

# Vertica 24.1 新機能紹介

# サポートプラットフォーム

## Red Hat Enterprise Linux 9のサポート

---

サポートプラットフォームにRed Hat Enterprise Linux 9が追加。

## Cloud: プラットフォームの変更

---

AWS, Azure, GCPで公開する24.1のイメージが使用するプラットフォームがRed Hat Enterprise Linux 8に変更。

# インストール

## RPMファイルの変更

---

Red Hat Enterprise Linux向けのRPMファイルが以下の通り変更となる。

vertica-24.1.0-0.[RHEL6](#).x86\_64.rpm → vertica-24.1.0-0.[RHEL8](#).x86\_64.rpm

# クライアント接続

## Workload Routing

Workloadに応じて特定のSubclusterにクライアント接続をルーティングする機能。以下が本バージョンでの変更点。

- ユーザおよびロールに基づいたWorkload Routing

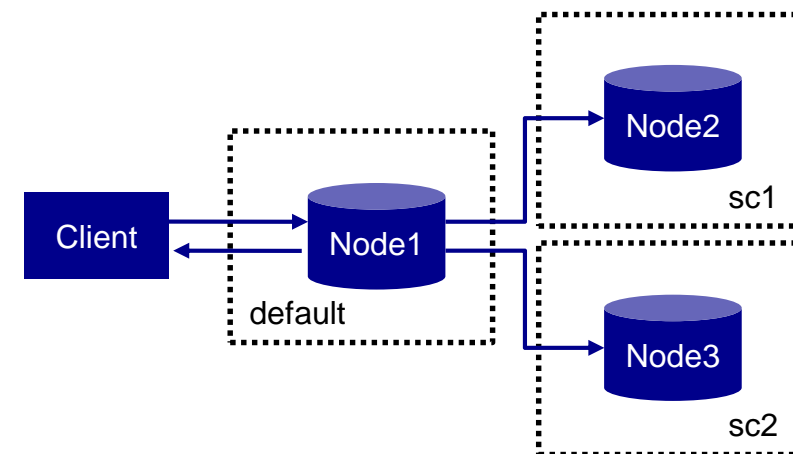
WorkloadのUSAGE権限をユーザまたはロールに付与できる。

```
=> CREATE ROUTING RULE ROUTE WORKLOAD analytics  
-> TO SUBCLUSTER sc1, sc2;  
=> GRANT USAGE ON ROUTING RULE analytics TO analytics_user_1;
```

- 優先度の指定

複数のWorkloadが使用できる場合、ランダムで使用するものを選択する。  
優先度を指定することで、使用するWorkloadの順序づけを行うことができる。

```
=> GRANT USAGE ON WORKLOAD analytics TO analytics_role PRIORITY 1;  
=> ALTER ROUTING RULE FOR WORKLOAD reporting SET PRIORITY TO 2;
```



# クライアント接続

## Workload Routing (続き)

---

Workloadに応じて特定のSubclusterにクライアント接続をルーティングする機能。以下が本バージョンでの変更点。(続き)

- 使用できるWorkloadの表示

ユーザ自身が使用できるWorkloadを表示することのできるSHOW AVAILABLE WORKLOADS文を実装。

```
=> SHOW AVAILABLE WORKLOADS;  
  
      name | setting  
-----+-----  
available workloads | reporting, analytics
```

# Client Driver

## ADO.NET: Windows Installerからのインストールを再開

---

23.4のWindows Client Driver InstallerからADO.NET Driverが削除され、NuGetからのインストールのみをサポートしていた。一部アプリケーションではNuGetからインストールされるDriverを認識できない制約があるため、本バージョンではWindows InstallerからのADO.NET Driverのインストールを再度サポートする。

## ADO.NET: Read-onlyファイルシステムのサポート

---

KubernetesなどのRead-onlyファイルシステムを用いた場合でも動作するよう、以下の内容を変更。

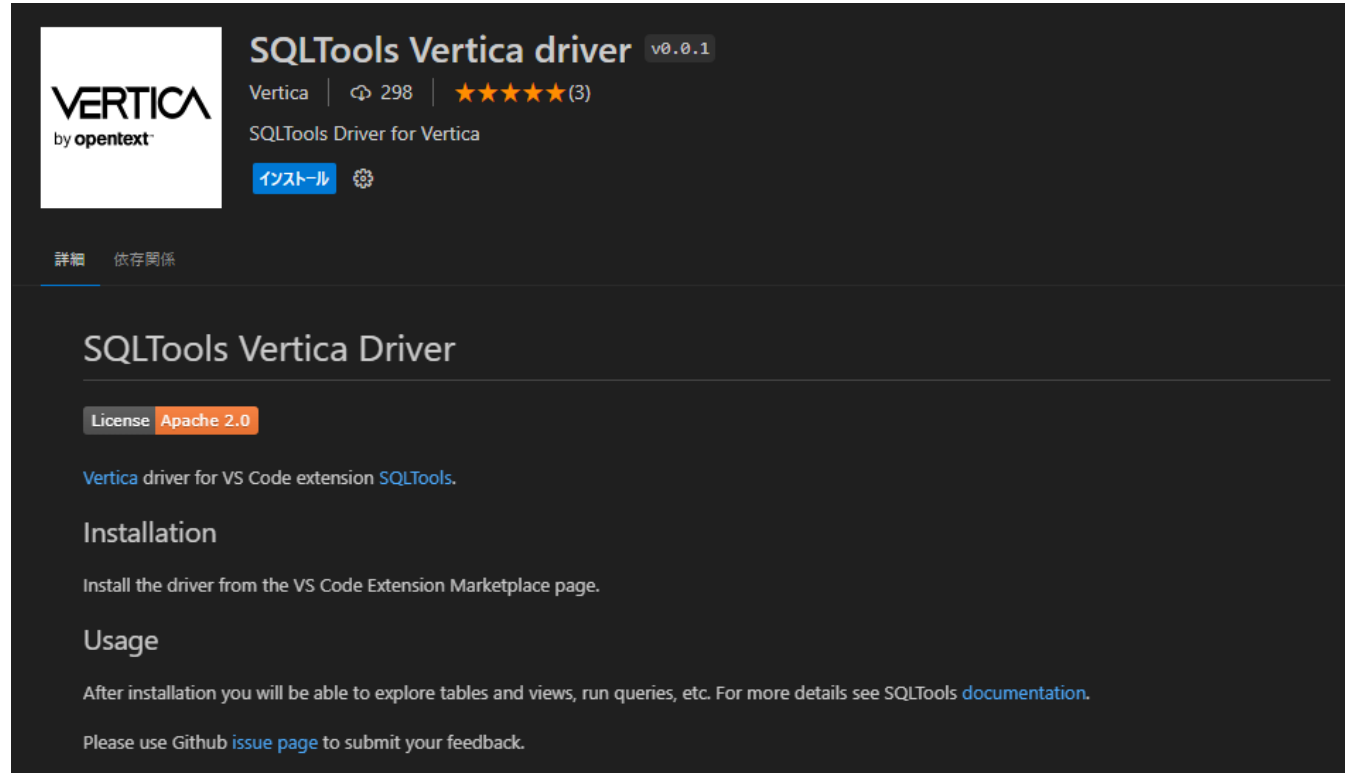
- 設定ファイルが存在しない場合でも新しい設定ファイルを自動的に作成しない。
- Windowsレジストリを変更したり、設定値を読み込むことをしない。
- ホームディレクトリまたはプロジェクトディレクトリに存在する設定ファイルを使用する。
- VerticalLogPropertiesクラスの以下のメソッドでログ出力の挙動を指定でき、アプリケーション終了まで維持する。
  - SetLogPath(String path)
  - SetLogNamespace(String lognamespace)
  - SetLogLevel(VerticalLogLevel loglevel)

# Client Driver

## Visual Studio Code extension SQLTools向けCommunity Driver

---

SQLToolsからVerticaに接続するためのDriverをVisual Studio Code Extension Marketplaceで公開。



The screenshot shows the Visual Studio Code Extension Marketplace page for the "SQLTools Vertica driver" extension. The extension is developed by "opentext" and is currently at version "v0.0.1". It has 298 downloads and a rating of 3 stars. The page includes an "インストール" (Install) button and a gear icon for settings. Below the header, there are tabs for "詳細" (Details) and "依存関係" (Dependencies). The main content area is titled "SQLTools Vertica Driver" and includes the following information:

- License:** Apache 2.0
- Description:** Vertica driver for VS Code extension SQLTools.
- Installation:** Install the driver from the VS Code Extension Marketplace page.
- Usage:** After installation you will be able to explore tables and views, run queries, etc. For more details see SQLTools [documentation](#).
- Feedback:** Please use Github [issue page](#) to submit your feedback.

# Data Collector

## ポリシー設定関数追加導入

---

以前まではSET\_DATA\_COLLECTOR\_POLICY関数とSET\_DATA\_COLLECTOR\_TIME\_POLICY関数を用いて個々のData Collector Componentに対してデータ保持ポリシーを設定していた。

本バージョンより、[SET\\_DATA\\_COLLECTOR\\_POLICY\(USING PARAMETERS\)](#)関数を用いることで必要な設定を一度に行うことができる。

指定できるパラメータは次の通り。

- memKB
- diskKB
- synchronous
- retention



# データロード

## Data Loader

---

Data Loaderの実行時に指定できるパラメータを追加実装。

```
EXECUTE DATA LOADER [schema.]name  
  [ WITH FILES file [{ EXACT_PATH | EXPAND_GLOB }][, ...] ]  
  [ FORCE RETRY ]  
  [ BATCH_SIZE { 'size{K|M|G|T}' | 0 | UNLIMITED } ]
```

- 特定のファイルのみを対象としてロード

COPY句に指定されたパスにあるファイルの中から特定のファイルのみをロードさせることができるWITH FILES句を追加。

- 対象ファイルの並列ロード

データサイズと実行ノード数に応じて対象ファイルをバッチに分け、同時並列ロードを行う。この際、1つのバッチの最大処理サイズをBATCH\_SIZE句を用いて指定することができる。

すべてのData Loaderの処理状況を確認できるDATA\_LOADER\_EVENTSシステムを提供。

# データロード

## IcebergテーブルのFallback Name Mappingサポート

---

Icebergはカラム情報を提供するためにフィールドIDを使用するが、対象のParquetファイルがIDを持たないケースもあり得る。その場合、Icebergは[Fallback Name Mapping](#)を使用してカラム情報を提供する。

ただし、Verticaの以前のバージョンではFallback Name Mappingを基にしたカラム情報を利用できなかったが、本バージョンからはそれをサポートする。これはVertica内部で自動的に行われるので、ユーザ側で変更の必要な設定はない。

# データベース管理

## ログローテーションサービス

---

以前はLinuxのlogrotateコマンドを用いてログローテーションを行っていた。

本バージョンでLogRotateサービスを提供し、Linuxコマンドを使用せずにログローテーションを行う。

以下の設定パラメータを用いて動作設定を行う。

- EnableLogRotate: LogRotateサービスの有効・無効の設定。デフォルト値は1(有効)。
- LogRotateInterval: ローテーションを行う時間間隔。デフォルト値は8h(8時間毎)。
- LogRotateMaxAge: ログファイルの保持期間。デフォルト値は7days。
- LogRotateMaxSize: 1つのログファイルの最大サイズ。デフォルト値は100M。

`do_logrotate_local`ファンクションを用いて、接続ノード上でログローテーションを手動実行することも可能。

旧バージョンからアップグレードした環境の場合、従来のLinux logrotateコマンド向けの設定ファイルが存在する場合にはLogRorateサービスは処理を行わない。

# データベース管理

## S3ストレージへの書き出しパフォーマンスの向上

---

VerticalはS3ストレージへのファイル書き出しをシングルスレッドを用いて行っているが、1回の書き出しに複数のファイルが含まれることが多く見られる。書き出し処理を並列実行させるために各ノードにスレッドプールを設け、すべてのS3書き出し処理で共有させる。

`ObjStoreUploadParallelism`設定パラメータを用いてスレッドプールの大きさを指定できる。デフォルト値は0で、従来通りのシングルスレッドによる書き出し処理となる。同時書き出しを行う場合、1つの書き出し処理に1~2つのスレッドを割り当てると最も効率的にパフォーマンスを向上させることができる。あらかじめ大きなプールを用意することもできる(最大1,000スレッド)が、多くのスレッドは多くのメモリを消費する。

## Node Management Agent, HTTPSサービスのエラーメッセージの統一

---

Node Management AgentおよびHTTPSサービスが返すエラーメッセージをRFC7807で定義されているメッセージに統一する。

# SQLステートメント・ファンクション

## Directed Queryの状況確認

---

`DIRECTED_QUERY_STATUS`システムテーブルを用いてDirected Queryの使用状況(データベース起動後の実行回数)を確認できる。

`CLEAR_DIRECTED_QUERY_USAGE`ファンクションで使用状況をクリアできる。

## ALTER TABLE ... ADD COLUMNによる複数カラム指定のサポート

---

ALTER TABLE ... ADD COLUMNで追加できるカラムが1つのみから複数指定可能となる。

```
ALTER TABLE table ADD COLUMN column1 datatype, column2 datatype, column3 datatype, ...;
```

# Eon Mode

## Namespaceのサポート

Schemaの上位となるNamespaceをサポートする。Namespace毎にShard数を指定することができる。

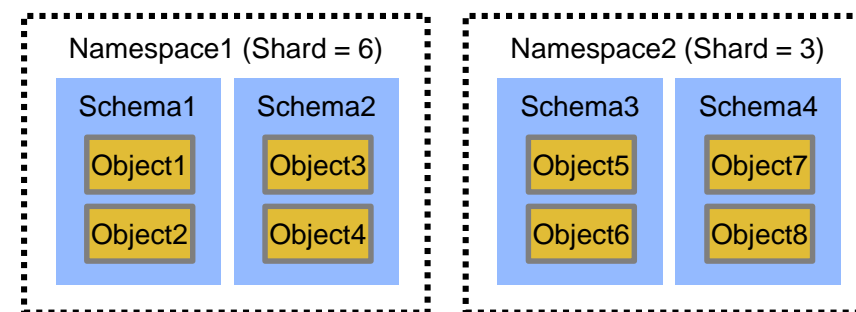
Namespaceの導入により、以下の利点がある。

- マルチテナントのデータベースにおいて、それぞれのユーザのデータを異なるNamespaceに格納する。
- 大きなファクトテーブルと小さなディメンションテーブルのように、それぞれのテーブルはカラム数、データ量、セグメンテーションのやり方が異なるため、1つのShard数では最適化を行う限界があり、複数のShard数を用いて最適化する。
- Shard数の最適化により、Catalogサイズの抑制が期待できる。

デフォルトのNamespaceとして `default_namespace` が作成される。

オブジェクトの指定は `<namespace>.<schema>.<object>` となり、Namespaceを省略すると `default_namespace` が使用される。

```
=> CREATE NAMESPACE analytics SHARD COUNT 3;
=> SELECT * FROM namespaces;
 namespace_oid | namespace_name | is_default | default_shard_count
-----+-----+-----+-----
 45035996273704988 | default_namespace | t          | 6
 45035996273854502 | analytics       | f          | 3
```



Reshardは `default_namespace`以外のNamespaceが存在する場合は実行できない。

# Management Console

## データベース作成時・Revive時に指定するS3 Bucketに対する追加条件

---

Management Consoleからデータベースを作成、またはReviveする際、使用するS3 BucketはCloudFormationのテンプレートを用いてデプロイする際に指定したS3 Bucketであることを条件として追加。

# Containers・Kubernetes

## VerticaDB operator 2.0.0

---

本バージョンを利用するKubernetesイメージではVerticaDB operator 2.0.0が使用され、以下の変更が行われる。

- VerticaDB operatorがvcluster-opsライブラリを使用し、クラスタ管理をNode Management AgentおよびHTTPSサービスを用いて行う。
- admintoolsおよびSSH接続用キーファイルがKubernetesイメージから削除される。
- VerticaDB Custom Resource Definitionの複数のパラメータが変更される。
- scrutinizeはvclusterコマンドで収集する。

## Google Secret Managerのサポート

---

VerticaDB operatorはGoogle Secret Managerに保管されたSecretを使用できるようになる。



# Containers・Kubernetes

## anyuid Security Context Constraintのサポート

---

Red Hat OpenShiftで事前定義されているanyuid Security Context Constraintを用いてVerticaを動作させることができるようになる。

## Spread通信の暗号化

---

Spread通信の暗号化がデフォルトで有効となる。

## スーパーユーザ名の指定

---

スーパーユーザ名としてdbadmin以外の指定が可能となる。

# 機械学習

## Modelのインポートおよびエクスポート

---

IMPORT\_MODELSおよびEXPORT\_MODELSファンクションにおいて、以下の変更を実施。

- 実行できるユーザに、[MLSUPERVISOR](#)ロールを持つユーザを追加。
- 入出力ファイルの保管場所として、Object Storageを含むVerticaのサポートするすべてのFilesystemを利用可能。

**opentext™**

