

VERTICA

11.1 新機能 紹介

マイクロフォーカスエンタープライズ株式会社
2022年2月24日

データ型

- Complex Typeのキャスト

- 以前まではプリミティブ型の複次元のARRAY型のキャストまでをサポート。

```
=> SELECT ARRAY[ARRAY[1, 2, 3], ARRAY[4, 5, 6]]::ARRAY[ARRAY[VARCHAR]];
      ARRAY
-----
[["1", "2", "3"], ["4", "5", "6"]]
```

- ROW型のキャストをサポート。同時にフィールドの名称の変更も可能。

```
=> SELECT ROW(1, 2, 3)::ROW(VARCHAR, FLOAT, NUMERIC);
      row
-----
{"f0":"1", "f1":2.0, "f2":"3.0000000000000000"}

=> SELECT ROW(1, 2, 3)::ROW(a VARCHAR, FLOAT, c NUMERIC);
      row
-----
{"a":"1", "f1":2.0, "c":"3.0000000000000000"}
```

データ型

- Complex Typeのキャスト（続き）
 - Complex Typeを持つARRAY型のキャストをサポート。

```
=> CREATE TABLE comp_native (  
    id INT,  
    data ARRAY[ROW(a INT, b INT)]  
);
```

```
=> SELECT data FROM comp_native;  
data
```

```
[{"a":1, "b":2}]  
[{"a":11, "b":12}]
```

```
=> SELECT data::ARRAY[ROW(field1 VARCHAR, field2 FLOAT)] FROM comp_native;  
data
```

```
[{"field1":"1", "field2":2.0}]  
[{"field1":"11", "field2":12.0}]
```

データ型

- Complex Typeのキャスト（続き）
 - ARRAY型の最大要素数の変更。
 - ・ 新しい最大要素数が実要素数より少ない場合：
 - 要素がプリミティブ型の場合、溢れた要素は切り捨てられる。
 - 要素がComplex Typeの場合、エラーとなる。

```
=> SELECT ARRAY[1, 2, 3]::ARRAY[VARCHAR, 2];  
      array
```

```
-----  
["1", "2"]
```

```
=> SELECT ARRAY[ARRAY[1, 2, 3], ARRAY[4, 5, 6]]::ARRAY[ARRAY[VARCHAR], 1];  
ERROR 9783: Multi-value expressions are not supported in this context  
DETAIL: Multi-value expression used in cast to partially-bounded complex type
```

データロード

- JSON, Avro ParserによるComplex TypeのStrong Typeのサポート。

[Flexible Complex Type]

```
=> CREATE EXTERNAL TABLE rest (  
    name VARCHAR, cuisine VARCHAR, location_city LONG VARBINARY, menu LONG VARBINARY  
    ) AS COPY FROM source_data PARSER FJSONPARSER();  
  
=> SELECT name, MAPTOSTRING(location_city) as location_city, MAPTOSTRING(menu) AS menu FROM rest;
```

[Strong Type Complex Type]

```
=> CREATE EXTERNAL TABLE rest (  
    name VARCHAR, cuisine VARCHAR, location_city ARRAY[VARCHAR(80), 50],  
    menu ARRAY[ROW(item VARCHAR(80), price FLOAT), 100]  
    ) AS COPY FROM source_data PARSER FJSONPARSER();  
  
=> SELECT name, location_city, menu FROM rest;
```

- ロードしたデータを直接参照できる。
- 定義に一致しないレコードはロードされない。

データロード

- Parquet Parser
 - [DataPageV2](#)で書き出されたParquetファイルのサポート。
 - [DELTA_BINARY_PACKED](#)エンコードを使用したParquetファイルのサポート。

SQLステートメント・ファンクション

- INFER_EXTERNAL_TABLE_DDLによるAvroフォーマットのサポート。

```
=> SELECT INFER_EXTERNAL_TABLE_DDL (' /data/avro_complex_a1.avro'  
    USING PARAMETERS format = 'avro', table_name = 'orders');
```

ストアドプロシージャ

- 実行権限・設定パラメータ

- SECURITY DEFINER（作成者で実行）が指定されたストアドプロシージャの実行は以下の権限・設定パラメータを反映。

- ・ デフォルトのロール
 - ・ USERレベルの設定パラメータ
 - ・ SESSIONレベルの設定パラメータ
 - ・ ユーザ設定情報
 - RESOURCE POOL
 - MEMORYCAP
 - TEMPSPACECAP
 - RUNTIMECAP

セキュリティ

- PasswordLockTimeUnit 設定パラメータ

- ログイン失敗回数によるアカウントロック

- FAILED_LOGIN_ATTEMPTS プロファイルパラメータ：ログイン失敗回数

- PASSWORD_LOCK_TIMEプロファイルパラメータ：アカウントロック日数



- PasswordLockTimeUnit 設定パラメータによりPASSWORD_LOCK_TIMEの単位を指定

- 'd' : 日 (デフォルト)

- 'h' : 時間

- 'm' : 分

- 's' : 秒

セキュリティ

- OAuth 2.0によるユーザ認証のサポート

```
=> CREATE AUTHENTICATION oauthhost METHOD 'oauth' HOST '0.0.0.0/0' ;  
=> GRANT AUTHENTICATION oauthhost TO user1;  
=> ALTER AUTHENTICATION oauthhost SET discovery_url = 'http://discovery.url';
```

- 対応クライアント : JDBC, ODBC
- 対応中クライアント : vsql, ADO.NET

Containers ・ Kubernetes

- Red Hat OpenShiftのサポート
 - OpenShift Container Platform Webコンソールからのインストール
 - Security Context Constraint (SCC) の対応
 - ・ Server Default SCCのうちprivilegedをサポート
 - ・ anyuid-extraを用いたCustom SCC
 - 複数のOpenShift Namespaceへのインストールに対応
- パブリッククラウドのサポート状況
 - サポート開始
 - ・ Amazon Elastic Kubernetes Service (EKS)
 - ・ Google Kubernetes Engine (GKE)
 - サポート予定
 - ・ Microsoft Azure Kubernetes Service (AKS)

Containers ・ Kubernetes

- Online Upgradeのサポート

- Eon ModeのRead-onlyモードを用いてユーザクエリに対するサービスを継続した状態でVerticaのUpgradeを実施

- 内部動作

1. クライアント要求を受け取る一時的なSubclusterを作成 (オプション)
2. Primary Subclusterに対するクライアント接続を他のSubclusterにルーティング
3. Primary Subclusterを新しいイメージに置き換えて起動
4. Primary Subclusterに対するクライアント接続を元に戻す
5. Secondary Subclusterに対するクライアント接続を他のSubclusterにルーティング
6. Secondary Subclusterを新しいイメージに置き換えて起動
7. Secondary Subclusterに対するクライアント接続を元に戻す
8. Secondary Subclusterが複数存在する場合は5~7を繰り返し
9. 1で作成した一時的なSubclusterを削除

Containers ・ Kubernetes

- Operatorのロギングの改善
 - 出力先として標準出力以外にファイルへの出力を追加
 - ログレベルの追加
- 環境変数の制御
 - コンテナに対する環境変数を制御するために、Custom Resource Definitionに **annotations**パラメータを追加

機械学習

- TensorFlow Integration

- サポートデータ型の拡充。

- ・ 入力 : `TF_INT8`, `TF_INT16`, `TF_INT32`, `TF_INT64`, `TF_FLOAT`, `TF_DOUBLE`

- ・ 出力 : `TF_DOUBLE`, `TF_INT8`, `TF_INT16`, `TF_INT32`, `TF_FLOAT`, `TF_INT64`

User-Defined Extensions

- クライアントへのメッセージ通知 (C++のみ)
 - 以前までのメッセージ通知
 - vt_report_errorマクロ: エラーメッセージと共にUDxの実行を停止
 - ServerInterface.vlog(): ロギングのみ (UDxFencedProcesses.logまたはvertica.log)
 - 次のServerInterfaceのメソッドを用いてVerticaと同様のメッセージ通知が可能
 - `reportError` (UDxの実行も停止)
 - `reportWarning`
 - `reportNotice`
 - `reportInfo`

システムテーブル

- NODESテーブル

- BUILD_INFOカラムの追加。

```
=> SELECT node_name, node_state, build_info FROM nodes;  
node_name | node_state | build_info  
-----+-----+-----  
v_db_node0001 | UP | v11.1.0-0-21df975199092ba888ba3ef1c536e1898d192274
```

- 新しいノードステータス (NODESおよびNODE_STATESテーブルのNODE_STATEカラム)

- **UNKNOWN** : ノードがクラスタの一部として認識されているにも関わらず、状態を確認するための通信ができない状態



VERTICA

www.vertica.com

www.microfocus-enterprise.co.jp