Hidden incompatibility in the commonly used SQL

Noriyoshi Shinoda

December 3, 2016

Author

Noriyoshi Shinoda

- Company
  - Hewlett-Packard Enterprise Japan
- Current Work
  - System design, tuning, consulting on RDBMS such as Oracle Database, PostgreSQL, Microsoft SQL Server, Vertica, Sybase ASE, etc.
  - Investigation and verification on open source products
  - Oracle ACE
  - Issued books for Oracle Database
  - Training "Introduction to PostgreSQL for Oracle Database Engineers" instructor
- URL (Japanese)
  - PostgreSQL Internals, etc.
    - http://h30507.www3.hp.com/t5/user/viewprofilepage/user-id/838802
  - What kind of person is Oracle ACE?
    - http://www.oracle.com/technetwork/jp/database/articles/vivadeveloper/index-1838335-ja.html

Agenda

Hidden incompatibility in the commonly used SQL

- Compare the SQL of some databases.
  - Basic functions and operators that can be used in many of the RDBMS
  - Sort ordering
  - Behavior at transaction failure
- Compared RDBMS
  - Oracle Database 12c Release 1
  - Microsoft SQL Server 2016
  - PostgreSQL 9.6
  - Oracle MySQL 5.7
  - IBM DB2 Express-C 11.1
  - HPE Vertica Analytic Database 8.0
- Basically default values are used for parameters etc.

1. Functions

1. Functions

MOD (% operator in SQL Server)

- Results of MOD(3, 0) function
  - Error? or No error?

RDBMS	Results
Oracle Database	3
MySQL	NULL
PostgreSQL	Div/0 Error
DB2	Div/0 Error
SQL Server	Div/0 Error
Vertica	Div/0 Error

1. Functions

GREATEST (No function in SQL Server)

– Results of GREATEST(1, 2, NULL) function

  – Handling of NULL values is different.

  – Same behavior with LEAST function

RDBMS	Results
Oracle Database	NULL
MySQL	NULL
PostgreSQL	2
DB2	NULL
SQL Server	-
Vertica	NULL

1. Functions

LENGTH (LEN function in SQL Server)

- Verify the execution result of LENGTH function (character code UTF-8)
  - Return byte length or return number of characters
  - OCTET_LENGTH and CHARACTER_LENGTH functions can be used
  - LENGTH('') is NULL because Oracle Database considers a zero-length string as NULL

RDBMS	LENGTH('あ')	LENGTH('𠮷')
Oracle Database	1	1
MySQL	3	4
PostgreSQL	1	1
DB2	3	4
SQL Server	1	2
Vertica	1	1

1. Functions

LENGTH function for CHAR type

- Execute LENGTH function to store the 'ABC' to CHAR(10) type column
  - Handling of the following space

RDBMS	Results
Oracle Database	10
MySQL	3
PostgreSQL	3
DB2	10
SQL Server	3
Vertica	3

1. Functions

OCTET_LENGTH function for CHAR type

- 'ABC' is stored to the CHAR(10) type column
  - Execute the OCTET_LENGTH and CHARACTER_LENGTH functions
  - Oracle Database uses the LENGTHB / LENGTH function
  - SQL Server uses the DATALENGTH / LEN function

RDBMS	OCTET_LENGTH	CHARACTER_LENGTH
Oracle Database	10	10
MySQL	3	3
PostgreSQL	10	3
DB2	10	10
SQL Server	10	3
Vertica	3	3

1. Functions

CURRENT_TIMESTAMP

- Execute CURRENT_TIMESTAMP function
  - What is the current time in RDBMS?

RDBMS	Results
Oracle Database	OS time
MySQL	OS time
PostgreSQL	Transaction start time
DB2	OS time
SQL Server	OS time
Vertica	Transaction start time

2. Data types and Operators

2. Data types and Operators

String concatenate operator (SQL Server, MySQL is CONCAT function)

– Execute SELECT NULL || 'ABC' statement

  – Handling of NULL value

RDBMS	Results	Notes
Oracle Database	ABC	CONCAT also same behavior
MySQL	NULL	|| operator is a logical sum
PostgreSQL	NULL	CONCAT return ABC
DB2	NULL	CONCAT also same behavior
SQL Server	ABC	NULL when + operator
Vertica	NULL	CONCAT also same behavior

2. Data types and Operators

Zero-length string and NULL

– '' IS NULL is True?

  – Handling of NULL value

RDBMS	Results
Oracle Database	True
MySQL	False
PostgreSQL	False
DB2	False
SQL Server	False
Vertica	False

2. Data types and Operators

Space at INSERT statement

– CREATE TABLE table1 (c1 CHAR(3), c2 VARCHAR(3))

– INSERT INTO table1 VALUES ('ABC<sp>', 'AB<sp>')

RDBMS	'ABC<sp>'	'AB<sp>'
Oracle Database	Error (c1 column error)	
MySQL	'ABC'	'AB<sp>'
PostgreSQL	'ABC'	'AB<sp>'
DB2	'ABC'	'AB<sp>'
SQL Server	'ABC'	'AB<sp>'
Vertica	'ABC'	'AB'

2. Data types and Operators

String type length limt

– Execute CREATE TABLE table_name (column_name CHAR(3))

  – 3 bytes?, 3 characters?

RDBMS	Results
Oracle Database	3 bytes
MySQL	3 characters
PostgreSQL	3 characters
DB2	3 characters
SQL Server	3 characters
Vertica	3 bytes

– Oracle Database can change the default behavior with the parameter nls_length_semantics

2. Data types and Operators

Implicit type conversion

– Execute SELECT ‘123’ + '456' and SELECT '123' + 'XYZ' statement

RDBMS	'123' + '456'	'123' + 'XYZ'
Oracle Database	579	Error
MySQL	579	123
PostgreSQL	Error	Error
DB2	579	Error
SQL Server	'123456'	'123XYZ'
Vertica	Error	Error

3. Syntax and Transaction

3. Syntax and Transaction

Reserved Words

- Names that can not be used as an object name
  - In total 601 reserved words
  - ANY, CASE, DESC, IS, JOIN, USER, WHEN etc, are reserved words of 5 RDBMS

RDBMS	# of words	Original reserved word
Oracle Database	110	ACCESS, CLUSTER, RAW
MySQL	262	CHANGE, CHANNEL, DEC
PostgreSQL	95	ASYMMETRIC, ISNULL
DB2	290	CLONE, CLUSTER, DATA
SQL Server	185	BACKUP, BULK, OVER
Vertica	95	ENCODED, FLEXIBLE, PLAN

3. Syntax and Transaction

ORDER BY

– Execute SELECT column_name FROM table_name ORDER BY 1 ASC

- – Handling of NULL value
- – NULLS FIRST | NULLS LAST syntax to determine the output of NULL values
- – The sort order of string type depends on the locale function and character code

RDBMS	NULL	NULLS FIRST
Oracle Database	Last	Yes
MySQL	First	No
PostgreSQL	Last	Yes
DB2	Last	Yes
SQL Server	First	No
Vertica	Depend on type	Yes

3. Syntax and Transaction

Replacement of primary key value by UPDATE statement

– CREATE TABLE table1 (c1 NUMERIC PRIMARY KEY)

– INSERT INTO table1 VALUES (1)

– INSERT INTO table1 VALUES (2)

– UPDATE table1 SET c1 = CASE c1 WHEN 1 THEN 2 WHEN 2 THEN 1 END

RDBMS	Results	Notes
Oracle Database	No error	
MySQL	PK violation	
PostgreSQL	PK violation	OK with DEFERRABLE
DB2	Syntax error	OK with CASE statement
SQL Server	No error	
Vertica	No error	

3. Syntax and Transaction

Error handling during transaction

- INSERT for table with primary key (column c1)
  - INSERT INTO table1(c1, c2) VALUES (100, 'data1')
  - INSERT INTO table1(c1, c2) VALUES (100, 'data2')
  - COMMIT
  - SELECT COUNT(*) FROM table1

RDBMS	COUNT(*)	Note
Oracle Database	1	
MySQL	1	
PostgreSQL	0	Rollback all statement
DB2	1	
SQL Server	1	
Vertica	2	By default, disabled constraints

7. Conclusion

Conclusion

- Do not get stuck in SQL that does not work anywhere
- Pay attention to boundary values and exception values such as NULL, 0, ''
- Be careful with units, such as bytes or letters

Hewlett Packard
Enterprise

Thank you

noriyoshi.shinoda@hpe.com