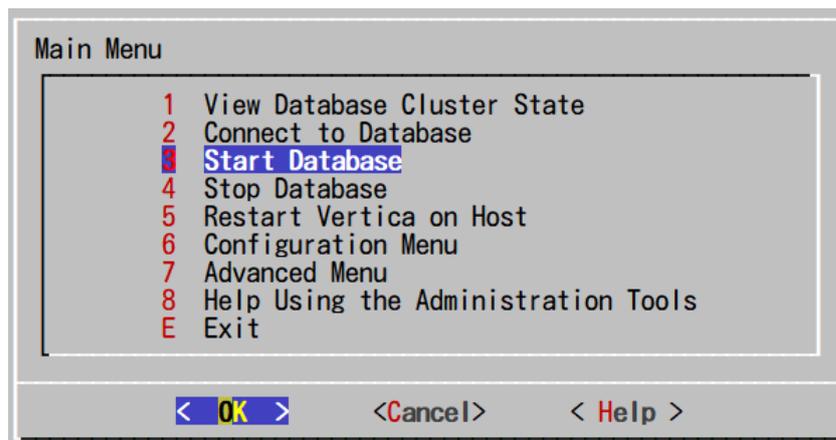
データベースの起動

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

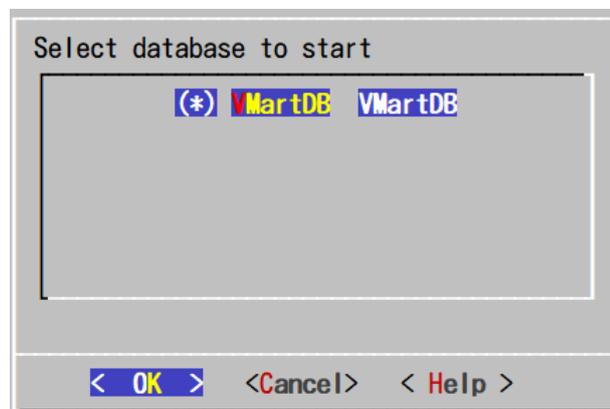
1. dbadmin ユーザーで、ノード#1 にログインします。
2. Administration Tools を起動します。

```
$ /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。



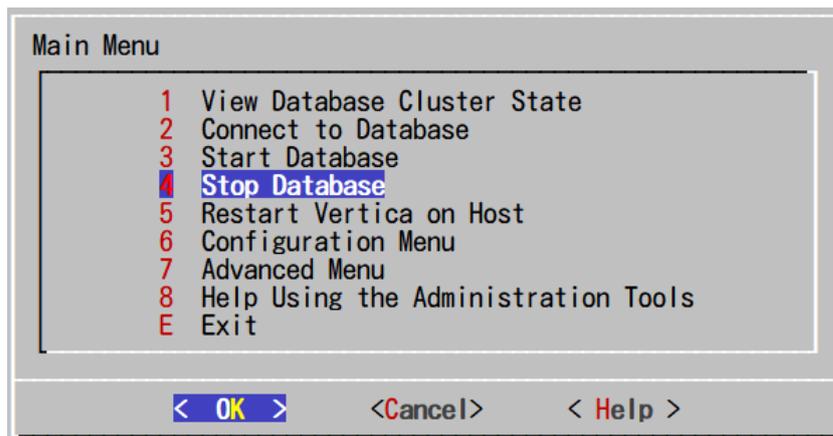
データベースの停止

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

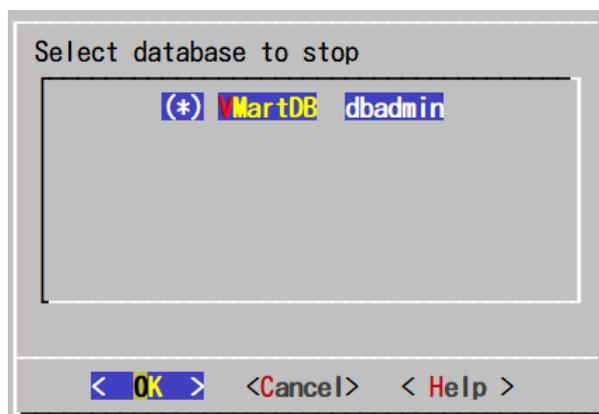
1. dbadmin ユーザーで、ノード#1 にログインします。
2. Administration Tools を起動します。

```
$ /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。



データベースの削除

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。

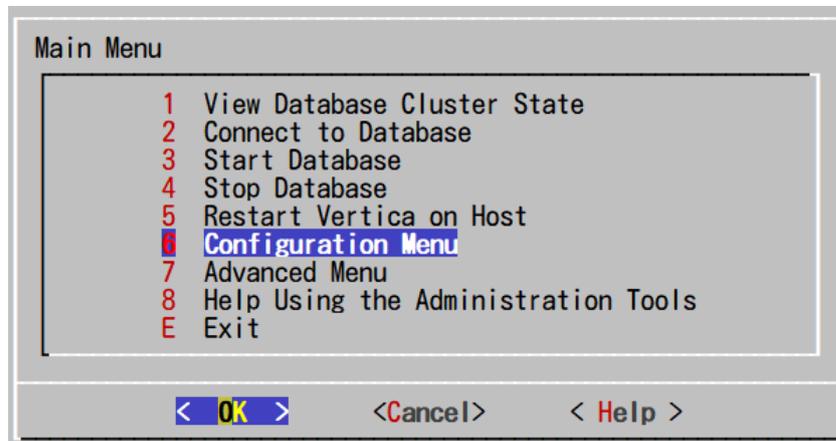
データベースの削除

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

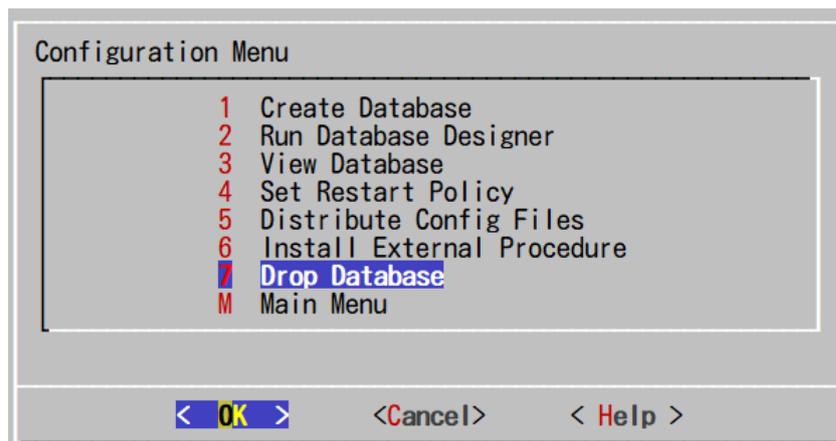
1. dbadmin ユーザーで、ノード#1 にログインします。
2. Administration Tools を起動します。

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

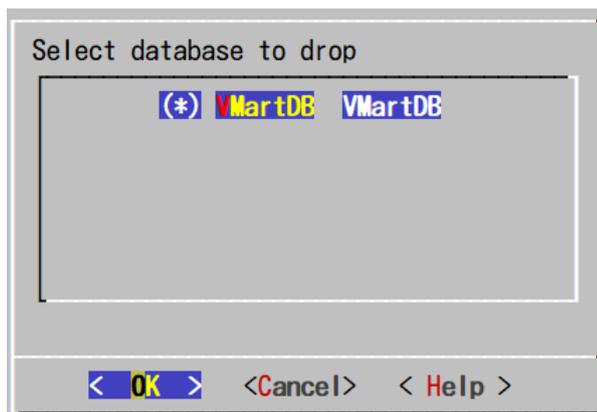
3. 「6 Configuration Menu」 を選択します。



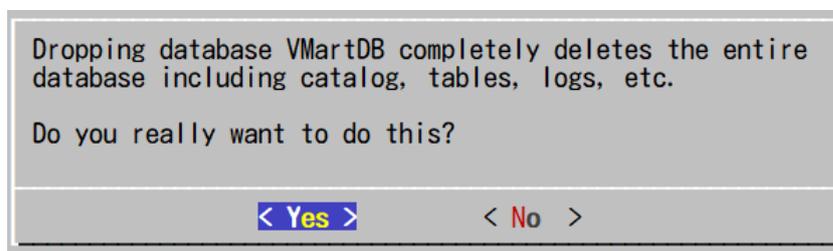
4. 「7 Drop Database」 を選択します。



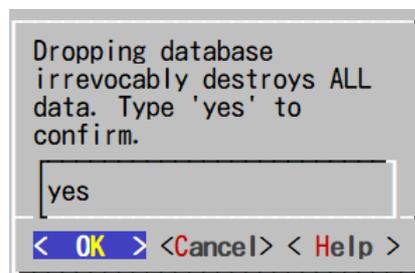
- 削除するデータベースを選択します。



- 確認画面で、「Yes」を選択します。



- 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。



- 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。



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。

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。

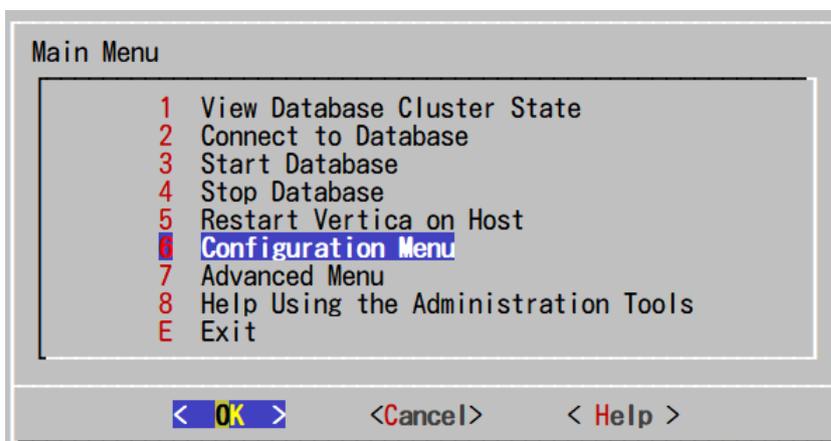
1. dbadmin ユーザーで、ノード#1 にログインします。
2. データベースデザイナーのログファイル等の出力ディレクトリを作成し、作成したディレクトリに移動します。

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

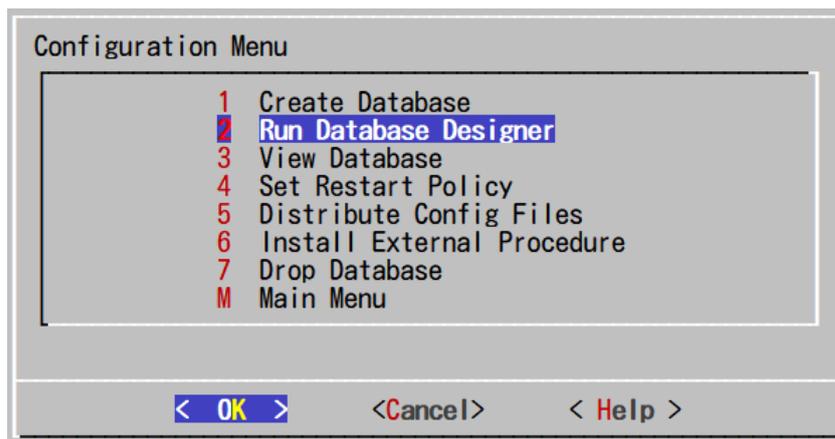
3. Administration Tools を起動します。

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

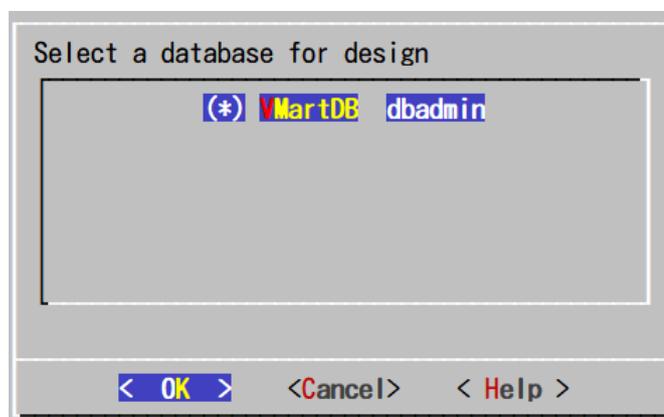
4. 「6 Configuration Menu」を選択します。



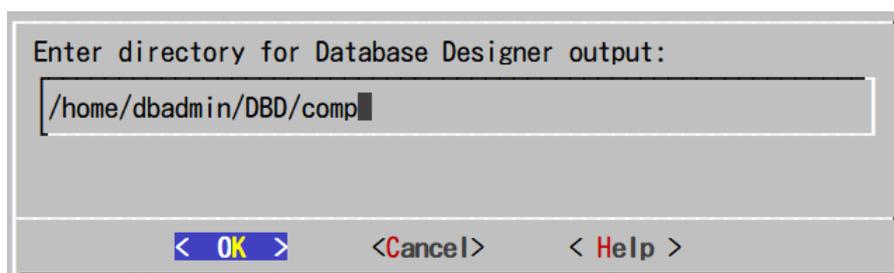
5. 「2 Run Database Designer」を選択します。



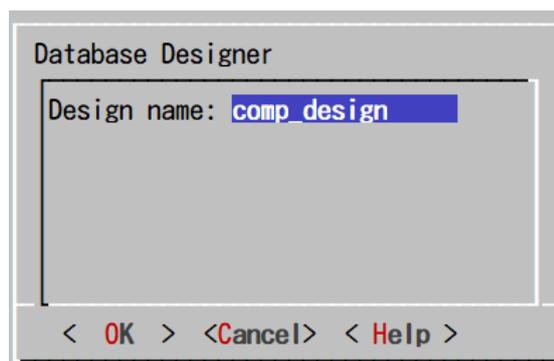
6. データベースデザイナーを実行するデータベースを選択します。



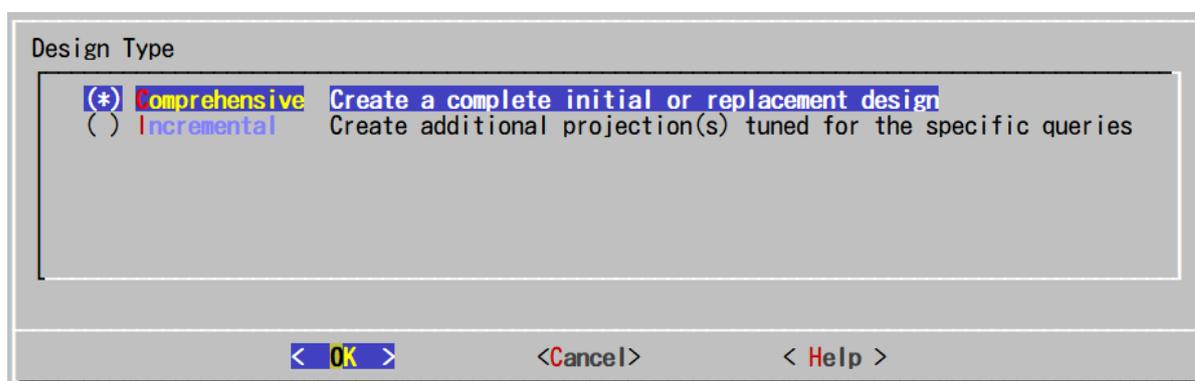
7. データベースのパスワードを入力します。
8. データベースデザイナーのアウトプットの出力先ディレクトリを指定します。



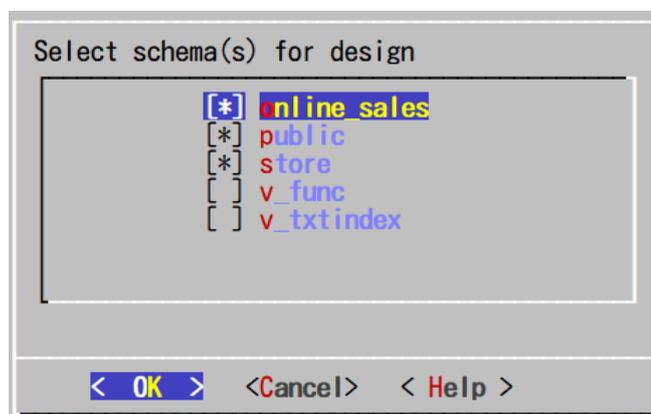
9. 任意のデザイン名を入力します。



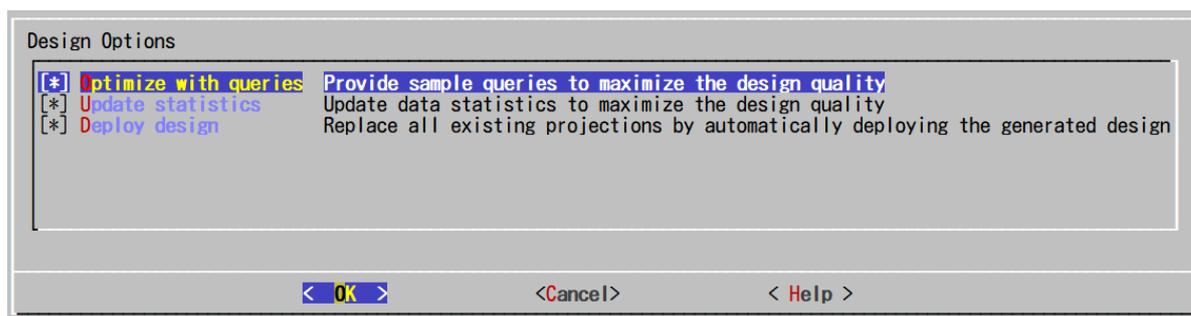
10. Design Type に「Comprehensive」を選択します。



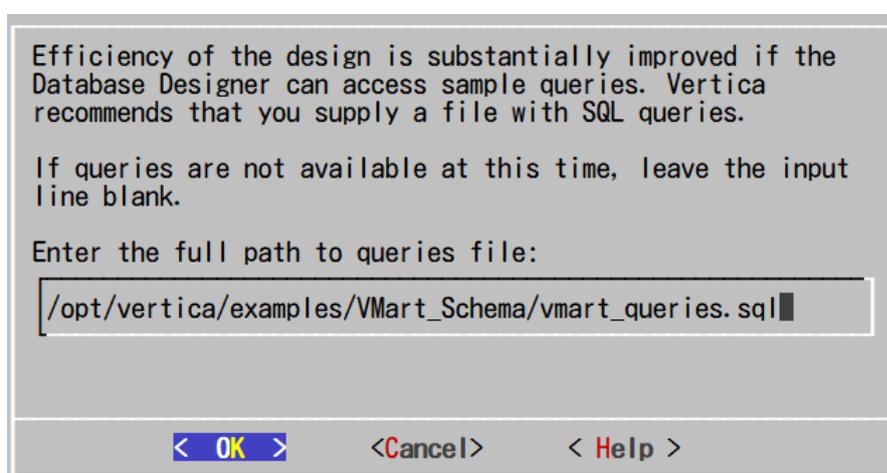
11. データベースデザイナーを実行する対象のスキーマを選択します。(サンプルスキーマを使用している場合は、「online_sales」「public」「store」を選択します。)



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。



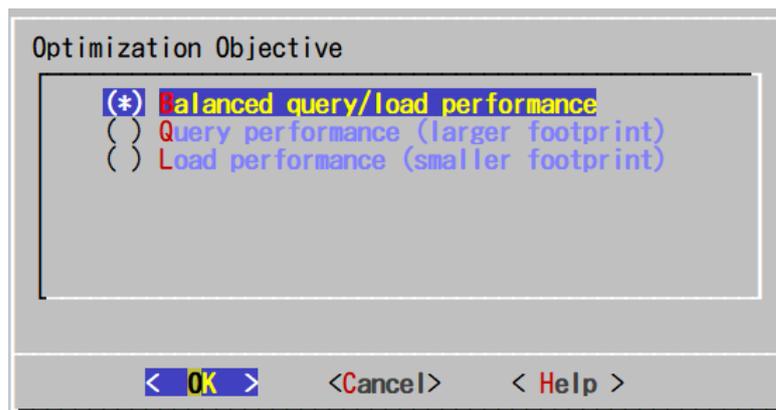
14. K-safety の値を「1」と指定します。

※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://www.vertica.com/docs/10.1.x/HTML/Content/Authoring/ConceptsGuide/Components/K-Safety.htm>



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.
```

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。

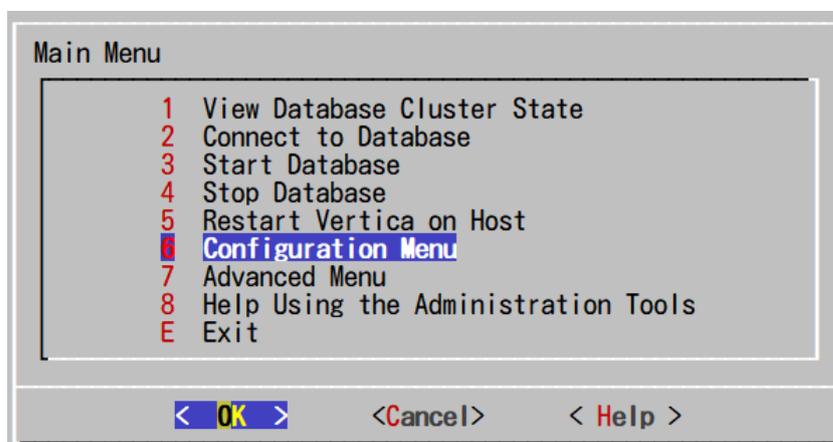
1. dbadmin ユーザーで、ノード#1 にログインします。
2. データベースデザイナーのログファイル等の出力ディレクトリを作成し、作成したディレクトリに移動します。

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

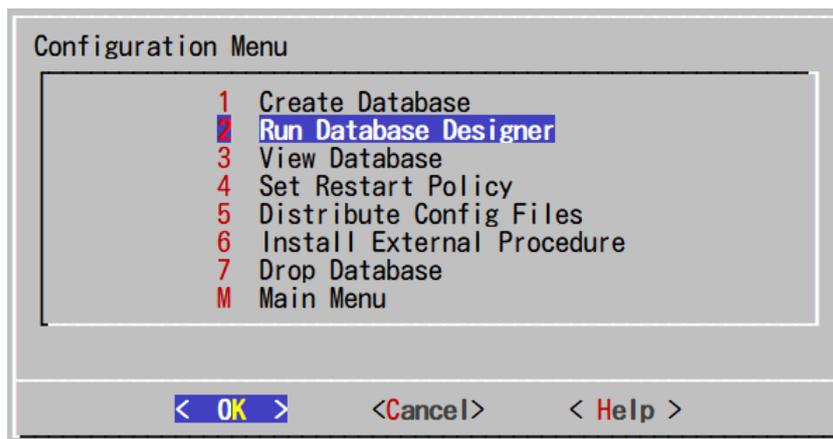
3. Administration Tools を起動します。

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

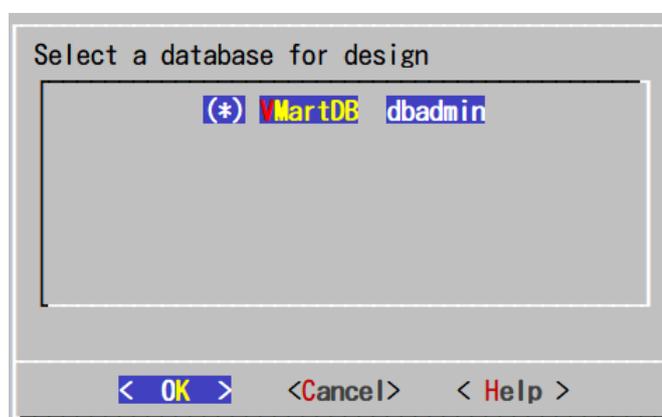
4. 「6 Configuration Menu」を選択します。



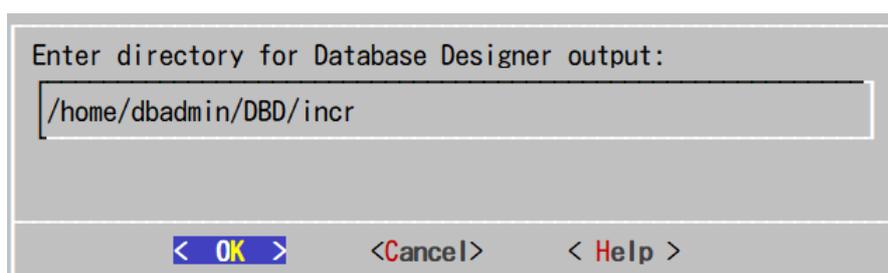
5. 「2 Run Database Designer」を選択します。



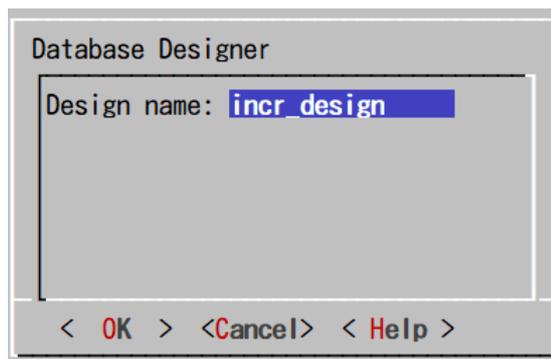
6. データベースデザイナーを実行するデータベースを選択します。



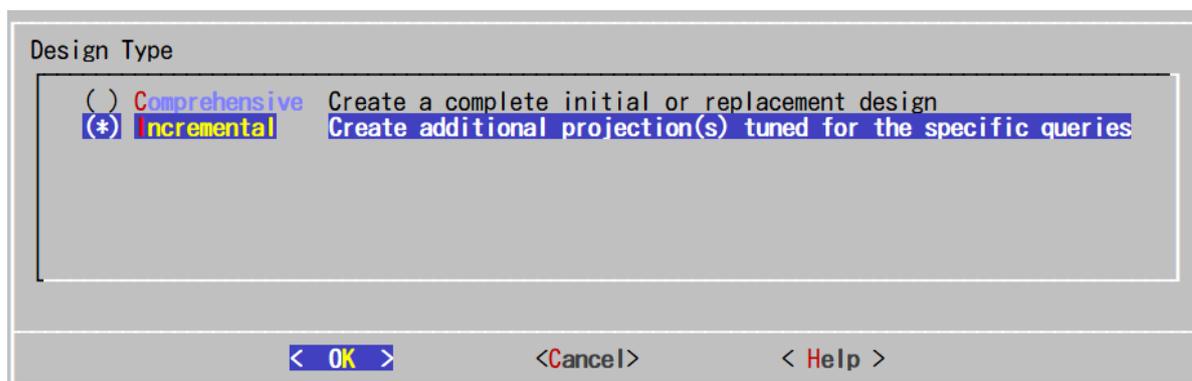
7. データベースのパスワードを入力します。
8. データベースデザイナーのアウトプットの出力先ディレクトリを指定します。



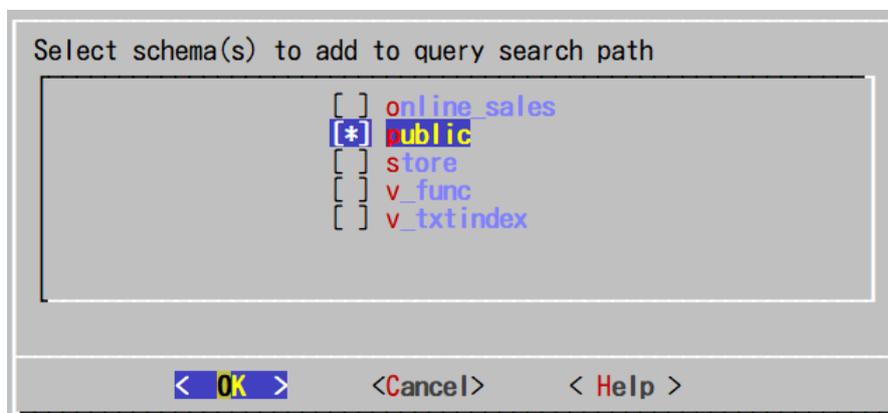
9. 任意のデザイン名を入力します。



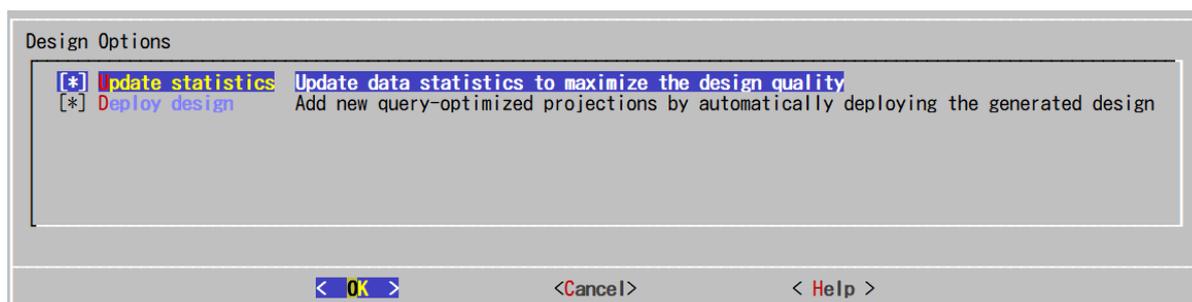
10. Design Type に「Incremental」を選択します。



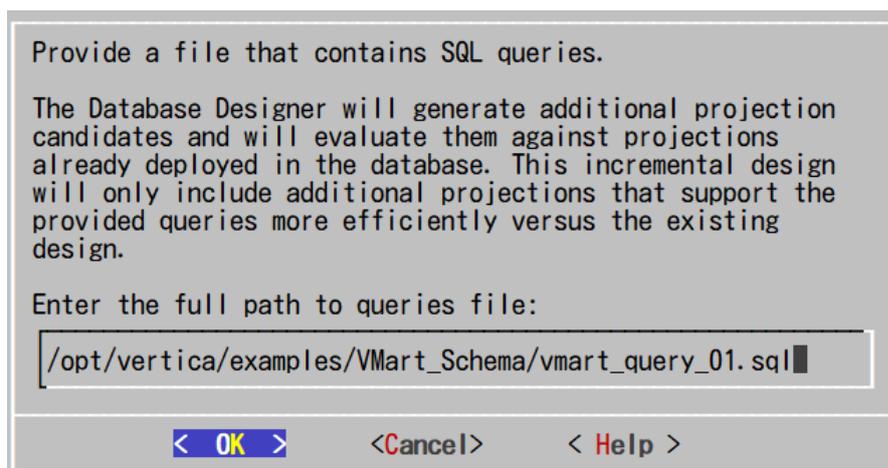
11. データベースデザイナーを実行する対象のスキーマを選択します。（最適化したいクエリが参照するスキーマを選択します。）



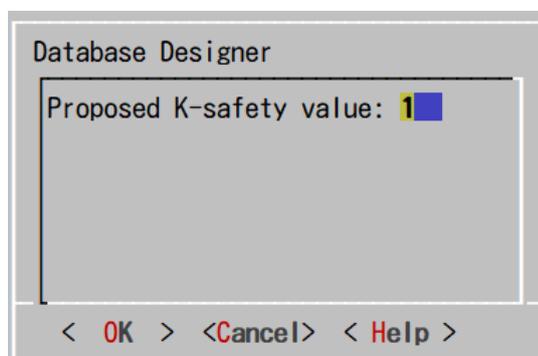
12. 全てのオプションをチェックします。



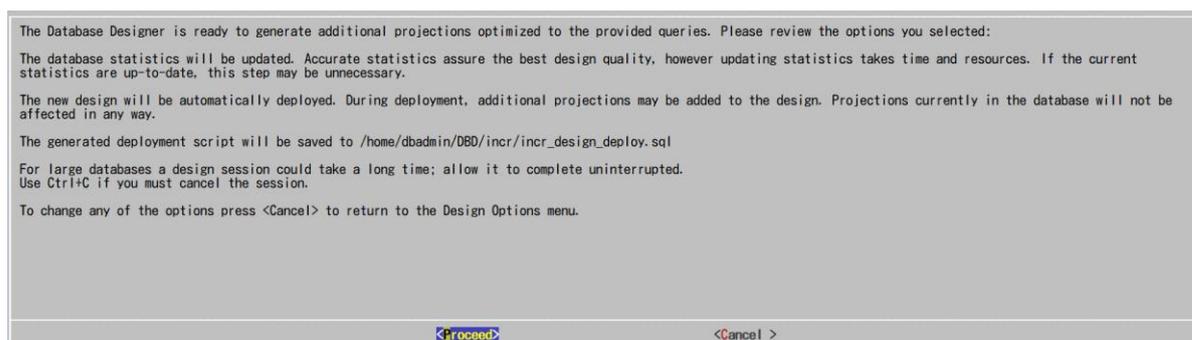
13. 最適化したいクエリファイルを指定します。



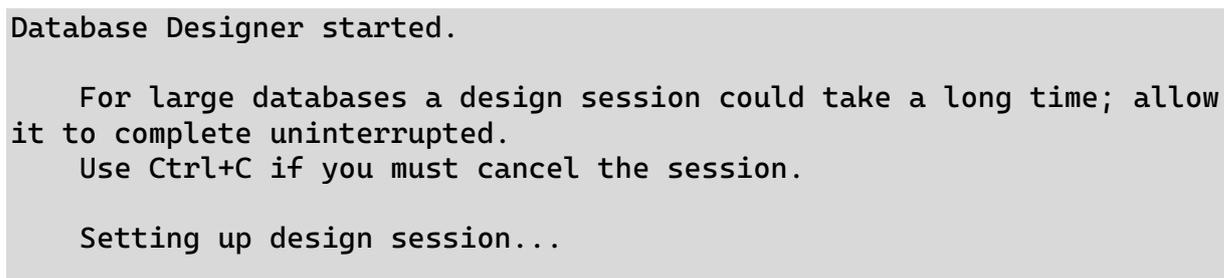
14. K-safety の値を「1」と指定します。



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。



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.

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#1 上にインストールする方法をご紹介します。

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#1 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
```

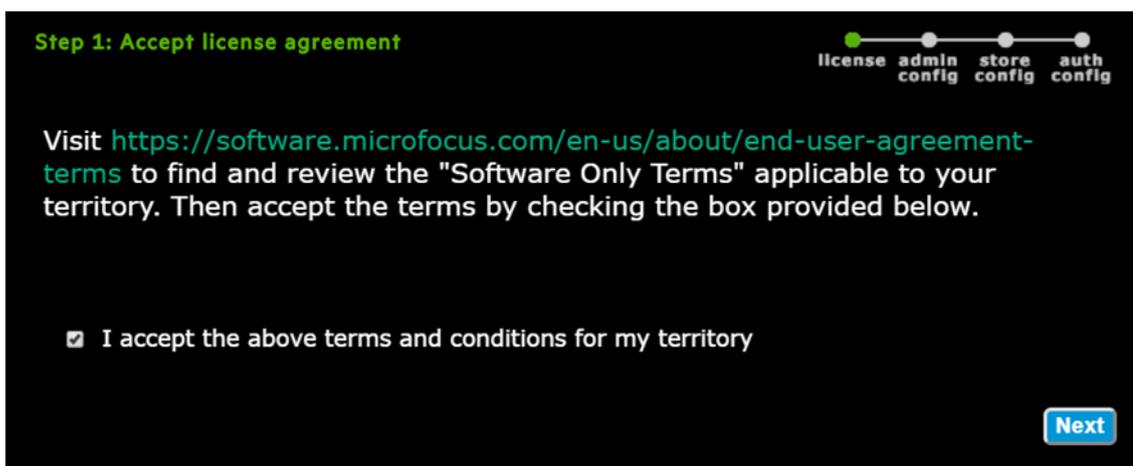
3. rpm をインストールします。

```
# dnf install vertica-console-10.1.0-0.x86_64.RHEL6.rpm
```

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://\[ノード#1のPublicネットワークのIP\]:5450/webui](https://[ノード#1のPublicネットワークのIP]:5450/webui)」に接続します。
10.1 の MC がサポートしているブラウザは以下の通りです。
 - Internet Explorer 11
 - Firefox
 - Chrome
2. License agreement にチェックを入れ、「Next」をクリックします。



- 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。

Step 2: Configure Management Console

license admin store auth
config config config config

Username: uidbadmin

Password:

Confirm password:

Unix group name: verticadba

Home directory: /home/uidbadmin

License directory: /home/uidbadmin

Management Console port: 5450

Master Passphrase: verticamanagementconsole

Use system auto generated

Previous Next

- セキュリティ質問を3つ選択し、それぞれ回答を入力します。これは、パスワードリセットの際に利用されます。

Step 3: Configure Security Questions

license admin Security store auth
config questions config config config

Security Question 1: Select Question 1

Answer 1: Enter Answer

Security Question 2: Select Question 2

Answer 2: Enter Answer

Security Question 3: Select Question 3

Answer 3: Enter Answer

Previous Next

5. 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。

Step 3: Configure storage locations

license admin store auth
config config config config

Catalog path:

Data path:

Previous Next

6. 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。

Step 4: Configure authentication

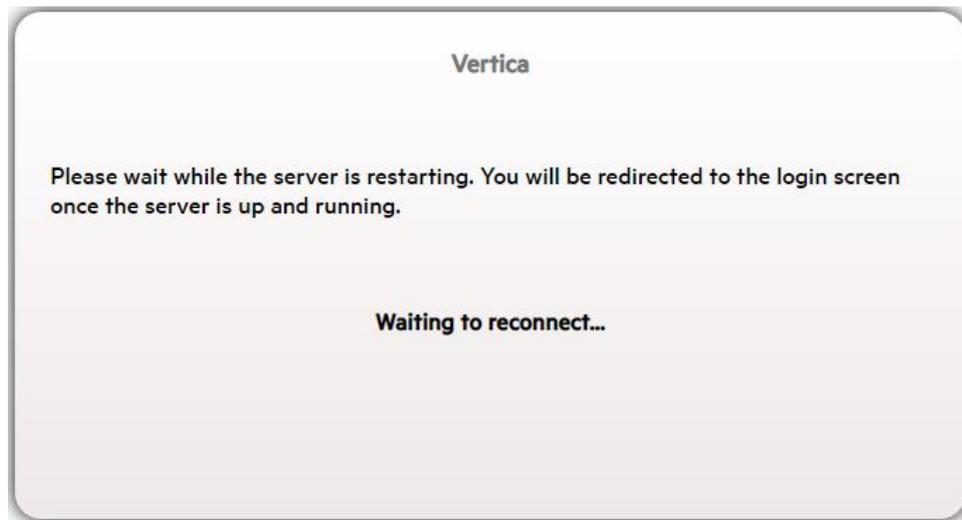
license admin store auth
config config config config

Use Management Console for authentication

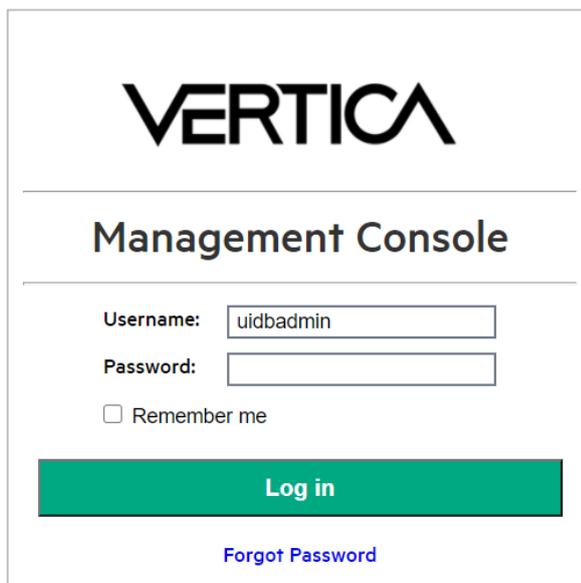
Use LDAP for authentication

Previous Finish

7. 設定が保存され、MC サービスが再起動されます。

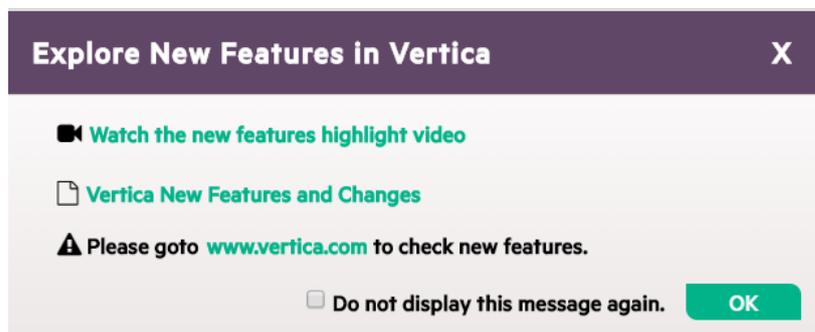


8. 再起動が終わると、ログイン画面に変わります。管理者ユーザーの情報を入力し、「Log in」をクリックします。

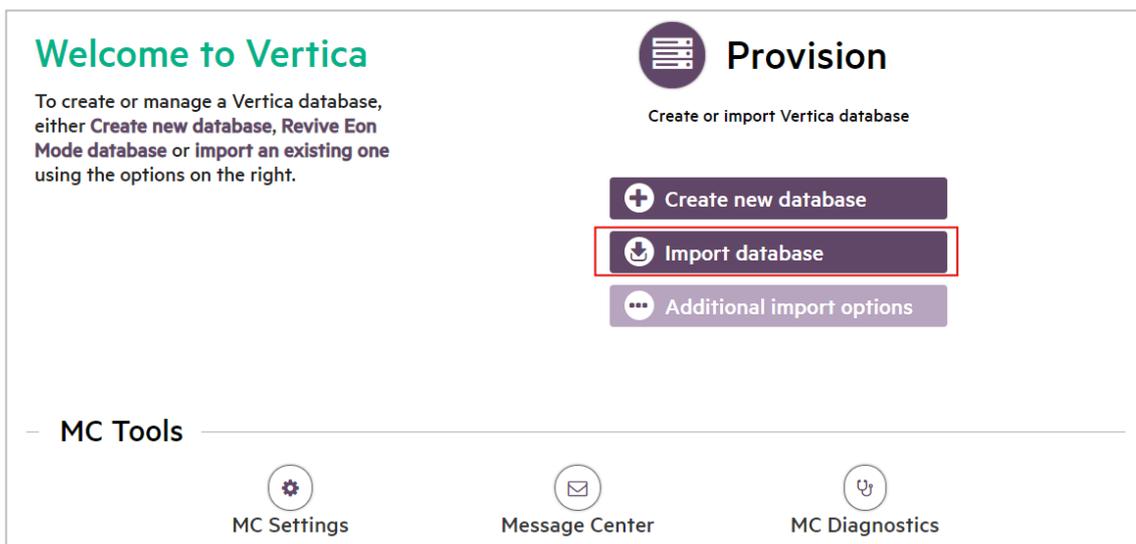


The image shows the Vertica Management Console login interface. At the top is the 'VERTICA' logo. Below it is the title 'Management Console'. There are two input fields: 'Username:' with the value 'uidbadadmin' and 'Password:'. Below the password field is a checkbox labeled 'Remember me'. A large green button labeled 'Log in' is centered below the fields. At the bottom, there is a blue link labeled 'Forgot Password'.

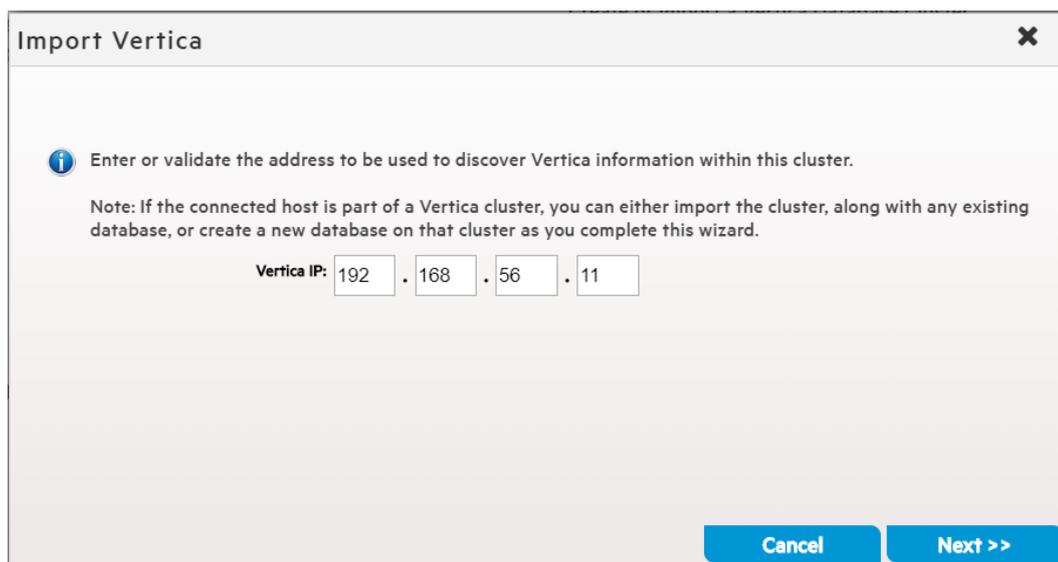
9. 新機能を紹介するウィンドウが表示されます。必要に応じて参照してください。「OK」をクリックして、ウィンドウを閉じます。



10. 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。



11. 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。



Import Vertica

i Enter or validate the address to be used to discover Vertica information within this cluster.

Note: If the connected host is part of a Vertica cluster, you can either import the cluster, along with any existing database, or create a new database on that cluster as you complete this wizard.

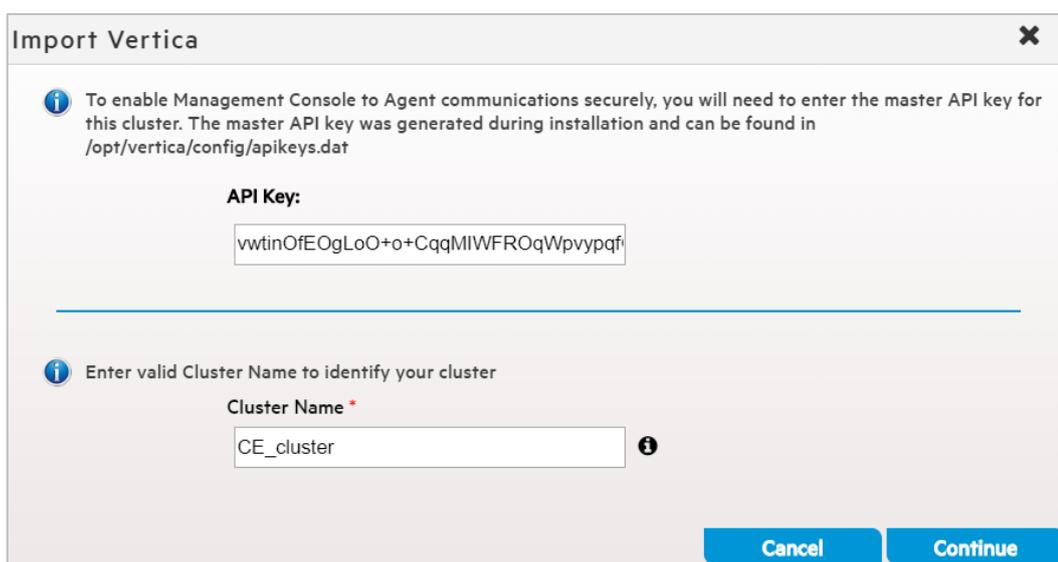
Vertica IP: . . .

Cancel **Next >>**

12. 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": "vwTinOfEOgLoO+o+CqqMIWFROqWpvypqfO+rOw",  
    "app": "vertica",  
    "level": "admin",  
    "requestor": "master"  
  }  
]
```

13. 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。



Import Vertica

i To enable Management Console to Agent communications securely, you will need to enter the master API key for this cluster. The master API key was generated during installation and can be found in /opt/vertica/config/apikeys.dat

API Key:

i Enter valid Cluster Name to identify your cluster

Cluster Name *

i

Cancel **Continue**

- 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。

Import Vertica

Cluster discovered with the information below.
Vertica Version: 10.1.0 Hosts: 3

Deselect the databases that you don't want to import with the cluster. Click Cancel if you don't want to import the cluster and the databases.

Name	Status	Nodes	Username	Password	Use TLS Connection
<input checked="" type="checkbox"/> VMartDB	UP	3	dbadmin	*****	<input type="checkbox"/> Use TLS

Username and password are required for the database that is up and running.

Cancel Import

- 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。

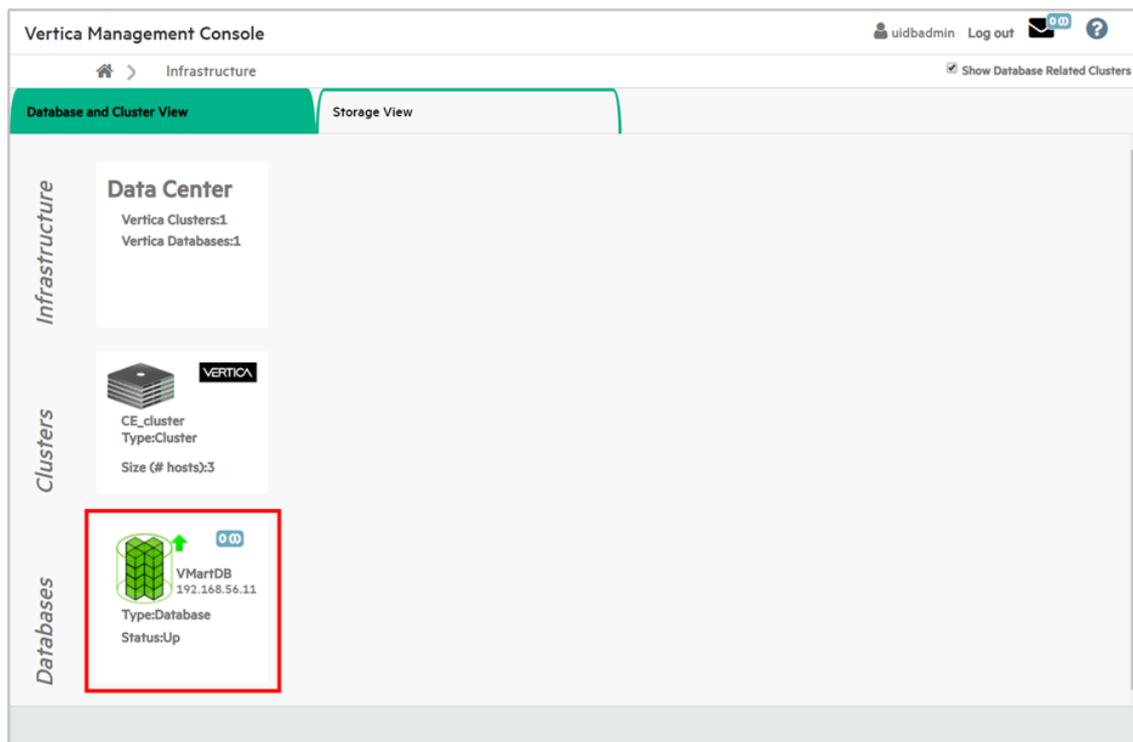
Import Vertica

You have successfully imported the cluster.

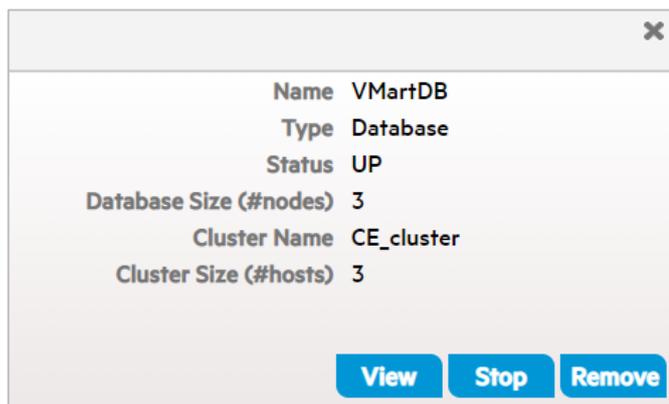
"VMartDB" is running on this cluster. You must shut down "VMartDB" before you can create another database cluster.
The default time for the Work Load Analyzer to run is 2:00 AM. To change this default go to [MC Setting -> Monitoring](#).

Done

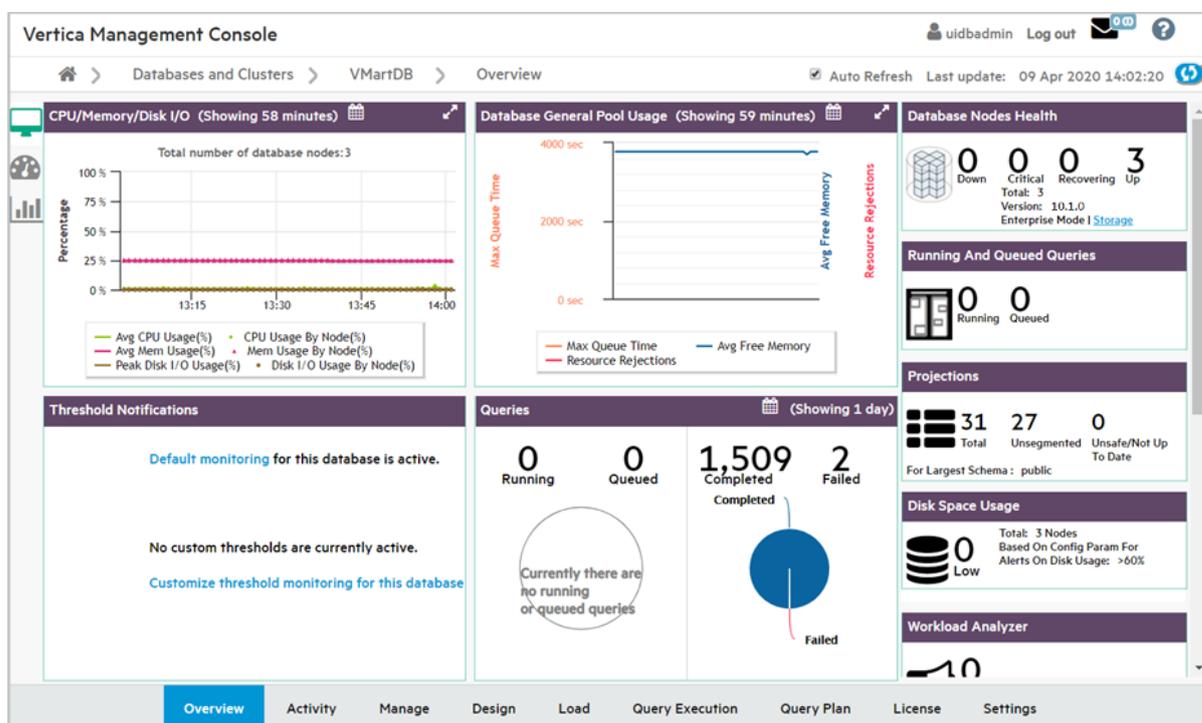
16. データベースをクリックします。



17. 「View」 をクリックします。



18. データベースの情報が表示されることを確認します。



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、弊社問い合わせ先 (jpn_vertica_info@microfocus.com) まで、下記のような件名で、メールでお問い合わせください。

- 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いますので、そちらもあわせてご活用ください。

- <https://www.vertica.com/>

2021 年 3 月吉日

Appendix I: vsql

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 Appendix 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。

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。

Administration Tools からの起動

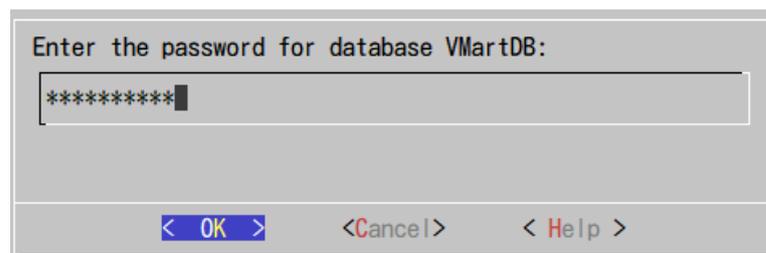
1. dbadmin ユーザーで、ノード#1 にログインします。Administration Tools を起動します。

```
$ /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。



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。

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