

Altibase 7.1.0.7.0 Patch Notes

Table of Contents

- [New Features](#)
 - [BUG-49593 SQL 함수로 날짜시간함수 TIMESTAMPADD를 지원합니다.](#)
- [Fixed Bugs](#)
 - [BUG-49488 LOB 데이터 타입을 포함한 setBytes\(\)를 executeBatch\(\)로 처리하는 경우 java.lang.ClassCastException: Altibase.jdbc.driver.datatype.ListBufferHandle 에러가 발생합니다.](#)
 - [BUG-49545 AIX에서 Altibase 7.1.0.6.0 이상 버전을 사용하는 경우 쿼리 옵티마이저 Fault Tolerance 기능 수행 후 Altibase 서버가 비정상 종료할 수 있습니다.](#)
 - [BUG-49557 이중화 객체에 DDL 수행 시 The table structure has been modified. 에러가 반복되는 경우 altibase gp.log에 로그가 과도하게 출력되는 현상을 개선합니다.](#)
 - [BUG-49611 WITH 절과 조인하는 SQL 수행 시 메모리 참조 오류로 Altibase 서버가 비정상 종료할 수 있습니다.](#)
 - [BUG-49618 병렬 질의에 집계 함수를 포함한 경우 동시성 문제로 Altibase 서버가 비정상 종료할 수 있습니다.](#)
- [Changes](#)
 - [Version Info](#)
 - [호환성](#)
 - [프로퍼티](#)
 - [성능 분](#)

New Features

BUG-49593 SQL 함수로 날짜시간함수 TIMESTAMPADD를 지원합니다.

- **module** : mt-function
- **Category** : Functionality
- **재현 빈도** : Always
- **설명** : SQL 함수로 날짜시간함수 TIMESTAMPADD를 지원합니다.

TIMESTAMPADD 함수에 관한 보다 자세한 설명은 [Altibase 7.1 SQL Reference](#) 매뉴얼에서 확인할 수 있습니다.

TIMESTAMPADD 함수 추가로 TIMESTAMPADD 이름을 가진 객체(테이블, 인덱스, 저장 프로시저, 저장 함수 등) 생성이 불가능합니다. 또한, Altibase 7.1.0.7.0 이전 버전에서 TIMESTAMPADD 이름을 가진 객체를 생성한 경우, Altibase 7.1.0.7.0 부터는 사용할 수 없습니다.

Altibase 7.1.0.7.0 이상에서 TIMESTAMPADD 이름을 가진 객체를 생성하거나 사용하는 경우 ERR-31001 : SQL syntax error 에러가 발생합니다.

- **재현 방법**
 - 재현 절차

```
SELECT TIMESTAMPADD( SQL_TSI_DAY,1, TO_DATE('2003-01-02', 'YYYY-MM-DD')) FROM DUAL;
```

- 수행 결과

```
ERROR at line 1:ORA-00904: "TIMESTAMPADD": invalid identifier
```

- 예상 결과

```
'2003-01-03'
```

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

Fixed Bugs

BUG-49488 LOB 데이터 타입을 포함한 `setBytes()`를 `executeBatch()`로 처리하는 경우 `java.lang.ClassCastException: Altibase.jdbc.driver.datatype.ListBufferHandle` 에러가 발생합니다.

- **module** : mm-jdbc
- **Category** : Functional Error
- **재현 빈도** : Always
- **설명** : BLOB 타입 컬럼을 대상으로 `PreparedStatement.setBytes()`를 `executeBatch()`로 실행 시 `java.lang.ClassCastException: Altibase.jdbc.driver.datatype.ListBufferHandle` 에러가 발생합니다. 이 현상은 BLOB 데이터가 이진 타입이 처리할 수 있는 최대 크기인 65,534 바이트를 초과할 때 발생하는 것으로, 이진 타입 대신 BLOB 타입으로 처리하도록 변경합니다.

변경 동작을 원치 않을 경우 기존 동작을 유지할 수 있도록 JDBC 드라이버 연결 속성을 추가합니다.

- **JDBC 드라이버 연결 속성**

`batch_setbytes_use_lob`

- **설명**

BLOB 타입 컬럼을 대상으로 `PreparedStatement.setBytes()`를 `executeBatch()`로 실행 시 이진 타입과 BLOB 타입 중 어느 것으로 처리할지 설정한다. `true`는 BLOB 타입으로 `false`는 이진 타입으로 처리한다.

`batch_setbytes_use_lob`의 기본값은 `true`로, 기존 동작을 유지하고 싶으면 `batch_setbytes_use_lob=false`를 추가하고 재컴파일 해야 합니다. 관련 설명은 [JDBC User's Manual](#) 에서도 확인할 수 있습니다.

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

BUG-49545 AIX에서 Altibase 7.1.0.6.0 ~ 7.1.0.6.7 버전을 사용하는 경우 쿼리 옵티마이저 Fault Tolerance 기능 수행 후 Altibase 서버가 비정상 종료할 수 있습니다.

- **module** : id
- **Category** : Fatal

- **설명** : AIX에서 Altibase 7.1.0.6.0 ~ 7.1.0.6.7 버전을 사용하는 경우 쿼리 옵티마이저 Fault Tolerance 기능 수행 후 Altibase 서버가 비정상 종료하는 현상을 수정합니다.
- **재현 방법**
 - 재현 절차
 - 수행 결과
 - 예상 결과
- **Workaround**
- **변경사항**
 - Performance view
 - Property
 - Compile Option
 - Error Code

BUG-49557 이중화 객체에 DDL 수행 시 The table structure has been modified. 에러가 반복되는 경우 altibase_qp.log에 로그가 과도하게 출력되는 현상을 개선합니다.

- **module** : rp-control
- **Category** : Enhancement
- **재현 빈도** : Always
- **설명** : 이중화 객체에 DDL 수행 시 The table structure has been modified. 에러가 반복되는 경우 altibase_qp.log에 로그가 과도하게 기록되는 현상을 개선합니다.

The table structure has been modified. 에러는 보통 재시도하면 해결되는 종류의 에러로, 실패 시 재시도를 무한 반복하였으나 10회 재시도 후 ERR-61069 : Internal server error in replication module (The number of DDL rebuild attempts

RPU_REPLICATION_DDL_REBUILD_ERROR_MAX_COUNTT was exceeded.).에러가 발생시키고 실패 처리하도록 변경하였습니다.

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

BUG-49611 WITH 절과 조인하는 SQL 수행 시 메모리 참조 오류로 Altibase 서버가 비정상 종료할 수 있습니다.

- **module** : qp
- **Category** : Fatal
- **재현 빈도** : Rare
- **설명** : WITH 절과 조인하는 SQL 수행 시 메모리 참조 오류로 Altibase 서버가 비정상 종료하는 현상을 수정합니다.
- **재현 방법**
 - **재현 절차**

```
DROP TABLE COMM_SN_SUBINFO;
CREATE TABLE COMM_SN_SUBINFO(
    SVCNB VARCHAR(6) FIXED ,
    SUBNB VARCHAR(20) FIXED ,
    SUBKIND INTEGER FIXED,
    COM_NAME VARCHAR(50) VARIABLE,
    OFC_CD VARCHAR(10) FIXED,
    OFC_NAME VARCHAR(50) VARIABLE,
    AGNC_NAME VARCHAR(50) VARIABLE,
    END_DATE DATE FIXED,
    BEGIN_DATE DATE VARIABLE,
    CUST_NO VARCHAR(22) VARIABLE,
    SVC_OPT1 VARCHAR(16) VARIABLE,
    SVC_OPT2 VARCHAR(16) VARIABLE,
    PARANB VARCHAR(20) VARIABLE,
    SVCSTATUS INTEGER VARIABLE,
    VXMURLPSTN VARCHAR(128) VARIABLE,
    VXMLURLMOBILE VARCHAR(128) VARIABLE,
    CHAKIND VARCHAR(10) VARIABLE,
    COM_MODE VARCHAR(6) VARIABLE,
    CHANB VARCHAR(20) VARIABLE,
    STORE_OWNER_NO VARCHAR(20) VARIABLE,
    STORE_NAME VARCHAR(50) VARIABLE );
DROP INDEX PK_COMM_SN_SUBINFO;
CREATE UNIQUE INDEX PK_COMM_SN_SUBINFO ON COMM_SN_SUBINFO( SUBNB );
INSERT INTO COMM_SN_SUBINFO ( SUBNB ) SELECT LEVEL FROM DUAL CONNECT BY LEVEL <
10;
DROP TABLE COMM_FP_SUBINFO;
CREATE TABLE COMM_FP_SUBINFO(
    SVCNB VARCHAR(6) FIXED ,
    SUBNB VARCHAR(20) FIXED ,
    SUBKIND INTEGER FIXED,
    COM_NAME VARCHAR(50) VARIABLE,
    OFC_CD VARCHAR(10) FIXED,
    OFC_NAME VARCHAR(50) VARIABLE,
    AGNC_NAME VARCHAR(50) VARIABLE,
```

```

SVC_OPT1 VARCHAR(16) VARIABLE,
SVC_OPT2 VARCHAR(16) VARIABLE,
PARANB VARCHAR(20) VARIABLE,
SVCSTATUS INTEGER VARIABLE,
VXMURLPSTN VARCHAR(128) VARIABLE,
VXMLURLMOBILE VARCHAR(128) VARIABLE,
CHAKIND VARCHAR(10) VARIABLE,
COM_MODE VARCHAR(6) VARIABLE,
CHANB VARCHAR(20) VARIABLE,
STORE_OWNER_NO VARCHAR(20) VARIABLE,
STORE_NAME VARCHAR(50) VARIABLE );

DROP INDEX PK_COMM_FP_SUBINFO;
CREATE UNIQUE INDEX PK_COMM_FP_SUBINFO ON COMM_FP_SUBINFO( SUBNB );
INSERT INTO COMM_FP_SUBINFO ( SUBNB ) SELECT LEVEL FROM DUAL CONNECT BY LEVEL <
10;
var v1 varchar(10);
exec :v1:= '1';
prepare WITH SEARCH_NB_TBL AS (
    SELECT
        TO_CHAR(regexp_substr(CAST(:v1 AS VARCHAR(10000)), '[^,]+', 1, level)) AS
SEARCH_NB
    FROM DUAL
    CONNECT BY regexp_substr(:v1, '[^,]+', 1, LEVEL) IS NOT NULL
)
SELECT SUBNB
    , CUST_NO
    FROM COMM_SN_SUBINFO A
    , SEARCH_NB_TBL ST
WHERE A.SUBNB = ST.SEARCH_NB
UNION
SELECT SUBNB
    , CUST_NO
    FROM COMM_FP_SUBINFO B
    , SEARCH_NB_TBL ST
WHERE B.SUBNB = ST.SEARCH_NB;

```

- 수행 결과
- 예상 결과

- **Workaround**

- **변경사항**

- Performance view
- Property
- Compile Option
- Error Code

BUG-49618 병렬 질의에 집계 함수를 포함한 경우 동시성 문제로 Altibase 서버가 비정상 종료할 수 있습니다.

- **module** : qp-select-execute
- **Category** : Fatal
- **재현 빈도** : Rare
- **설명** : 병렬 질의에 집계 함수를 포함한 경우 동시성 문제로 Altibase 서버가 비정상 종료할 수 있습니다.
- **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
7.1.0.7.0	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.

Meta Version

메타 버전은 변경되지 않았다.

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