異なるデータベース間の SQL 比較と Oracle Database 12c の新機能

Noriyoshi Shinoda

March 8, 2017

自己紹介
篠田典良（しのだのりよし）

- 所属
  - 日本ヒューレット・パッカード株式会社 テクノロジーコンサルティング事業統括
- 現在の業務
  - Oracle DatabaseをはじめPostgreSQL, Microsoft SQL Server, Vertica, Sybase ASE等 RDBMS全般に関するシステムの設計、チューニング、コンサルティング
  - Oracle ACE
  - Oracle Database関連書籍の執筆
  - オープンソース製品に関する調査、検証
- 関連する URL
  - Oracle ACEってどんな人？
    - http://www.oracle.com/technetwork/jp/database/articles/vivadeveloper/index-1838335-ja.html
  - 「PostgreSQL 虎の巻」シリーズ
    - http://h30507.www3.hp.com/t5/user/viewprofilepage/user-id/838802

Agenda

– 自動採番列 (12.1)
– 文字列型の最大長 (12.1)
– FETCH n ROWS (12.1)
– オブジェクト名最大長 (12.2)
– 照合 (12.2)
– JSONデータ (12.1 / 12.2)

Oracle Database 12*c*新機能

Oracle Database 12c新機能
自動採番列

RDBMS	自動採番列の構文
MySQL	AUTO_INCREMENT列属性
PostgreSQL	serial, bigserial, smallserial型
DB2	GENERATE ALWAYS 列属性
SQL Server	IDENTITY列属性
Vertica	AUTO_INCREMENT列属性 IDENTITY列属性

Oracle Database 12c新機能
自動採番列

- Oracle Database 12c (12.1) から利用可能
- GENERATED ALWAYS AS IDENTITY 列属性
  - 常に自動採番値を生成、更新不可
- GENERATED BY DEFAULT AS IDENTITY 列属性
  - ユーザーが更新可能

```
SQL> CREATE TABLE idtbl1 (id NUMBER
       GENERATED BY DEFAULT AS IDENTITY (START WITH 10),
         val VARCHAR2(10)) ;
```

- 実体は自動生成される SEQUENCE オブジェクトと DEFAULT の組み合わせ
- 利用するためには CREATE TABLE システム権限に加えて CREATE SEQUENCE システム権限が必要

Oracle Database 12c新機能
文字列型最大長

RDBMS	CHAR	VARCHAR
MySQL	255	65,535
PostgreSQL	10,485,760	10,485,760
DB2	255	32,704
SQL Server	8,000	2 GB
Vertica	65,000	65,000
Oracle Database 11*g* (11.2)	2,000	4,000

Oracle Database 12c新機能
文字列型最大長

- Oracle Database 12c (12.1) から VARCHAR2 型 / NVARCHAR2 型 / RAW 型の最大長を 32,767 バイトに拡張可能
  - CHAR / NCHAR 型の最大長はそのまま
  - PL/SQL の最大長に合致
- 初期化パラメーター MAX_STRING_SIZE = EXTENDED が必要
  - @?/rdbms/admin/utl32k.sql の実行
  - @?/rdbms/admin/utlrp.sql の実行
  - 再起動
  - 一旦変更すると元に戻せない
- 初期化パラメーター COMPATIBLE >= 12.0.0.0 も必要
- 内部的には初期化パラメーター _scalar_type_lob_storage_threshold (Default 2,000) を超えるデータはインライン BLOB 型に変換して保存

Oracle Database 12c新機能
FETCH n ROWS

RDBMS	最初の 10 レコード取得
MySQL	LIMIT 10
PostgreSQL	FETCH FIRST 10 ROWS ONLY
DB2	FETCH FIRST 10 ROWS ONLY
SQL Server	TOP 10 FETCH FIRST 10 ROWS ONLY
Vertica	LIMIT 10

– Oracle Database 11*g* の記述

```
SELECT first_name, last_name, salary FROM (
    SELECT first_name, last_name, salary,
        ROW_NUMBER() OVER (ORDER BY salary DESC)  ranking
    FROM employees)
WHERE ranking <= 10
```

Oracle Database 12c新機能
FETCH n ROWS

– Oracle Database 12c (12.1) から OFFSET n ROWS FETCH FIRST m ROWS ONLY 構文が利用可能

```
SQL> SELECT first_name, last_name, salary FROM employees
     ORDER BY salary DESC
     FETCH FIRST 10 ROWS ONLY ;
```

– OFFSET n ROWS 句を指定して途中を抜き出すこともできる

– 実行計画を確認すると内部的には ROW_NUMBER 関数を使っている

```
1 - filter("from$_subquery$_002"."rowlimit_$$_rownumber"<=10)
2 - filter(ROW_NUMBER() OVER (
          ORDER BY "EMPLOYEES".“SALARY" )<=10)
```

Oracle Database 12c新機能
オブジェクト名最大長

– 主要 RDBMS のオブジェクト名の最大長

RDBMS	最大長
MySQL	64 char
PostgreSQL	63 bytes
DB2	128 bytes
SQL Server	128 char
Vertica	128 bytes
Oracle Database 12c (12.1)	30 bytes

Oracle Database 12c新機能
オブジェクト名最大長

- Oracle Database 12c (12.2) から主なオブジェクトの最大長は 128 バイト
- 初期化パラメーター COMPATIBLE >= 12.2 が必要
- 以下は例外

オブジェクト	最大長
データベース名	8 bytes
表領域名	30 bytes
サービス名	64 bytes
ASMディスクグループ	30 bytes
PDB名	30 bytes

- V$PDBS ビューの NAME 列, DBA_PDBS ビューの PDB_NAME 列は 128 バイトだが？

Oracle Database 12c新機能
照合（COLLATION）

– 日本語環境で標準インストールした場合
– 文字列型による大文字／小文字の区別

RDBMS	大文字／小文字区別	列単位に変更
MySQL	しない	○
PostgreSQL	する	×
DB2	する	×
SQL Server	しない	○
Vertica	する	×
Oracle Database 12c (12.1)	する	×

– Oracle Database のデフォルトは初期化パラメーター NLS_COMP, NLS_SORT に依存

Oracle Database 12c新機能
照合（COLLATION）

- Oracle Database 12c (12.2) では列／テーブル単位に COLLATE 指定可能
  - 確認方法は ALL_TAB_COLUMNS ビューの COLLATION 列、ALL_TABLES ビューの DEFAULT_COLLATION 列

```
SQL> CREATE TABLE table_name1 (val VARCHAR2(10))
     DEFAULT COLLATION BINARY_AI ;

SQL> CREATE TABLE table_name2 (id NUMERIC,
                    val VARCHAR2(10) COLATE BINARY_CI) ;
```

- 使用条件
  - 初期化パラメーター COMPATIBLE >= 12.2
  - 初期化パラメーター MAX_STRING_SIZE=EXTENDED

Oracle Database 12c新機能
照合（COLLATION）

```
SQL> CREATE TABLE collate1 (col1 VARCHAR2(10),
            col2 VARCHAR2(10)  COLLATE BINARY_CI) ;
SQL> INSERT INTO collate1  VALUES ('ABC', 'ABC') ;
SQL> INSERT INTO collate1  VALUES ('abc' 'abc') ;

SQL> SELECT * FROM collate1 WHERE col1='abc' ;
COL1       COL2
---------- ----------
abc        abc

SQL> SELECT * FROM collate1 WHERE col2='abc' ;
COL1       COL2
---------- ----------
ABC        ABC
abc        abc
```

Oracle Database 12c新機能
照合（COLLATION）

– COLLATION が異なる列は直接結合できない

```
SQL> SELECT * FROM collate1 c1 INNER JOIN collate2 c2 ON
                c1.col1 = c2.col2 ;
SELECT * FROM collate1 c1 INNER JOIN collate2 c2 ON
                c1.col1 = c2.col2
                          *
行1でエラーが発生しました。:
ORA-43915: 照合を決定できません: 引数に競合する照合があります
```

– COLLATION を合わせてから結合する

```
SQL> SELECT * FROM collate1 c1 INNER JOIN collate2 c2 ON
          c1.col1 COLLATE BINARY_CI = c2.col2 ;
```

Oracle Database 12c新機能
照合（COLLATION）

– 実行計画を確認すると、自動的に NLSSORT 関数を実行している

```
SQL> SELECT * FROM collate1 WHERE col2='ABC' ;

Predicate Information (identified by operation id):
---------------------------------------------------
   1 - filter(NLSSORT("COL2",'nls_sort=''BINARY_CI''')=HEXTORAW('61626300'))
```

Oracle Database 12c新機能
JSONデータ対応

RDBMS	データ型	関数	演算子
MySQL	JSON	○	○
PostgreSQL	JSON, JSONB	○	○
DB2	-	○	-
SQL Server	-	○	-
Vertica	-	○	○
Oracle Database 12c (12.1)	-	○	-

– Oracle Database 12c (12.2) では、JSON データを生成する関数、パッケージが追加された

Oracle Database 12c新機能
JSONデータ対応

- JSON 専用のデータ型は未提供（VARCHAR2, CLOB を利用）
- 条件式が利用可能
  - IS JSON / IS NOT JSON
  - JSON_EXISTS
  - JSON_TEXTCONTAINS
- 関数
  - JSON_ARRAY / JSON_ARRAYAGG (12.2)
  - JSON_DATAGUIDE (12.2)
  - JSON_OBJECT / JSON_OBJECTAGG (12.2)
  - JSON_QUERY
  - JSON_TABLE
  - JSON_VALUE
- DBMS_JSON パッケージ (12.2)
- CREATE SEARCH INDEX FOR JSON

Oracle Database 12c新機能
JSONデータ対応

– JSON データ生成関数が追加（12.2）

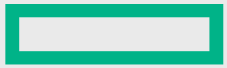
```
SQL> SELECT JSON_ARRAY (
  2       JSON_OBJECT('Percentage' VALUE .30),
  3       JSON_ARRAY('A', 'B', 'C')
  4  ) FROM DUAL ;

JSON_ARRAY(JSON_OBJECT('PERCENTAGE'VALUE.30),JSON_ARRAY('A
','B','C'))
--------------------------------------------------------------------------------
[{"Percentage":0.3},["A","B","C"]]
SQL>
```

新機能の情報

新機能の情報
マニュアル

– 新機能ガイド

– 各マニュアルの先頭部分



Hewlett Packard
Enterprise

Thank you

noriyoshi.shinoda@hpe.com