

Altibase 7.1.0.6.8 Patch Notes

Table of Contents

- [New Features](#)
 - [BUG-49486 Oracle syntax is added to the EXTRACT function.](#)
 - [BUG-49525 Adds an Altibase server property that sets the size of the VARCHAR type returned by the LISTAGG function.](#)
- [Fixed Bugs](#)
 - [BUG-48650 When a timeout occurs during memory index discovery, the trace log altibasese sm.log displays an error message \[A timeout occuring index fetch.Index \[index name\] link error is expected.\]](#)
 - [BUG-49365 After executing DDL and ALTER REPLICATIN ~ ADD TABLE on the replication target table, \[The row already exists in a unique index.\] error occurs and the replication sender stops.](#)
 - [BUG-49373 \[invalid data type length\] error occurs when inputting CHAR/VARCHAR type data retrieved with the REMOTE TABLE or REMOTE TABLE STORE keyword to the local DB.](#)
 - [BUG-49476 Debug logs are added to analyze the abnormal termination of the Altibase server due to lack of exception handling during transaction commit.](#)
 - [BUG-49489 Invalid size of data to bind to a host variable error occurs if you input data exceeding 32000 bytes in VARCHAR type in the setObject method to a CLOB column.](#)
 - [BUG-49527 \[The tablespace does not have enough free space\] error may occur if the number of extents of the temporary tablespace required in one statement during SQL execution is a multiple of 128.](#)
 - [BUG-49530 Prevents a core dump from being generated when an abnormal shutdown occurs during Altibase server operation.](#)
 - [BUG-49532 In the case of a LEFT OUTER JOIN with an inline view that includes a set operation on the left and the ON clause has a SUBQUERY in the AND condition, the Altibase server will abnormally terminate.](#)
 - [BUG-49538 When executing SQL statements that use temporary tablespaces, the Altibase server may abnormally terminate, an Internal server error error, or an error result may occur.](#)
- [Changes](#)
 - [Version Info](#)
 - [Altibase Server Properties](#)
 - [Performance Views](#)

New Features

BUG-49486 Oracle syntax is added to the EXTRACT function.

- **module** : mt
- **Category** : Enhancement
- **Reproducibility** : Always
- **Description** : Add the following Oracle syntax to the EXTRACT function.

```
EXTRACT( date_field_name FROM expression )
```

date_field_name supports only YEAR, MONTH, DAY, HOUR, MINUTE, SECOND common with Oracle.

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

BUG-49525 Adds an Altibase server property that sets the size of the VARCHAR type returned by the LISTAGG function.

- **module** : mt
- **Category** : Enhancement
- **Reproducibility** : Always
- **Description** : Adds ALTIBASE server property LISTAGG_PRECISION, which sets the size of the return data of the LISTAGG function.

[ERR-21061: result of string concatenation is too long.] error occurs if the returned data size exceeds 4000 bytes due to the size limitation of the VARCHAR type returned by the LISTAGG function.

With the LISTAGG_PRECISION property, you can return from 1 byte to a maximum VARCHAR size of 32000 bytes.

- **How to reproduce this bug**
 - **Reproduction conditions**
 - **Actual Results**
 - **Expected Results**
- **Workaround**
- **Changes**
 - Performance view

- Property

- Add LISTAGG_PRECISION property

data type

Unsigned Integer

default

4000

attributes

Read-write, Single value

Range

[0, 32000]

Description

This property specifies the size of the VARCHAR type returned by the LISTAGG function. The value of this property can be changed with the ALTER SYSTEM statement while Altibase is running.

- Compile Option

- Error Code

Fixed Bugs

BUG-48650 When a timeout occurs during memory index discovery, the trace log altibasese_sm.log displays an error message [A timeout occurring index fetch.Index [index name] link error is expected.]

- **module** : sm-mem-index
- **Category** : Message Error
- **Reproducibility** : Rare
- **Description** : Fixes the occurrence of a [A timeout occurring index fetch.Index [index name] link error is expected.] error message in the trace log altibasese_sm.log even though the index is normal.

If the index is corrupted, the SQL action fails and changes to output the [The maximum number of loops allowed is over 1000001.Index [Index name] link error is expected.] error message in altibase_sm.log.

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

BUG-49365 After executing DDL and ALTER REPLICATIN ~ ADD TABLE on the replication target table, [The row already exists in a unique index.] error occurs and the replication sender stops.

- **module** : rp-control
- **Category** : Usability
- **Reproducibility** : Rare
- **Description** :

In the replication environment set as REPLICATION_DDL_ENABLE = 1, after performing the following 2 tasks sequentially, an ERR-11058 error occurs and replication stops.

1. Execute DDL on replication target table
2. Perform ALTER REPLICATION ~ ADD TABLE. (All ADD TABLE targets include not only DDL-performed tables but also replication target tables)

When this bug occurs, the following log is output in altibase_rp.log.

```
ERR-11058(errno=0) The row already exists in a unique index.  
ERR-610f8(errno=0) [Sender] Failed to add a XLog [TID:-1603452059,  
SN:133280338892, Type:2, Log Type:1, Change Log Type:0]  
ERR-61014(errno=0) [Sender] Failed to make XLOG in a log file at  
SN[133280338892]
```

When this bug occurs, you must DROP and then recreate the replication object.

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

BUG-49373 [invalid data type length] error occurs when inputting CHAR/VARCHAR type data retrieved with the REMOTE_TABLE or REMOTE_TABLE_STORE keyword to the local DB.

- **module** : dblink
- **Category** : Usability
- **Reproducibility** : Always
- **Description** : Fixes a bug in which an invalid data type length error occurs when inputting CHAR/VARCHAR type data retrieved with the REMOTE_TABLE or REMOTE_TABLE_STORE keyword to a local DB.

This bug occurs when all of the conditions below are satisfied.

- Using DB Link
- Use a character set with a character size of 2 bytes or more as a local database character set
- Create identical schemas with CHAR/VARCHAR datatypes in local and remote databases
- Retrieve CHAR/VARCHAR type data from remote database using REMOTE_TABLE or REMOTE_TABLE_STORE keyword
- Enter the queried CHAR/VARCHAR type data into the local database using CURSOR

In this bug, the AltLinker property NLS_BYTE_PER_CHAR was added while improving the method of converting CHAR or VARCHAR type data length between local server and remote server database.

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

```
local:  
CREATE TABLE DBLINK_TEST(ID NUMBER, TEST_DT CHAR(8), TEST_DT2  
VARCHAR(8));
```

```

INSERT INTO DBLINK_TEST values(1,'20211001','20211101');

remote:
CREATE TABLE DBLINK_TEST(ID NUMBER, TEST_DT CHAR(8), TEST_DT2
VARCHAR(8));
INSERT INTO DBLINK_TEST values(1,'20211001','20211101');
INSERT INTO DBLINK_TEST values(2,'20211002','20211102');
CREATE OR REPLACE PROCEDURE SP_DBLINK_TEST
AS
CURSOR CUR1 IS
  SELECT * FROM REMOTE_TABLE(link1,'SELECT * FROM DBLINK_TEST WHERE ID =
  ''2''');
BEGIN
  FOR M IN CUR1
  LOOP
    PRINTLN(length(M.TEST_DT));
    PRINTLN(length(M.TEST_DT2));
    INSERT INTO DBLINK_TEST VALUES(M.ID, M.TEST_DT, M.TEST_DT2);
  END LOOP;
END;
-- result
isql> exec SP_DBLINK_TEST();
16
8
[ERR-21069 : Invalid data type length : TEST_DT.
In SP_DBLINK_TEST
0012 :    INSERT INTO DBLINK_TEST VALUES(M.ID, M.TEST_DT, M.TEST_DT2);

```

◦ Actual Results

```
insert success
```

◦ Expected Results

• Workaround

```
Use with substring() or use varchar type
```

• Changes

- Performance view
- Property
- Compile Option

• Error Code

BUG-49476 Debug logs are added to analyze the abnormal termination of the Altibase server due to lack of exception handling during transaction commit.

- **module** : sm
- **Category** : Other
- **Reproducibility** : Unknown

- **Description** : Debug logs are added to analyze the abnormal termination of the Altibase server due to lack of exception handling during transaction commit. Call stack and transaction information are left in the trace log altibase_error.log.
- **How to reproduce this bug**
 - **Reproduction conditions**
 - **Actual Results**
 - **Expected Results**
- **Workaround**
- **Changes**
 - Performance view
 - Property
 - Compile Option
 - Error Code

BUG-49489 Invalid size of data to bind to a host variable error occurs if you input data exceeding 32000 bytes in VARCHAR type in the setObject method to a CLOB column.

- **module** : mm-jdbc
- **Category** : Functionality
- **Reproducibility** : Always
- **Description** : This is a bug to correct the phenomenon that Invalid size of data to bind to a host variable error occurs when CLOB data is input in Apache NiFi version 1.12.1 or earlier.
- **How to reproduce this bug**
 - **Reproduction conditions**

```
iSQL>CREATE TABLE t1 (c1 int, c2 clob);
Connection sConn = getConnection();
PreparedStatement sPstmt = sConn.prepareStatement("INSERT INTO t1 VALUES
(?, ?)");
sPstmt.setInt(1, 1);
sPstmt.setObject(2, "very long text...", Types.VARCHAR);
sPstmt.addBatch();
sPstmt.executeBatch();
sConn.commit();
```

- **Actual Results**

```
java.sql.BatchUpdateException: Batch update exception occurred:Invalid
size of data to bind to a host variable [ Param ID = 7, Data Type =
MTD_VARCHAR, Data Size = 42954, Declared Size of Host Variable = 32002 ]
```

- **Expected Results**

```
insert without error
```

- **Workaround**


```
PreparedStatement.setObject(int, Object, Types.CLOB);
```

- **Changes**

- Performance view
- Property
- Compile Option
- Error Code

BUG-49527 [The tablespace does not have enough free space] error may occur if the number of extents of the temporary tablespace required in one statement during SQL execution is a multiple of 128.

- **module** : sm-disk-resource
- **Category** : Efficiency
- **Reproducibility** : Always
- **Description** : [The tablespace does not have enough free space] error may occur if the number of extents of the temporary tablespace required in one statement during SQL execution is a multiple of 128. You can check if a bug has occurred by executing the monitoring query below before/after executing the statement that causes the error.

```
SELECT NAME TBS_NAME,  
       ALLOCATED_PAGE_COUNT/EXTENT_PAGE_COUNT AS ALLOCATED_EXTENT_COUNT,  
       CACHED_FREE_EXTENT_COUNT  
FROM X$TABLESPACES I, X$TABLESPACES_HEADER H  
WHERE I.ID = H.ID AND H.TYPE IN (5, 6);
```

If CACHED_FREE_EXTENT_COUNT does not return after executing the statement or if the difference between ALLOCATED_EXTENT_COUNT and CACHED_FREE_EXTENT_COUNT is large, this bug is suspected.

- **How to reproduce this bug**

- **Reproduction conditions**

```
When performing a query that uses the disk temp table
```

- **Actual Results**

```

iSQL> SELECT ALLOCATED_PAGE_COUNT/64, CACHED_FREE_EXTENT_COUNT FROM
X$TABLESPACES WHERE ID ='4';
ALLOCATED_PAGE_COUNT/64 CACHED_FREE_EXTENT_COUNT
-----
5646                      5645
1 row selected.
iSQL> SELECT ... (Query using disk temp table)
iSQL> SELECT ALLOCATED_PAGE_COUNT/64, CACHED_FREE_EXTENT_COUNT FROM
X$TABLESPACES WHERE ID ='4';
ALLOCATED_PAGE_COUNT/64 CACHED_FREE_EXTENT_COUNT
-----
5646                      5517

```

◦ Expected Results

```

iSQL> SELECT ALLOCATED_PAGE_COUNT/64, CACHED_FREE_EXTENT_COUNT FROM
X$TABLESPACES WHERE ID ='4';
ALLOCATED_PAGE_COUNT/64 CACHED_FREE_EXTENT_COUNT
-----
5646                      5645
1 row selected.
iSQL> SELECT ... (Query using disk temp table)
iSQL> SELECT ALLOCATED_PAGE_COUNT/64, CACHED_FREE_EXTENT_COUNT FROM
X$TABLESPACES WHERE ID ='4';
ALLOCATED_PAGE_COUNT/64 CACHED_FREE_EXTENT_COUNT
-----
5646                      5645

```

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

BUG-49530 Prevents a core dump from being generated when an abnormal shutdown occurs during Altibase server operation.

- **module** : id
- **Category** : Maintainability
- **Reproducibility** : Always
- **Description** : Prevents a core dump from being generated when an abnormal shutdown occurs during Altibase server operation. Set the core file size to 0 with setrlimit. The time to call setrlimit is "Initialize Operating System Parameters" during the Altibase server start-up phase.
- **How to reproduce this bug**
 - **Reproduction conditions**
 - **Actual Results**

- Expected Results
- Workaround
- Changes
 - Performance view
 - Property
 - Compile Option
 - Error Code

BUG-49532 In the case of a LEFT OUTER JOIN with an inline view that includes a set operation on the left and the ON clause has a SUBQUERY in the AND condition, the Altibase server will abnormally terminate.

- **module** : qp-dml-pvo
- **Category** : Fatal
- **Reproducibility** : Always
- **Description** : Fixed the bug where the Altibase server abnormally terminates when a LEFT OUTER JOIN includes an inline view with a set operation on the left and the ON clause has a SUBQUERY in the AND condition.
- **How to reproduce this bug**
 - **Reproduction conditions**

```
DROP TABLE T1;
DROP TABLE T2;
DROP TABLE T3;

CREATE TABLE T1 ( I1 VARCHAR(10), I2 VARCHAR(10));
CREATE TABLE T2 ( I1 VARCHAR(10), I2 VARCHAR(10));
CREATE TABLE T3 ( I1 VARCHAR(10), I2 VARCHAR(10), I3 VARCHAR(10));

SELECT A.I1
FROM T1 A
LEFT OUTER JOIN (SELECT I1 , SUM(I2) AS TOT_GWASE_PYOJUN_PRC
                  FROM T2 TA
                  GROUP BY I1 ) C
ON A.I1 = C.I1
AND C.TOT_GWASE_PYOJUN_PRC >= (SELECT I1 FROM T3 ) ;
```

- Actual Results
- Expected Results
- Workaround

```

SELECT A.i1
FROM t1 A
LEFT OUTER JOIN (SELECT i1 , SUM(i2) AS TOT_GWASE_PYOJUN_PRC
                  FROM t2 TA
                  GROUP BY i1 ) C
ON A.i1 = C.i1
WHERE C.TOT_GWASE_PYOJUN_PRC >= (SELECT i1 FROM t3 ) ;

```

- **Changes**

- Performance view
- Property
- Compile Option
- Error Code

BUG-49538 When executing SQL statements that use temporary tablespaces, the Altibase server may abnormally terminate, an Internal server error error, or an error result may occur.

- **module** : sm
- **Category** : Functional Error
- **Reproducibility** : Always
- **Description** : Fix the issue where the Altibase server may abnormally terminate, throw an Internal server error, or return an error result when executing SQL statements that use temporary tablespaces.

This bug occurs with a very low probability if all of the following conditions are met.

- Altibase 7.1.0.5.1 or higher
- When a disk table is included in the SQL statement and the GROUP BY clause is used
- Perform SORT operation during disk temp table operation
- When using a disk temporary tablespace due to insufficient SORT_AREA_SIZE space

If this issue occurs, the following error will be logged in altibase_error.log.

```

IDE_ERROR( sisValid == ID_TRUE )[sdtSortSegment.h:751], errno=[11]
[FAILURE] ERR-0109E(error=11) Internal server error.

```

- **How to reproduce this bug**

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

- **Workaround**

- **Changes**

- Performance view
- Property
- Compile Option
- Error Code

Changes

Version Info

altibase version	database binary version	meta version	cm protocol version	replication protocol version
7.1.0.6.8	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.

Meta Version

The meta version has not changed.

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

- LISTAGG_PRECISION

Changed Properties

Deleted Properties

Performance Views

Added Performance Views

Changed Performance Views

Deleted Performance Views