Altibase 7.1.0.7.8 Patch Notes

Table of Contents

New Features

BUG-49782 Adds the ability to generate execution plans similar to the Altibase 6.3.1 optimizer in Altibase 7 and later.

- **module** : qp

- **Category** : Enhancement

- **Reproducibility** : Always

- **Description** : Adds the ability to generate execution plans similar to the Altibase 6.3.1 optimizer in Altibase 7 and later. When using in-memory temporary tables in transactions that access disk tables, HASH joins are chosen instead of index nested loop joins, allowing slow query performance to be tuned with properties.

  This function is provided for a special purpose, so if you want more details, please contact the Altibase Technical Support Center.

- **How to reproduce this bug**

  - **Reproduction conditions**
  - **Actual Results**
  - **Expected Results**
- **Workaround**

- **Changes**

  - Performance view
  - Property
  - Compile Option
  - Error Code

Fixed Bugs

BUG-48012 Add the ST_GEOMETRYFROMTEXT function like the GEOMFROMTEXT function. Change the two functions to return NULL if the value of SRID is NULL.

- **module** : st-function

- **Category** : Functionality

- **Reproducibility** : Always

- **Description** : Add the ST_GEOMETRYFROMTEXT function like the GEOMFROMTEXT function. Change the two functions to return NULL if the value of SRID is NULL.

  The result differs when the query statement meets the conditions before/after bug reflection. For the difference between Actual Results, see Actual Results and Expected Results.

- **How to reproduce this bug**

  - **Reproduction conditions**

    ```
    SELECT ST_ASEWKT(ST_GEOMETRYFROMTEXT( 'POINT(10 20)', 120));
    SELECT ST_ASEWKT(ST_GEOMFROMTEXT( 'POINT(10 20)', NULL));
    ```

  - **Actual Results**

    ```
    iSQL> SELECT ST_ASEWKT(ST_GEOMETRYFROMTEXT( 'POINT(10 20)', 120));
    [ERR-31129 : Procedure or function not found :
    0001 : SELECT ST_ASEWKT(ST_GEOMETRYFROMTEXT( 'POINT(10 20)')
                             ^               ^
    .]

    iSQL> SELECT ST_ASEWKT(ST_GEOMFROMTEXT( 'POINT(10 20)', NULL));
    ST_ASEWKT(ST_GEOMFROMTEXT('POINT(1020)',NU
    --------------------------------------------
    SRID=0;POINT(10 20)
    1 row selected.
    ```

  - **Expected Results**

    ```
    iSQL> SELECT ST_ASEWKT(ST_GEOMETRYFROMTEXT( 'POINT(10 20)', 120));
    ST_ASEWKT(ST_GEOMETRYFROMTEXT('POINT(1020)
    --------------------------------------------
    SRID=120;POINT(10 20)
    1 row selected.

    iSQL> SELECT ST_ASEWKT(ST_GEOMFROMTEXT( 'POINT(10 20)', NULL));
    ST_ASEWKT(ST_GEOMFROMTEXT('POINT(1020)',NU
    --------------------------------------------
    1 row selected.
    ```

- **Workaround**

- **Changes**
  - Performance view
  - Property
  - Compile Option
  - Error Code

BUG-49730 The adapter crashes when performing an UPDATE statement on a table that has more than one LOB column where null data is stored.

- **module** : rp-jdbcAdapter

- **Category** : Testcase Fail

- **Reproducibility** : Always

- **Description** : Fix the abnormal termination of the adapter when an UPDATE statement is executed on a table with two or more LOB columns that store null data.

- **How to reproduce this bug**

  - **Reproduction conditions**
  - **Actual Results**
  - **Expected Results**
- **Workaround**
- **Changes**
  - Performance view
  - Property
  - Compile Option
  - Error Code

BUG-49743 When accessing an index that has not been rebuilt while running the Altibase server, the Altibase server abnormally terminates.

- **module** : sm

- **Category** : Fatal

- **Reproducibility** : Always

- **Description** : Fixes a phenomenon in which the Altibase server abnormally terminates when accessing an index that has not been rebuilt while running the Altibase server. This bug phenomenon When the private property INDEX_REBUILD_AT_STARTUP value is set to 0 and the Altibase server is started, DML accessing the user index is executed It happens. Fixed the following error message to occur when accessing an index that has not been rebuilt during DML execution in the future.

  ERR-111BE: Failed to scan the index because it was not rebuilt. (Index Name : index_name)

  This error message is a newly added error message, see Error Code for details.

- **How to reproduce this bug**

  - **Reproduction conditions**
  - **Actual Results**
  - **Expected Results**

- **Workaround**
- **Changes**
  - Performance view
  - Property
  - Compile Option
  - Error Code

```
0x111BE (  70078) smERR_ABORT_NOT_BUILT_INDEX Failed to scan the index
because it was not rebuilt. (Index Name :<0%s>)
# *Cause: This index was not rebuilt when the Altibase server was
starting up. The value of INDEX_REBUILD_AT_STARTUP property is set to 0.
# *Action: Rebuild this index. Or to rebuild all the indexes, delete
INDEX_REBUILD_AT_STARTUP = 0 in altibase.properties and restart the
Altibase server.
```

BUG-49763 Exception handling is added to analyze the cause of abnormal termination of the Altibase server due to a column information problem in the target table during disk table lookup.

- **module** : sm
- **Category** : Fatal
- **Reproducibility** : Impossible
- **Description** : Altibase server is abnormally terminated if there is an error in the column information of the target table during a disk table query. When this bug occurs, the following error occurs in altibase_error.log.

```
Signal 5(SIGTRAP) caught.
Fault address was 0x0000000100369858
```

Change exception handling to prevent abnormal shutdown of the Altibase server and add exception handling for cause analysis.

- **How to reproduce this bug**
  - **Reproduction conditions**
  - **Actual Results**
  - **Expected Results**
- **Workaround**
- **Changes**
  - Performance view
  - Property
  - Compile Option
  - Error Code

BUG-49766 When changing the REGEXP_MODE property value to 1, the database server character set is checked.

- **module** : qp
- **Category** : Usability
- **Reproducibility** : Always
- **Description** : When changing the REGEXP_MODE property value to 1, the database server character set is checked. If the character set is not supported by PCRE2 library, [ERR-2106B: Unsupported character set by PCRE2 library] error occurs.
- **How to reproduce this bug**
    - **Reproduction conditions**

      ```
      ALTER SESSION SET REGEXP_MODE = 1;
      ```

    - **Actual Results**

      ```
      Alter success.
      ```

    - **Expected Results**

      ```
      ERR-2106B : Unsupported character set by PCRE2 library
      ```

- **Workaround**
- **Changes**
    - Performance view
    - Property
    - Compile Option
    - Error Code

BUG-49769 Delete the unsupported REPLICATION_NET_CONN_IP_STACK property from the altibase.properties.sample file.

- **module** : rp
- **Category** : Maintainability
- **Reproducibility** : Always
- **Description** : Delete the unsupported REPLICATION_NET_CONN_IP_STACK property from the altibase.properties.sample file.

  The deleted part is as follows.

```
# Replication receiver IP stack configuration
# If this property is not set, it will be the same as the value of
# NET_CONN_IP_STACK. If you want to set the IP stackconfiguration
# differently for replication, you need to set this property.
# - REPLICATION_NET_CONN_IP_STACK    = 0 # 0: IPv4 setack only
                                        # 1: Dual stack
                                        # 2: IPv6 stack only
```

- **How to reproduce this bug**
  - **Reproduction conditions**
  - **Actual Results**
  - **Expected Results**
- **Workaround**
- **Changes**
  - Performance view
  - Property
  - Compile Option
  - Error Code

BUG-49778 When disk index is selected and subquery using aggregate function is converted to Subquery Unnesting, result error occurs.

- **module** : qp

- **Category** : Functional Error

- **Reproducibility** : Always

- **Description** : Fixes a phenomenon in which a result error occurs when a disk index is selected and a subquery using an aggregate function is converted to Subquery Unnesting. This is a bug that fixes a result error. When a query that satisfies the condition is executed, the result and execution plan are changed. Please refer to the Actual Results and Expected Results items for the difference between the execution plan before and after the bug fix.

- **How to reproduce this bug**
  - **Reproduction conditions**

    ```sql
    DROP TABLE T2;

    CREATE TABLE T2 (
    "TRADE_DATE" CHAR(8) NOT NULL,
    "DATE_ID"    NUMERIC(10),
    "DAY_TYPE"   NUMERIC(4)
    ) TABLESPACE SYS_TBS_DISK_DATA;

    ALTER TABLE T2 ADD CONSTRAINT "T2_SKSDATEI" PRIMARY KEY("TRADE_DATE");

    INSERT INTO T2 VALUES('19801207', 1, 1);
    INSERT INTO T2 VALUES('20220625', 2, 2);
    INSERT INTO T2 VALUES('20220627', 3, 3);

    ALTER SESSION SET OPTIMIZER_DISK_INDEX_COST_ADJ = 1;
    ```

```
ALTER SESSION SET EXPLAIN PLAN = ON;
ALTER SESSION SET TRCLOG_DETAIL_PREDICATE = 1;

SELECT MAX(TRADE_DATE) FROM T2 WHERE TRADE_DATE < (SELECT
MAX(TRADE_DATE) FROM T2);
```

○ **Actual Results**

```
iSQL> SELECT MAX(TRADE_DATE) FROM T2 WHERE TRADE_DATE < (SELECT
MAX(TRADE_DATE) FROM T2);
MAX(TRADE_DATE)
------------------------------------------------
19801207
1 row selected.
------------------------------------------------------------------
PROJECT ( COLUMN_COUNT: 1, TUPLE_SIZE: 10, COST: 381.30 )
 FILTER
  [ FILTER ]
  $VIEW1.COL1 IS NOT NULL
  VIEW ( ACCESS: 1, COST: 379.98 )
   PROJECT ( COLUMN_COUNT: 2, TUPLE_SIZE: 14, COST: 260.58 )
    WINDOW SORT ( ITEM_SIZE: 24, ITEM_COUNT: 3, DISK_PAGE_COUNT: 0,
ACCESS: 7, SORT_COUNT: 0, COST: 251.34 )
      [ ANALYTIC FUNCTION INFO ]
      SORT_KEY[0]: ()
       MAX(TRADE_DATE) OVER ()
      SCAN ( TABLE: SYS.T2, INDEX: SYS.T2_SKSDATEI, RANGE SCAN, ACCESS:
3, DISK_PAGE_COUNT: 64, COST: 117.82 )
      ------------------------------------------------------------
```

○ **Expected Results**

```
iSQL> SELECT MAX(TRADE_DATE) FROM T2 WHERE TRADE_DATE < (SELECT
MAX(TRADE_DATE) FROM T2);
MAX(TRADE_DATE)
--------------------
20220625
1 row selected.
------------------------------------------------------------------
PROJECT ( COLUMN_COUNT: 1, TUPLE_SIZE: 10, COST: 384.37 )
 GROUP-AGGREGATION ( ITEM_SIZE: 32, GROUP_COUNT: 1, BUCKET_COUNT: 1,
ACCESS: 1, COST: 384.37 )
  FILTER
   [ FILTER ]
   $VIEW1.COL1 IS NOT NULL
   VIEW ( ACCESS: 3, COST: 379.98 )
    PROJECT ( COLUMN_COUNT: 2, TUPLE_SIZE: 14, COST: 260.58 )
     WINDOW SORT ( ITEM_SIZE: 24, ITEM_COUNT: 3, DISK_PAGE_COUNT: 0,
ACCESS: 9, SORT_COUNT: 0, COST: 251.34 )
       [ ANALYTIC FUNCTION INFO ]
       SORT_KEY[0]: ()
        MAX(TRADE_DATE) OVER ()
       SCAN ( TABLE: SYS.T2, INDEX: SYS.T2_SKSDATEI, FULL SCAN, ACCESS:
3, DISK_PAGE_COUNT: 64, COST: 117.82 )
```

```
----------------------------------------------------------------
```

- **Workaround**

  Use the NO_UNNEST hint

  ```
  SELECT MAX(TRADE_DATE) FROM T2
   WHERE TRADE_DATE < (SELECT /*+ NO_UNNEST */ MAX(TRADE_DATE) FROM T2);
  ```

- **Changes**
    - Performance view
    - Property
    - Compile Option
    - Error Code

Changes

Version Info

altibase version	database binary version	meta version	cm protocol version	replication protocol version
7.1.0.7.8	6.5.1	8.10.1	7.1.7	7.4.7

> You can check the module version change history in Version Histories.

Compatibility

Database binary version

The database binary version has not changed.

> The database binary version indicates the compatibility of database image files and log files. If this version needs to be patched to a different version, the database must be reorganized.

Meta Version

The meta version has not changed.

> If you want to roll back the patch after patching to a version with a changed meta version, see Meta Downgrade.

CM protocol Version

The cm protocol version has not changed.

Replication protocol Version

The replication protocol version has not changed.

Altibase Server Properties

Added Properties

Changed Properties

Deleted Properties

Performance Views

Added Performance Views

Changed Performance Views

Deleted Performance Views