



Vertica 23.3 新機能紹介

2023年7月14日 | マイクロフォーカスエンタープライズ株式会社

Client Driver

ODBC: COPY, INSERTのエラー情報取得の改善

データロードをCOPY文またはPrepared Statementを用いたINSERT文で行った場合、SQLGetDiagRec()を用いて最初のエラー情報を取得することができるが、これを50のエラー情報まで取得できるように改善される。

ADO.NET: LinuxおよびmacOSのサポート

ADO.NET Driverが.NET Coreをベースとしたものとなり、Windowsの他にLinuxおよびmacOSでも使用できるようになる。

インストール要件はMicrosoft .NET Standard 2.0以降 (Microsoft .NET Framework 4.6.1以降および.NET Core 3.1以降) の事前インストールとなり、推奨は.NET 6.0 (長期サポート最新版)。

クライアント接続

Workload Routing

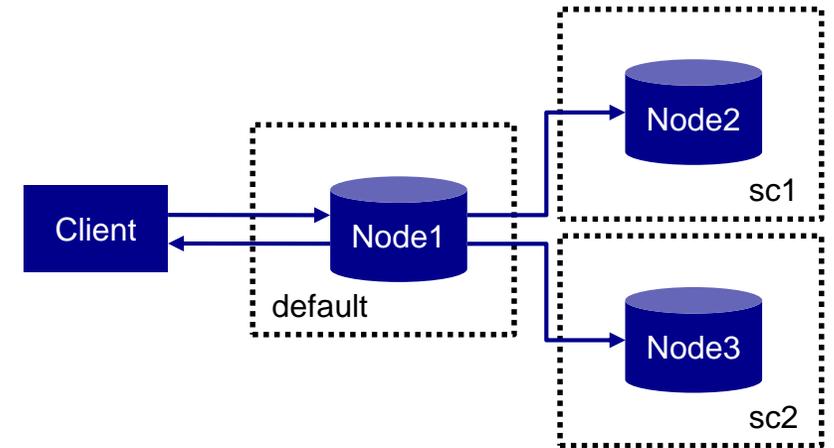
Workloadに応じて特定のSubclusterにクライアント接続をルーティングする機能。

- **Connection Node**: クライアントが接続したノード。Proxyのように振る舞う。
- **Execution Node**: Workload Routingによってクライアント接続が振り分けられ、その接続のサーバセッションを持つノード。
このノードがクエリ実行のInitiator Nodeとなる。

vsql, JDBC driver, ODBC driver, ADO.NET driver, vertica-sql-go, vertica-python, vertica-nodejsのすべてで使用可能。
(一部リリース待ち)

```
=> SELECT subcluster_name, node_name FROM subclusters;
subcluster_name | node_name
-----+-----
default_subcluster | v_eondbrh8_node0001
sc1_cluster      | v_eondbrh8_node0002
sc2_cluster      | v_eondbrh8_node0003

=> CREATE ROUTING RULE ROUTE WORKLOAD analytics
-> TO SUBCLUSTER sc1_cluster;
=> CREATE ROUTING RULE ROUTE WORKLOAD reporting
-> TO SUBCLUSTER sc2_cluster;
```



クライアント接続

Workload Routing(続き)

```
=> SELECT workload, subcluster_name FROM v_catalog.workload_routing_rules;
workload | subcluster_name
-----+-----
analytics | sc1_cluster
reporting | sc2_cluster
=> SELECT distinct node_name FROM session_subscriptions WHERE is_participating;
node_name
-----
v_eondbrh8_node0001
=> SET SESSION WORKLOAD TO analytics;
=> SHOW WORKLOAD;
name | setting
-----+-----
workload | analytics
=> SELECT distinct node_name FROM session_subscriptions WHERE is_participating;
node_name
-----
v_eondbrh8_node0002
```

Eon Mode

In-Database Restore Point

手軽にデータコピーを行わないバックアップ代替機能の提供。

データベースを過去の特定の時点までロールバックさせる。

vbrが提供するバックアップを置き換える機能ではないので、Communal Storageのデータ破損などの対策としては機能しない。

```
=> CREATE TABLE tbl1 (value1 INTEGER);
```

```
=> CREATE ARCHIVE arch1 LIMIT 3;
```

```
=> SAVE RESTORE POINT TO ARCHIVE arch1;
```

```
=> SELECT * FROM archive_restore_points;
```

id	save_time	index	archive	state
eca2339c-dba4-40fa-91ae-fbcaeb781d7f	2023-06-26 13:45:41.965803+09	1	arch1	COMPLETE

```
=> DROP TABLE tbl1;
```

```
=> ¥d
```

```
No relations found.
```

Eon Mode

In-Database Restore Point(続き)

```
$ admintools -t revive_db -x auth_params.conf --communal-storage-location=s3://bucket/directory  
-s hosts -d db_name --restore-point-archive arch1 --restore-point-index 1
```

```
=> ¥d
```

```
      List of tables
```

```
Schema | Name | Kind | Owner | Comment
```

```
-----+-----+-----+-----+-----
```

```
public | tbl1 | table | dbadmin |
```

```
=> REMOVE RESTORE POINT FROM ARCHIVE arch1 INDEX 1;
```

SandboxへのSubclusterの追加

既存のSandboxに対して新たなSubclusterの追加、および不要となったSubclusterの削除をサポート。

データベース管理

Node Management Agent

REST APIを用いたノード管理を行う新しいAgentの実装。ネットワークポート 5554 を使用。

```
$ /opt/vertica/bin/manage_node_agent.sh start node_management_agent

$ curl -X GET https://localhost:5554/v1/vertica/version -k ¥
  --key /opt/vertica/config/https_certs/dbadmin.key ¥
  --cert /opt/vertica/config/https_certs/dbadmin.pem ¥
  --cacert /opt/vertica/config/https_certs/rootca.pem

{"vertica_version": "Vertica Analytic Database v23.3.0-0"}
```

HTTPSサービス

REST APIおよびPrometheus server metricsで使用するHTTPSサービス。

[HTTPServerPortOffset](#)設定パラメータで使用するネットワークポートを変更可能。

Projection

パーティション化されたProjectionのRefreshの改善

以前までは、すべてのデータのRefreshが完了するまでProjectionはクエリで使用されることはなかった。

今後は、パーティション化されたProjectionについては、最も大きい値の範囲のパーティションから順にRefreshを行い、Refreshが完了したパーティションからクエリで使用できるようになった。

```
=> SELECT * FROM vs_configuration_parameters WHERE parameter_name = 'UseIncrementalProjectionRefresh'
   ORDER BY node_name LIMIT 1;
-[ RECORD 1 ]-----+-----
node_name          | v_db_node0001
parameter_name     | UseIncrementalProjectionRefresh
current_value      | 0
default_value      | 0
is_system_parameter | t
change_under_support_guidance | t
change_requires_restart | f
description        | Value of 1 will allow Vertica to consider using PRPs during projection
                    | refresh.
```

データエクスポート

既存ディレクトリへのエクスポート

すべてのEXPORT TO文がすでに存在するディレクトリへのエクスポートを許可していなかった。

新しいifDirExistsパラメータを使用することで、既存ディレクトリの上書きまたは追記ができるようになる。

ifDirExistsパラメータ

- fail(デフォルト値): 指定されたディレクトリが存在する場合はエラー
- overwrite: 指定されたディレクトリが存在する場合は削除してからエクスポート
- append: 指定されたディレクトリが存在する場合、何もせずにエクスポート

データロード

Parquet Loose Schema MatchingによるComplex Typeのサポート

Parquet Parserの`do_soft_schema_match_by_name`オプションを用いて名前一致を用いたSchema Matchingを行う場合、Complex Typeのカラム定義はできるが参照できない制約があった。

このバージョンより、Complex Typeの参照も可能となる。

External Table

Dynamic Partition Pruningのサポート

Join SIPを用いたExternal TableのPartitioningをサポートし、パフォーマンスの改善を実現。

```
SELECT * FROM t_big JOIN t_small ON t_big.c = t_small.a WHERE t_small.b = 5;
```

Apache IcebergのMAPデータ型のサポート

Apache Icebergを用いてExternal Tableを作成する際、map<key,value>型をARRAY[ROW(key,value)]として読み込むことをサポート。

Complex Types

Complex Typeを持つテーブルのカラムに対するDEFAULTおよびSET USING

以前までは、Complex Typeを持つテーブルのスカラー型のカラムへのDEFAULTおよびSET USINGの指定はできなかった。

今後は、スカラー型のカラムへのDEFAULTおよびSET USINGの指定ができるようになる。

ただし、Complex Typeのカラムへの指定はできない。

[12.0.4]

```
=> CREATE TABLE tbl (id INTEGER, something INTEGER DEFAULT 47, x ROW(z INTEGER, asdf ARRAY[INTEGER]));  
ERROR 9202: Tables with complex data types cannot have default columns
```

[23.3]

```
=> CREATE TABLE tbl (id INTEGER, something INTEGER DEFAULT 47, x ROW(z INTEGER, asdf ARRAY[INTEGER]));  
CREATE TABLE
```

Complex Types

EXPLODEおよびUNNESTの出力を用いたLEFT JOIN

CROSS JOINに加えて、サブクエリを用いずにEXPLODEおよびUNNESTの出力結果をLEFT JOINで用いることができるようになる。

```
=> SELECT student, MIN(score), AVG(score) FROM tests  
-> LEFT JOIN LATERAL EXPLODE(scores) AS t(pos, score) GROUP BY student;
```

SQLステートメント・ファンクション

SQLステートメント・ストアドプロシージャが参照するオブジェクトの表示

SQLステートメントまたはストアドプロシージャが参照するデータベースオブジェクトの一覧を表示する機能。

```
=> SELECT t1.id, t2.pid, t1.value FROM tbl1 t1 JOIN tbl2 t2 USING (id) WHERE t2.pid = 1;
```

```
id | pid | value
```

```
-----+-----+-----
```

```
1 | 1 | 1
```

```
4 | 1 | 8
```

```
5 | 1 | 0
```

```
=> ALTER SESSION SET ShowTablesInQueries = 1;
```

```
=> SELECT t1.id, t2.pid, t1.value FROM tbl1 t1 JOIN tbl2 t2 USING (id) WHERE t2.pid = 1;
```

```
-----
```

```
TBL1
```

```
TBL2
```

SQLステートメント・ファンクション

セッションユーザの変更

現在ログイン中のセッションユーザを変更する機能。テスト用途など一時的にユーザを切り替えたいユースケースを想定。
dbadminユーザのみが使用できる。

```
=> SELECT current_user(), session_user();
current_user | session_user
-----+-----
dbadmin      | dbadmin

=> SET SESSION AUTHORIZATION user01;

=> SELECT current_user(), session_user();
current_user | session_user
-----+-----
user01       | user01

=> SET SESSION AUTHORIZATION DEFAULT;
```

ストアードプロシージャ

トランザクション制御

以前までは、Stored Procedureは常にAuto Commitを行い、実行を失敗したSQLステートメントがあってもRollbackしなかった。

今後は、Stored Procedureが正常に完了した場合にCommitし、それ以外はRollbackするようになる。Stored Procedureの途中でもCommitの実行も可能。

ネストしたストアードプロシージャのサポート

ストアードプロシージャからストアードプロシージャまたはメタファンクションの実行ができるようになる。最大100階層までネスト可能。

[EnableNestedStoredProcedures](#) 設定パラメータを 0 に設定することで今までの動作と同様にネストを禁止することも可能。

セッションに対する設定の維持

ALTER SESSIONなどを用いてストアードプロシージャで変更されたセッション設定は、ストアードプロシージャ実行後のセッションでも維持されるようになる。

セキュリティ・ユーザ認証

Control channel (Spread) のTLS設定

Control channel (Spread) のTLSを有効化するためには、次の手順が必要だった。

- 1 EncryptSpreadComm設定パラメータの変更
- 2 データベースの再起動
- 3 SECURITY_CONFIG_CHECKファンクションによる設定確認

この手順に加え、データベース作成時に有効化する手順が追加。

```
$ admintools -t create_db -d verticadb -s 192.0.2.100,192.0.2.101,192.0.2.102  
  --config-param EncryptSpreadComm='vertica'
```

ODBC OAuth認証を用いたSingle Sign-On

ODBC driverを用いたデスクトップアプリケーションを使用する場合、ODBC driverが自動的にブラウザを起動してKeycloakの認証エンドポイントを表示させ、SSOを用いたサインインをユーザに行わせることができるようになる。
これは、Identity Providerを用いたユーザ認証において、OAuthAccessTokenを手動で取得する方法の一つとなる。

セキュリティ・ユーザ認証

Catalogの暗号化

Catalogおよびvertica.confはすべてテキストデータとして保管されており、一部の設定情報には機密性の高いものもあり、特にVerticaノードでOSログインを許可している環境においては、Catalogの暗号化が適切となるケースも存在する。

```
$ admintools -t crypt_db --encrypt -d db -p pwd -c cat_encryption.json
```

[暗号化前]

```
$ file v_db_node0001_catalog/Catalog/config.cat
config.cat: ASCII text, with very long lines
```

```
$ file vertica.conf
vertica.conf: ASCII text
```

[暗号化後]

```
$ file v_db_node0001_catalog/Catalog/config.cat
config.cat: data
```

```
$ file vertica.conf
vertica.conf: data
```

データベース起動時に復号化のためのパスワードを入力する必要がある。

```
$ admintools -t start_db -d trial --encrypt-pass-prompt
password used to decrypt the catalog: <pwd>
```

Kubernetes・Containers

データベース削除時のPV, PVCの取り扱い

以前までは、データベースを削除する際はPodのみを削除し、使用していたPersistentVolume (PV) と PersistentVolumeClaim (PVC) は残されていたため、手動で削除する必要があった。

KubernetesにてPodの削除と同時に関連するPV, PVCの削除が可能となったため、VerticaDB operatorでもこれを実装。

S3 Server Side Encryptionのサポート

Server Side Encryptionが有効となっているS3をCommunal Storageとして使用することができるようになる。

- communal.s3ServerSideEncryption: Server Side Encryptionのタイプ (SSE-S3, SSE-KMS, SSE-C)
- communal.s3SseCustomerKeySecret: SSE-Cを利用する際に使用するSecret
- communal.additionalConfig > S3SseKmsKeyId: SSE-KMSを利用する際に使用するKey ID

Event Trigger

データベースを作成してからアプリケーションで使用を開始する前に、事前準備を行いたいケースがある。

- クライアント認証の設定やDCテーブルのポリシー変更など

CRDのEvent Triggerを用いてDBInitializedのステータスになった際に任意のコマンドを実行できるように設定できるようになる。

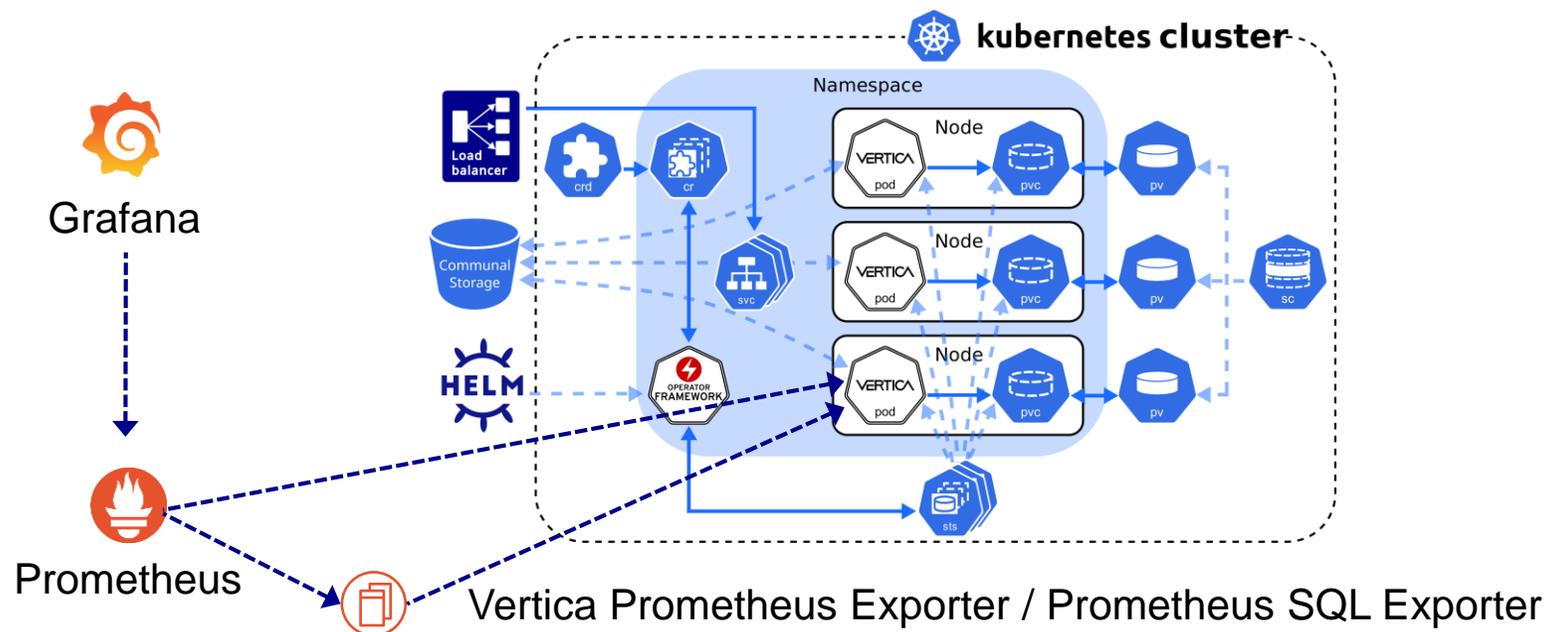
Kubernetes・Containers

Prometheus server metrics

Prometheus metricsをExporterなしで提供できるようになる。

In-Databaseのため高速に提供できる一方、Metricsが事前定義となるので必要なMetricsがないケースもある。

その場合は引き続きExporterでの対応が必要。



機械学習

PMMLサポートの拡張

サポートするPMML Tagに[FieldRef](#)と[LocalTransformations](#)を追加。

サポートするSegmentation Tagの属性として[missingThreshold](#)と[missingPredictionTreatment](#)を追加。

Segmentation TagのmultipleModelMethod属性の値として[average](#)をサポート。

インポートしたTensorFlowモデルによるComplex Typeのサポート

既存のPREDICT_TENSORFLOWファンクションでは入出力に必要な数のカラムを指定する必要があった。

TensorFlowモデルにおいてROW型かつフィールドに一次元の配列を持つデータ型での入出力が有効であるケースが多いことから、新たに入出力にそのデータ型を指定する[PREDICT_TENSORFLOW_SCALAR](#)ファンクションを実装。

```
=> SELECT pid, PREDICT_TENSORFLOW_SCALAR(inputs USING PARAMETERS model_name='tf_model') AS outputs
-> FROM input_table;
```

pid	outputs
0	{"output_op1:0": [1.0, 2.0], "output_op1:1": [3.0, 4.0], "output_op2:0": [5.0, 6.0]}

機械学習

Modelオブジェクトによるスキーマ権限継承のサポート

テーブルなどで利用できるスキーマ権限継承がModelオブジェクトでもサポート。

```
=> ALTER MODEL <name> INCLUDE SCHEMA PRIVILEGES;  
=> ALTER MODEL <name> EXCLUDE SCHEMA PRIVILEGES;
```

Management Console

Keycloakと連携したユーザ認証

Management Consoleがユーザ認証にKeycloakとの連携を用いることにより、次の3つの認証方法をサポート。

- Local
- Federated: LDAP / LDAPSとの連携
- Identity Provider: 次の認証プロトコルを使用するIdentity Providerとの連携
 - SAML v2.0
 - OpenID Connect v1.0
 - Keycloak OpenID Connect

Management Console

コーポレートリブランディング

OpenTextへのリブランディングを実施。

The image displays the OpenText Vertica Management Console interface, illustrating the corporate rebranding. It is divided into two main sections: the login page and the main dashboard.

Login Page: Features the OpenText logo, the text "VERTICA Management Console", and a login form with fields for "Username" and "Password". A "Remember me" checkbox and a "Log In" button are also present. A "Forgot Password?" link is located below the password field. The footer includes "Copyright 2007 - 2023 Open Text".

Main Dashboard: The dashboard is titled "opentext | Vertica Management Console" and shows the user "uadmin" with a "Log out" button. The navigation breadcrumb is "Databases and Clusters > trial > Overview". Key metrics include "Total number of database nodes: 3" and "Auto Refresh" status. The dashboard contains several panels:

- CPU/Memory/Disk I/O (Showing 58 minutes):** A line chart showing usage over time.
- Database General Pool Usage:** A line chart showing Max Queue Time, Avg Free Memory, and Resource Rejections.
- Database Nodes Health:** A summary of node status: 0 Down, 0 Critical, 3 Up, 0 Recovering. Total: 3 nodes, Version: 23.3.0-0, Enterprise Mode I Storage.
- Running And Queued Queries (Database):** Summary of query status: 0 Running, 0 Queued.
- Queries (Showing 1 day):** Summary of query execution: 0 Running, 0 Queued, 23 Completed, 1 Failed. A circular gauge shows "Currently there are no running or queued queries".
- Projections:** Summary of projection status: 10 Total, 0 Unsegmented, 0 Unsafe/Not Up To Date. For Largest Schema: grasp.
- Disk Space Usage:** Summary of disk usage: 0 Low. Total: 3 Nodes, Based On Config Param For Alerts On Disk Usage: >60%.

The bottom navigation bar includes "MC Settings", "Message Center", and "MC Diagnostics". A footer section provides links for "Documentation and Community" and "Video Tutorials And Training". The footer text reads: "Vertica Management Console, v23.3.0 - Build: 23.3.0 - 0 Copyright 2007-2023 Open Text".

The image features the OpenText logo, which consists of the word "opentext" in a bold, lowercase, sans-serif font, followed by a trademark symbol (TM). The text is white and centered horizontally. The background is black, with several bright blue, glowing, curved lines that sweep across the frame from the bottom left towards the top right, creating a sense of motion and energy.

opentextTM