

Altibase 7.1.0.7.2 Patch Notes

Table of Contents

- [Altibase 7.1.0.7.2 Patch Notes](#)
- [New Features](#)
 - [BUG-49567 Adapter for Oracle에서 지원하는 데이터 타입에 LOB을 추가합니다.](#)
 - [BUG-49616 Altibase 7.1 Standard Edition, Enterprise Edition에서 라이선스 발급 기준으로 MEM MAX DB SIZE를 추가합니다.](#)
- [Fixed Bugs](#)
 - [BUG-49546 DatabaseMetaData.getColumns 메서드의 반환 결과 중 DATA TYPE의 값을 정의하는 연결 속성 getcolumns return jdbcType을 추가합니다.](#)
 - [BUG-49549 파티션드 테이블에 사용한 집계 함수가 병렬 수행될 경우 Altibase 서버가 비정상 종료할 수 있습니다.](#)
 - [BUG-49559 복합 인덱스가 있는 테이블에 ROW NUMBER 함수를 사용한 경우 중복된 정렬 작업을 제거하여 SQL 수행 성능을 개선합니다.](#)
 - [BUG-49601 User Defined Type을 지원하지 않을 때 DatabaseMetaData.getUDTs\(\)에서 SQLFeatureNotSupportedException 예외 대신 빈 ResultSet을 반환하도록 변경합니다.](#)
 - [BUG-49602 Updatable ResultSet 객체의 결과가 0건인 경우에도 ResultSet.insertRow\(\) 메소드가 동작해야 합니다.](#)
 - [BUG-49603 SQL 문에 데이터베이스 소유자가 명시된 경우 Updatable ResultSet 객체에서 java.sql.SQLException: Table or view was not found 에러가 발생합니다.](#)
 - [BUG-49605 Updatable ResultSet을 이용해 데이터 입력 시 Unsupported type conversion SQLException 에러가 발생하는 현상을 수정합니다.](#)
 - [BUG-49613 = 연산자와 CAST 연산자에 사용된 서브쿼리의 결과가 NULL일 때 Altibase 서버가 비정상 종료할 수 있습니다.](#)
 - [BUG-49619 디스크 인덱스에서 언두 테이블스페이스를 사용하는 경우 다른 트랜잭션에 의해 언두 영역이 재사용되어 Altibase 서버가 비정상 종료할 수 있습니다.](#)
 - [BUG-49620 NVL EQUAL\(expr1, expr2, expr3\)에서 expr1에 인덱스가 존재하고 expr2의 컬럼의 데이터 타입이 expr1과 다른 경우 테이블 스캔 방식을 변경하여 제품의 안정성을 향상합니다.](#)
 - [BUG-49632 반환 데이터 타입이 LOB인 저장 함수가 ORDER BY/GROUP BY/윈도우 함수의 PARTITION BY 하위절에 사용될 때 예외 처리를 변경하여 제품의 안정성을 향상합니다.](#)
 - [BUG-49633 집계 함수에 DISTINCT 키워드를 사용한 SQL 수행 시 DISTINCT 처리를 위한 메모리 과다 사용으로 SQL 수행 성능이 하락합니다.](#)
 - [BUG-49636 Altibase 서버 프로퍼티 ARCHIVE FULL ACTION 설정 값에 따른 아카이브로그 쓰레드의 동작을 개선합니다.](#)
 - [BUG-49647 Ubuntu 16, Ubuntu 18에서 altiMon 실행 시 com.altibase.picl.LibLoader.OsNotSupportedException 에러가 발생합니다.](#)
 - [BUG-49655 디스크 테이블 및 디스크 인덱스에서 사용 중인 데이터 페이지수를 조회할 수 있도록 X\\$SEGMENT에 TOTAL USED PAGE CNT 컬럼을 추가합니다.](#)
- [Changes](#)
 - [Version Info](#)
 - [호환성](#)
 - [프로퍼티](#)
 - [성능 분](#)

New Features

BUG-49567 Adapter for Oracle에서 지원하는 데이터 타입에 LOB을 추가합니다.

- **module** : rp-oraAdapter
- **Category** : Enhancement
- **재현 빈도** : Always
- **설명** : Adapter for Oracle에서 지원하는 데이터 타입에 LOB을 추가합니다.

LOB 데이터 타입 지원 관련한 Adapter for Oracle 동작 및 제약 사항은 Adapter for Oracle 매뉴얼을 참고하시기 바랍니다.

[Adapter for Oracle User's Manual - 2.설치와 설정 - 프로퍼티 - ORACLE ARRAY DML MAX SIZE](#)

[Adapter for Oracle User's Manual - 2.설치와 설정 - 프로퍼티 - ORACLE ERROR RETRY COUNT](#)

[Adapter for Oracle User's Manual - 2.설치와 설정 - 프로퍼티 - ORACLE SKIP ERROR](#)

[Adapter for Oracle User's Manual - 2.설치와 설정 - 프로퍼티 - ADAPTER LOB TYPE SUPPORT](#)

[Adapter for Oracle User's Manual - 3.사용법 - oraAdapter 제약조건 - LOB 데이터 타입 제약 사항](#)

- **재현 방법**
 - 재현 절차
 - 수행 결과
 - 예상 결과
- **Workaround**
- **변경사항**
 - Performance view
 - Property
 - Compile Option
 - Error Code

BUG-49616 Altibase 7.1 Standard Edition, Enterprise Edition에서 라이선스 발급 기준으로 MEM_MAX_DB_SIZE를 추가합니다.

- **module** : id
- **Category** : Enhancement
- **재현 빈도** : Unknown
- **설명** : Altibase 7.1 Standard Edition, Enterprise Edition에서 라이선스 발급 기준으로 MEM_MAX_DB_SIZE를 추가합니다.
- **재현 방법**
 - 재현 절차
 - 수행 결과
 - 예상 결과
- **Workaround**
- **변경사항**

- Performance view
- Property
- Compile Option
- Error Code

Fixed Bugs

BUG-49546 DatabaseMetaData.getColumns 메서드의 반환 결과 중 DATA_TYPE의 값을 정의하는 연결 속성 getcolumns_return_jdbctype을 추가합니다.

- **module** : mm-jdbc
 - **Category** : Functional Error
 - **재현 빈도** : Always
 - **설명** : DatabaseMetaData.getColumns 메서드의 반환 결과 중 DATA_TYPE의 값을 정의하는 연결 속성 [getcolumns_return_jdbctype](#)을 추가합니다.
 - 연결 속성 이름
getcolumns_return_jdbctype
 - 설명
DatabaseMetaData.getColumns 메서드의 반환 결과 중 DATA_TYPE의 값을 정의한다.
true는 JDBC API에서 정의한 java.sql.Types의 SQL 데이터 형식으로 반환하고 false는 V\$DATATYPE에 정의된 데이터 타입 형식으로 반환한다.
 - 기본값
false
- 본 버그를 적용하려면 Altibase JDBC 드라이버를 Altibase 7.1.0.7.2 이상으로 패치하고 연결 속성 getcolumns_return_jdbctype을 추가해야 합니다.
- **재현 방법**
 - **재현 절차**

```
isQL> CREATE TABLE T1 (C1 CLOB, C2 BLOB);
```

```
Connection sConn = getConnection("20300");
DatabaseMetaData sMeta = sConn.getMetaData();
ResultSet sRs = sMeta.getColumns(null, "SYS", "T1", "%");
while (sRs.next())
{
    String sColumnName = sRs.getString("COLUMN_NAME");
    String sDataType = sRs.getString("DATA_TYPE");
    System.out.println("COLUMN_NAME==>" + sColumnName);
    System.out.println("DATA_TYPE==>" + sDataType);
}
```

- **수행 결과**

```

COLUMN_NAME==>C1
DATA_TYPE==>40
COLUMN_NAME==>C2
DATA_TYPE==>30

```

◦ 예상 결과

```

COLUMN_NAME==>C1
DATA_TYPE==>2005
COLUMN_NAME==>C2
DATA_TYPE==>2004

```

- Workaround

- 변경사항

- Performance view
- Property
- Compile Option
- Error Code

BUG-49549 파티션드 테이블에 사용한 집계 함수가 병렬 수행될 경우 Altibase 서버가 비정상 종료할 수 있습니다.

- module : qp
- Category : Fatal
- 재현 빈도 : Frequence
- 설명 : 파티션드 테이블에 사용한 집계 함수가 병렬 수행될 경우 Altibase 서버가 비정상 종료하는 현상을 수정합니다.
- 재현 방법
 - 재현 절차

```

DROP TABLE t1 cascade;
CREATE TABLE T1( I1 INTEGER, I2 INTEGER ) PARTITION BY HASH(I1)
( PARTITION P1, PARTITION P2 );
ALTER TABLE T1 PARALLEL 2;
VAR v1 integer;
PREPARE SELECT /*+ no_plan_cache */ COUNT(*) FROM T1 WHERE (I1=:v1 OR I1=999) AND
(I2=1 OR I2=2);

```

◦ 수행 결과

Altibase 서버 비정상 종료

◦ 예상 결과

```
COUNT(*)
-----
0
1 row selected.
```

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

BUG-49559 복합 인덱스가 있는 테이블에 ROW_NUMBER 함수를 사용한 경우 중복된 정렬 작업을 제거하여 SQL 수행 성능을 개선합니다.

- **module** : qp
- **Category** : Enhancement
- **재현 빈도** : Always
- **설명** : 복합 인덱스가 있는 테이블에 ROW_NUMBER 함수를 사용한 경우 중복된 정렬 작업을 피하고자 Preserved Order 기능을 추가하여 SQL 수행 성능을 개선하였습니다.

참고 : Preserved Order 기능

Altibase는 이미 정렬된 테이블을 ORDER BY 구문으로 똑같이 정렬시킬 때 중복된 정렬 작업을 피하기 위해 Preserved Order라는 개념을 사용합니다. 예를 들어 테이블 t1에 i1 컬럼이 Ascending 정렬된 인덱스를 가지고 있을 때, i1 컬럼을 ORDER BY 구문으로 Ascending 정렬하면 따로 정렬 작업을 수행하지 않고 그대로 읽도록 하는 개념입니다.

관련하여 히든 프로퍼티 __OPTIMIZER_ROW_NUMBER 가 추가되었습니다. 히든 프로퍼티로 매뉴얼에 설명을 추가하지 않습니다.

- 이름
 - __OPTIMIZER_ROW_NUMBER
- 설명
 - 본 버그에서 추가된 기능 사용 여부를 결정합니다.
- 값
 - 0 : Preserved Order 기능을 사용하지 않음.
 - 1 : Preserved Order 기능을 사용함.
- 기본값
 - 0
- 속성
 - 변경 가능, 비공개

Altibase 6.5.1, Altibase 7.1에서 이전 버전과의 호환성을 위해 이 프로퍼티 기본값 0 으로 개선한 기능 사용이 비활성화되어 있습니다. Altibase 6.5.1.9.1 이상, Altibase 7.1.0.7.2 이상에서 이 버그에서 개선한 기능을 사용하려면 이 프로퍼티 값을 1로 설정해야 합니다. 프로퍼티 설정값을 1로 설정하면, 이 버그의 영향을 받은 SQL 문의 실행 계획에서 WINDOW SORT 노드의 SORT_COUNT 수가 줄어듭니다.

이 프로퍼티 변경은 ALTER SYSTEM 으로 변경할 수 있습니다. 기본값과 다른 값을 영구 적용하고자 할 경우 altibase.properties 에 추가하고 Altibase 서버를 재시작해야 합니다.

- 재현 방법

- 재현 절차

```
DROP TABLE T1;
CREATE TABLE T1 ( I1 INT, I2 INT, I3 INT, I4 INT);
DROP INDEX IDX1;
CREATE INDEX IDX1 ON T1 ( I1, I2, I3 DESC );
ALTER SESSION SET EXPLAIN PLAN = ON;
SELECT ROW_NUMBER() OVER ( ORDER BY I3 DESC ), * FROM T1 WHERE I1 = 1 AND I2 = 2;
```

- 수행 결과

```
-----
PROJECT ( COLUMN_COUNT: 5, TUPLE_SIZE: 24, COST: 0.26 )
  WINDOW SORT ( ITEM_SIZE: 24, ITEM_COUNT: 0, ACCESS: 0, SORT_COUNT: 1, COST: 0.19
)
    SCAN ( TABLE: SYS.T1, INDEX: SYS.IDX1, RANGE SCAN, ACCESS: 0, COST: 0.01 )
-----
```

- 예상 결과

SORT_COUNT 가 줄어듭니다.

```
-----
PROJECT ( COLUMN_COUNT: 5, TUPLE_SIZE: 24, COST: 0.26 )
  WINDOW SORT ( ITEM_SIZE: 24, ITEM_COUNT: 0, ACCESS: 0, SORT_COUNT: 0, COST: 0.19
)
    SCAN ( TABLE: SYS.T1, INDEX: SYS.IDX1, RANGE SCAN, ACCESS: 0, COST: 0.01 )
-----
```

- Workaround

- 변경사항

- Performance view
 - Property
 - Compile Option
 - Error Code

BUG-49601 User Defined Type을 지원하지 않을 때 DatabaseMetaData.getUDTs()에서 SQLFeatureNotSupportedException 예외 대신 빈 ResultSet을 반환하도록 변경합니다.

- **module** : mm-jdbc
- **Category** : Functional Error
- **재현 빈도** : Always
- **설명** : JDBC specification에 따라 User Defined Type을 지원하지 않을 때 DatabaseMetaData.getUDTs()에서 SQLFeatureNotSupportedException 예외 대신 빈 ResultSet을 반환하도록 변경합니다.
 - 참고 : <https://docs.oracle.com/javame/config/cdc/opt-pkgs/api/jsr169/java/sql/DatabaseMetaData.html>

본 버그를 적용하려면 Altibase JDBC 드라이버를 7.1.0.7.2 이상으로 패치해야 합니다.

- **재현 방법**
 - **재현 절차**

```
Connection sConn = getConnection("20300");
DatabaseMetaData sMeta = sConn.getMetaData();
ResultSet sRs = sMeta.getUDTs(null, null, null, null);
System.out.println("sRs.next()==>" + sRs.next());
```

- **수행 결과**

```
Exception in thread "main" java.sql.SQLFeatureNotSupportedException: User defined type
```

- **예상 결과**

```
sRs.next()==>false
```

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

BUG-49602 Updatable ResultSet 객체의 결과가 0건인 경우에도 ResultSet.insertRow() 메소드가 동작해야 합니다.

- **module** : mm-jdbc
- **Category** : Functional Error
- **재현 빈도** : Always

- **설명** : Updatable ResultSet 객체의 결과가 0건인 경우에도 ResultSet.insertRow() 메소드가 동작하지 않는 현상을 개선합니다.

이 버그의 영향을 받는 ResultSet 객체의 스크롤 및 업데이트 허용 옵션은 아래와 같습니다.

- 스크롤 옵션
 - TYPE_FORWARD_ONLY
 - TYPE_SCROLL_INSENSITIVE
- 업데이트 허용 옵션
 - CONCUR_UPDATABLE

이 버그를 적용하려면 Altibase JDBC 드라이버를 Altibase 7.1.0.7.2 이상으로 패치해야 합니다.

- **재현 방법**

- **재현 절차**

```
iSQL> CREATE TABLE T1 (C1 INT, C2 VARCHAR(10));
```

```
Connection sConn = getConnection("20300");
Statement sStmt = sConn.createStatement(ResultSet.TYPE_FORWARD_ONLY,
                                           ResultSet.CONCUR_UPDATABLE);
ResultSet sRs = sStmt.executeQuery("SELECT c1, c2 FROM t1");
sRs.moveToInsertRow();
sRs.updateObject(1, 1);
sRs.updateObject(2, "test..");
sRs.insertRow();
```

```
iSQL> SELECT * FROM t1;
```

- **수행 결과**

```
iSQL> SELECT * FROM t1;
C1          C2
-----
No rows selected.
```

- **예상 결과**

```
iSQL> SELECT * FROM t1;
C1          C2
-----
1          test..
1 row selected.
```

- **Workaround**

- **변경사항**

- Performance view
- Property
- Compile Option

- Error Code

BUG-49603 SQL 문에 데이터베이스 소유자가 명시된 경우 Updatable ResultSet 객체에서 java.sql.SQLException: Table or view was not found 에러가 발생합니다.

- **module** : mm-jdbc
- **Category** : Functional Error
- **재현 빈도** : Always
- **설명** : SQL 문에 데이터베이스 소유자가 명시된 경우 Updatable ResultSet 객체에서 java.sql.SQLException: Table or view was not found 에러가 발생하는 현상을 수정합니다.

Updatable ResultSet 객체 : 업데이트 허용(CONCUR_UPDATABLE) 옵션을 사용한 ResultSet 객체

본 버그를 적용하려면 Altibase JDBC 드라이버를 7.1.0.7.2 이상으로 패치해야 합니다.

- **재현 방법**
 - 재현 절차

```
isQL> CREATE USER altitest IDENTIFIED BY altitest;
Create success.
isQL> CONNECT altitest;
Write Password :
Connect success.
isQL> CREATE TABLE T2 (C1 INT, C2 VARCHAR(10));
Create success.
```

```
Connection sConn = getConnection("20300"); // connect with SYS
Statement sStmt = sConn.createStatement(ResultSet.TYPE_FORWARD_ONLY,
                                           ResultSet.CONCUR_UPDATABLE);
ResultSet sRs = sStmt.executeQuery("SELECT c1, c2 FROM altitest.t2");
sRs.moveToInsertRow();
sRs.updateObject(1, 1);
sRs.updateObject(2, "test..");
sRs.insertRow();
```

- 수행 결과

```
Exception in thread "main" java.sql.SQLException: Table or view was not found :
0001 : DELETE FROM T2 WHERE _PROWID=?
      ^ ^
      at Altibase.jdbc.driver.ex.Error.processServerError(Error.java:397)
      at
Altibase.jdbc.driver.AltibasePreparedStatement.prepare(AltibasePreparedStatement.
java:139)
      at Altibase.jdbc.driver.AltibasePreparedStatement.
(AltibasePreparedStatement.java:94)
      at
Altibase.jdbc.driver.AltibaseConnection.prepareStatement(AltibaseConnection.java:
1346)
      at Altibase.jdbc.driver.AltibaseUpdatableResultSet.
(AltibaseUpdatableResultSet.java:66)
      at
Altibase.jdbc.driver.AltibaseResultSet.createResultSet(AltibaseResultSet.java:90)
      at
Altibase.jdbc.driver.AltibaseStatement.processExecutionQueryResult(AltibaseStatem
ent.java:414)
      at
Altibase.jdbc.driver.AltibaseStatement.executeQuery(AltibaseStatement.java:754)
```

◦ 예상 결과

정상적으로 insert 성공

- Workaround
- 변경사항
 - Performance view
 - Property
 - Compile Option
 - Error Code

BUG-49605 Updatable ResultSet을 이용해 데이터 입력 시 Unsupported type conversion SQLException 에러가 발생하는 현상을 수정합니다.

- **module** : mm-jdbc
- **Category** : Functional Error
- **재현 빈도** : Always
- **설명** : Updatable ResultSet 객체를 이용해 데이터 입력 시 Unsupported type conversion SQLException 에러가 발생하는 현상을 수정합니다.

Updatable ResultSet 객체 : 업데이트 허용(CONCUR_UPDATABLE) 옵션을 사용한 ResultSet 객체

본 버그를 적용하려면 Altibase JDBC 드라이버를 7.1.0.7.2 이상으로 패치해야 합니다.

- **재현 방법**
 - 재현 절차

```
CommonBinaryColumn.setValueSub(Duration);
CommonBinaryColumn.setValueSub(ByteArrayInputStream);
CommonCharVarcharColumn.setValueSub(Clob);
CommonCharVarcharColumn.setValueSub(StringReader);
CommonCharVarcharColumn.setValueSub(ByteArrayInputStream)FloatColumn.getObjectSub();
TimestampColumn.setValueSub(Time);
```

○ 수행 결과

Unsupported type conversion SQLException 발생

○ 예상 결과

정상 처리

• Workaround

• 변경사항

- Performance view
- Property
- Compile Option
- Error Code

BUG-49613 = 연산자와 CAST 연산자에 사용된 서브쿼리의 결과가 NULL일 때 Altibase 서버가 비정상 종료할 수 있습니다.

- **module** : qp
- **Category** : Fatal
- **재현 빈도** : Always
- **설명** : = 연산자와 CAST 연산자에 사용된 서브쿼리의 결과가 NULL일 때 Altibase 서버가 비정상 종료하는 현상을 수정합니다.
- **재현 방법**
 - 재현 절차

```
CREATE TABLE T1 ( I1 INTEGER, I2 INTEGER, I3 INTEGER ) TABLESPACE
SYS_TBS_DISK_DATA;
INSERT INTO T1 VALUES (1, 1, NULL);
CREATE TABLE T2 ( I1 INTEGER ) TABLESPACE SYS_TBS_DISK_DATA;
SELECT * FROM T1 WHERE I1 = (SELECT DISTINCT I1 || I2 FROM T2);
```

○ 수행 결과

[ERR-91015 : Communication failure.]

○ 예상 결과

No rows selected.

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

BUG-49619 디스크 인덱스에서 언두 테이블스페이스를 사용하는 경우 다른 트랜잭션에 의해 언두 영역이 재사용되어 Altibase 서버가 비정상 종료할 수 있습니다.

- **module** : sm-disk-index
- **Category** : Fatal
- **재현 빈도** : Rare
- **설명** : 디스크 인덱스에서 언두 테이블스페이스를 사용하는 경우 다른 트랜잭션에 의해 언두 영역이 재사용되어 Altibase 서버가 비정상 종료하는 현상을 수정합니다.

본 버그로 인한 Altibase 서버 비정상 종료 발생 시 트레이스 로그에 아래와 같은 로그가 기록됩니다.

- altibase_error.log

```
[ASSERT] ERROR LINE => [sdnIndexCTL.cpp:2208]
```

- altibase_boot.log 에 아래와 같은 형태의 인덱스 및 언두 페이지 정보 기록

```
[2022/02/09 16:42:10] [Thread-28749] [Level-0]
Undo PageID           : 56023
Undo Slot Number      : 91
UndoRec Hdr Type      : 2
Chained CTS's Commit SCN : 61801506952
Min DskView SCN       : 61801348159
```

- **재현 방법**
 - 재현 절차
 - 수행 결과
 - 예상 결과
- **Workaround**
- **변경사항**
 - Performance view
 - Property
 - Compile Option
 - Error Code

BUG-49620 NVL_EQUAL(expr1, exp2, expr3)에서 expr1에 인덱스가 존재하고 expr2의 컬럼의 데이터 타입이 expr1과 다른 경우 테이블 스캔 방식을 변경하여 제품의 안정성을 향상합니다.

- **module** : qp
- **Category** : Fatal
- **재현 빈도** : Always
- **설명** : NVL_EQUAL(expr1, exp2, expr3)에서 expr1에 인덱스가 존재하고 expr2의 컬럼의 데이터 타입이 expr1과 다른 경우 Altibase 서버가 비정상 종료하는 현상을 수정합니다.
- **재현 방법**
 - 재현 절차

```
CREATE TABLE T1( I1 INTEGER, I2 VARCHAR(10) ) PARTITION BY HASH( I1 ) ( PARTITION
P1, PARTITION P2 TABLESPACE SYS_TBS_MEM_DATA, PARTITION P3 ) TABLESPACE
SYS_TBS_DISK_DATA;
CREATE INDEX IDX1 ON T1(I1) LOCAL;
SELECT * FROM T1 WHERE NVL_EQUAL(I1,I2,2);
```

- 수행 결과

Altibase 서버 비정상 종료

ALTER SESSION SET EXPLAIN PLAN = ONLY; 수행 후 SELECT 수행 시 실행 계획

```
-----
PROJECT ( COLUMN_COUNT: 2, TUPLE_SIZE: 16, COST: 58.66 )
PARTITION-COORDINATOR ( TABLE: SYS.T1, PARTITION: 3/3, ACCESS: ??, COST: 57.87 )
SCAN ( PARTITION: P3, INDEX: SYS.IDX1, RANGE SCAN, ACCESS: ??, COST: 23.09 )
SCAN ( PARTITION: P2, INDEX: SYS.IDX1, RANGE SCAN, ACCESS: ??, COST: 11.69 )
SCAN ( PARTITION: P1, INDEX: SYS.IDX1, RANGE SCAN, ACCESS: ??, COST: 23.09 )
-----
```

- 예상 결과

I1 I2

No rows selected.

ALTER SESSION SET EXPLAIN PLAN = ONLY; 수행 후 SELECT 수행 시 실행 계획

```
-----
PROJECT ( COLUMN_COUNT: 2, TUPLE_SIZE: 16, COST: 386.68 )
PARTITION-COORDINATOR ( TABLE: SYS.T1, PARTITION: 3/3, FULL SCAN, ACCESS: 0,
COST: 385.88 )
SCAN ( PARTITION: P3, FULL SCAN, ACCESS: 0, DISK_PAGE_COUNT: 64, COST: 129.28 )
SCAN ( PARTITION: P2, FULL SCAN, ACCESS: 0, COST: 127.32 )
SCAN ( PARTITION: P1, FULL SCAN, ACCESS: 0, DISK_PAGE_COUNT: 64, COST: 129.28 )
-----
```

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

BUG-49632 반환 데이터 타입이 LOB인 저장 함수가 ORDER BY/GROUP BY/윈도우 함수의 PARTITION BY 하위절에 사용될 때 예외 처리를 변경하여 제품의 안정성을 향상합니다.

- **module** : qp
- **Category** : Fatal
- **재현 빈도** : Always
- **설명** : 반환 데이터 타입이 LOB인 저장 함수가 ORDER BY/GROUP BY/윈도우 함수의 PARTITION BY 하위절에 사용될 때 Altibase 서버가 비정상 종료하는 현상을 개선합니다.
- **재현 방법**
 - **재현 절차**

```
CREATE TABLE T1( I1 CHAR(10), I2 CHAR(10), I3 CHAR(10), I4 CHAR(10) ) TABLESPACE
SYS_TBS_MEM_DATA;
CREATE FUNCTION FUNC1
(V1 CLOB)
RETURN CLOB
AS
    V2 CLOB;
BEGIN
    V2 := V1;
    RETURN V2;
END;
/
INSERT INTO T1(I2) VALUES (637);
SELECT GROUPING_ID(FUNC1(I1))
FROM T1
GROUP BY GROUPING SETS(1), FUNC1(I1);
```

- **수행 결과**

Altibase 서버 비정상 종료

- **예상 결과**

[ERR-3123E : An incomparable data type (GEOMETRY,LOB) cannot be used in a SELECT statement that has a DISTINCT clause.]

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

BUG-49633 집계 함수에 DISTINCT 키워드를 사용한 SQL 수행 시 DISTINCT 처리를 위한 메모리 과다 사용으로 SQL 수행 성능이 하락합니다.

- **module** : qp-select
- **Category** : Enhancement
- **재현 빈도** : Always
- **설명** :

집계 함수에 DISTINCT 키워드를 사용한 SQL 수행 시 DISTINCT 처리를 위한 해시 버킷 수 과다 사용으로 메모리 사용이 증가하여 SQL 수행 성능이 하락하는 현상을 개선합니다. 본 버그 현상이 발생하는 경우 운영 환경에 적합한 해시 버킷 수를 설정할 수 있는 프로퍼티를 제공합니다. 하든 프로퍼티로 매뉴얼에 설명을 추가하지 않습니다.

- 프로퍼티 추가

- 이름

__AGGREGATION_DISTINCT_BUCKET_COUNT_MAX

- 설명

집계 함수에서 DISTINCT 처리를 위한 최대 해시 버킷 수를 지정하는 프로퍼티이다.

- 값

1 ~ 102400000

- 기본값

102400000

- 속성

변경 가능, 비공개

또한, SQL에서 사용된 해시 버킷 수를 실행 계획에 표시합니다. 이 값을 참고하여 __AGGREGATION_DISTINCT_BUCKET_COUNT_MAX 설정할 수 있습니다. DISTINCT 키워드가 집계 함수, OVER, GROUP BY ROLLUP, GROUP BY CUBE와 함께 사용한 SQL의 실행 계획에서 확인할 수 있습니다.

- 영향을 받는 SQL 예시

```
SELECT COUNT(DISTINCT i1) FROM t1;
SELECT SUM( DISTINCT I2 ) OVER ( PARTITION BY I1 ) FROM T1;
SELECT SUM( DISTINCT I2 ) FROM T1 GROUP BY ROLLUP ( I1, I3, I4 );
SELECT SUM( DISTINCT I2 ) FROM T1 GROUP BY CUBE ( I1, I3, I4 );
```

__AGGREGATION_DISTINCT_BUCKET_COUNT_MAX 프로퍼티의 기본값은 102400000 입니다. 운영 중 이보다 작게 설정하기를 원할 경우 ALTER SYSTEM 으로 변경할 수 있습니다. 영구적으로 작은 값을 설정하려면 altibase.properties에 추가한 후 Altibase 서버를 재구동해야 합니다.

- 재현 방법
 - 재현 절차
 - 수행 결과
 - 예상 결과
- Workaround
- 변경사항
 - Performance view
 - Property
 - Compile Option
 - Error Code

BUG-49636 Altibase 서버 프로퍼티 ARCHIVE_FULL_ACTION 설정 값에 따른 아카이브로그 쓰레드의 동작을 개선합니다.

- **module** : sm
- **Category** : Functionality
- **재현 빈도** : Always
- **설명** : 디스크 공간 부족으로 로그 파일 백업이 실패하는 경우 ARCHIVE_FULL_ACTION 설정 값에 따른 아카이브로그 쓰레드의 동작을 개선합니다.

ARCHIVE_FULL_ACTION 설정 값에 따른 동작 차이

- ARCHIVE_FULL_ACTION = 0
 - 변경 전 : 디스크 공간 부족 발생으로 로그 파일 백업이 실패하는 경우 백업이 성공할 때까지 대기합니다.
 - 변경 후 : 디스크 공간 부족 발생으로 로그 파일 백업이 실패하는 경우 트레이스 로그 (altibase_sm.log)에 기록하고 다음 로그 파일 백업을 시도합니다.
- ARCHIVE_FULL_ACTION = 1
 - 변경이 없습니다.
- ARCHIVE_FULL_ACTION = 2
 - 설정값 2가 추가되었습니다. 디스크 공간 부족 실패 외에 다른 이유로 로그 파일 백업이 실패하는 경우 트레이스 로그(altibase_sm.log)에 에러 메시지를 출력하고 다음 로그 파일의 백업을 시도합니다.

ARCHIVE_FULL_ACTION 속성 변경

읽기 전용에서 변경 가능으로 변경합니다.

- 변경 전

```
iSQL> ALTER SYSTEM SET ARCHIVE_FULL_ACTION = 2;
[ERR-0104E : The property [ARCHIVE_FULL_ACTION] is read-only.]
```

- 변경 후

```
iSQL> ALTER SYSTEM SET ARCHIVE_FULL_ACTION = 2;
Alter success
```

ARCHIVE_FULL_ACTION 설정 값에 관한 설명은 [General Reference-1.Data Types & Altibase Properties](#)에서도 확인할 수 있습니다.

- 재현 방법
 - 재현 절차
 - 수행 결과
 - 예상 결과
- Workaround
- 변경사항
 - Performance view
 - Property
 - Compile Option
 - Error Code

BUG-49647 Ubuntu 16, Ubuntu 18에서 altiMon 실행 시 com.altibase.picl.LibLoader.OsNotSupportedException 에러가 발생합니다.

- **module** : ux-altiMon
- **Category** : Portability
- **재현 빈도** : Always
- **설명** : Linux kernel 4 미지원으로 Ubuntu 16, Ubuntu 18에서 altiMon 실행 시 com.altibase.picl.LibLoader.OsNotSupportedException 에러가 발생합니다.
altiMon에서 Linux kernel 4.x 지원하도록 추가합니다.
- 재현 방법
 - 재현 절차
 - 수행 결과
 - 예상 결과
- Workaround
- 변경사항
 - Performance view
 - Property
 - Compile Option
 - Error Code

BUG-49655 디스크 테이블 및 디스크 인덱스에서 사용 중인 데이터 페이지 수를 조회할 수 있도록 X\$SEGMENT에 TOTAL_USED_PAGE_CNT 컬럼을 추가합니다.

- **module** : sm
- **Category** : Other
- **재현 빈도** : Always

- **설명** : 디스크 테이블 및 디스크 인덱스에서 사용 중인 데이터 페이지 수를 조회할 수 있도록 X\$SEGMENT에 TOTAL_USED_PAGE_CNT 컬럼을 추가합니다. TOTAL_USED_PAGE_CNT 컬럼은 메타 페이지와 FREE 페이지를 제외한 데이터만 있는 페이지 수를 의미합니다.
- **재현 방법**
 - 재현 절차
 - 수행 결과
 - 예상 결과
- **Workaround**
- **변경사항**
 - Performance view
 - Property
 - Compile Option
 - Error Code

Changes

Version Info

altibase version	database binary version	meta version	cm protocol version	replication protocol version
7.1.0.7.2	6.5.1	8.10.1	7.1.7	7.4.7

Altibase 7.1 패치 버전별 히스토리는 [Version Histories](#) 에서 확인할 수 있다.

호환성

Database binary version

데이터베이스 바이너리 버전은 변경되지 않았다.

데이터베이스 바이너리 버전은 데이터베이스 이미지 파일과 로그파일의 호환성을 나타낸다. 이 버전이 다른 경우의 패치(업그레이드 포함)는 데이터베이스를 재구성해야 한다.

Meta Version

메타 버전은 변경되지 않았다.

패치를 롤백하려는 경우, [메타다운그레이드](#)를 참고한다.

CM protocol Version

통신 프로토콜 버전은 변경되지 않았다.

Replication protocol Version

Replication 프로토콜 버전은 변경되지 않았다..

프로퍼티

추가된 프로퍼티

변경된 프로퍼티

삭제된 프로퍼티

성능 뷰

추가된 성능 뷰

변경된 성능 뷰

삭제된 성능 뷰