

# Altibase 7.1.0.7.3 Patch Notes

---

# Table of Contents

---

- [New Features](#)
  - [BUG-49580 메모리 테이블 또는 휘발성 메모리 테이블에 LOB 데이터 입력 시 과도한 메모리 사용을 개선합니다.](#)
  - [BUG-49668 동시에 수행할 수 있는 디스크 트랜잭션의 최대 수\(TRANSACTION SEGMENT COUNT 프로퍼티의 최대값\)를 512에서 16384로 증가합니다.](#)
- [Fixed Bugs](#)
  - [BUG-48065 DROP SEQUENCE 구문과 해당 시퀀스에 SELECT 구문을 동시에 수행하는 경우 Altibase 서버가 비정상 종료할 수 있습니다.](#)
  - [BUG-49505 복잡도가 높은 SQL문 수행 시 PREPARE STMT MEMORY MAXIMUM 초과로 SQL Plan Cache에 실행 계획을 등록하지 못한 경우 예외 처리를 추가합니다.](#)
  - [BUG-49569 UNION ALL 연산자를 사용한 SELECT 문에서 select list 절에 LOB, GEOMETRY 또는 이진 데이터 타입이 사용된 경우 예외 처리를 추가합니다.](#)
  - [BUG-49649 LOB 컬럼을 포함한 복제 트랜잭션 수행 시 Unable to find record in ~ 에러가 발생한 경우 수신자\(Receiver\)가 중지됩니다.](#)
  - [BUG-49654 문자열을 포함한 CHECK 제약조건을 가진 테이블을 대상으로 이중화 객체 생성 후 이중화를 시작하면 ERR-31001 SQL syntax error 에러가 발생합니다.](#)
  - [BUG-49661 디스크 임시 테이블스페이스를 사용하는 SQL문 수행 시 특정 상황에서 ERR-11069: Internal server error in the storage manager \(\[FAILURE\] ERR-110C1\(error=11\) The data file containing page \[129832016\] does not exist. 에러가 발생합니다.](#)
- [Changes](#)
  - [Version Info](#)
  - [호환성](#)
    - [Database binary version](#)
    - [Meta Version](#)
    - [CM protocol Version](#)
    - [Replication protocol Version](#)
  - [프로퍼티](#)
    - [추가된 프로퍼티](#)
    - [변경된 프로퍼티](#)
    - [삭제된 프로퍼티](#)
  - [성능 분](#)
    - [추가된 성능 분](#)
    - [변경된 성능 분](#)
    - [삭제된 성능 분](#)

# New Features

## BUG-49580 메모리 테이블 또는 휘발성 메모리 테이블에 LOB 데이터 입력 시 과도한 메모리 사용을 개선합니다.

- **module** : sm-mem-resource
- **Category** : Other
- **재현 빈도** : Always
- **설명** : 메모리 테이블 또는 휘발성 메모리 테이블에 LOB 데이터 입력 시 LOB 정보를 관리하는 런타임 객체에서 메모리를 과도하게 사용하는 현상을 개선합니다.
  - 이 버그 현상 예시
 

4G 크기의 LOB 데이터 입력 시 메모리 196G를 사용합니다. INSERT 진행 중 메모리를 과도하게 사용하다가 트랜잭션 종료 후 반납합니다.
- **재현 방법**
  - 재현 절차
  - 수행 결과
  - 예상 결과
- **Workaround**
- **변경사항**
  - Performance view
  - Property
  - Compile Option
  - Error Code

## BUG-49668 동시에 수행할 수 있는 디스크 트랜잭션의 최대 수 (TRANSACTION\_SEGMENT\_COUNT 프로퍼티의 최대값)를 512에서 16384로 증가합니다.

- **module** : sm\_recovery
- **Category** : Usability
- **재현 빈도** : Always
- **설명** : 디스크 테이블 트랜잭션의 성능 향상을 위해 동시에 수행할 수 있는 디스크 트랜잭션의 최대 수를 512에서 16384로 증가합니다.

관련 프로퍼티인 TRANSACTION\_SEGMENT\_COUNT의 최댓값과 속성이 변경되었습니다.

- 최댓값이 512에서 16384로 변경되었습니다.
- 프로퍼티의 속성이 '변경 가능'에서 '읽기 전용'으로 변경되었습니다.

TRANSACTION\_SEGMENT\_COUNT 프로퍼티의 설정값이 900을 초과할 경우 언두 테이블스페이스의 세그먼트 헤더 정보를 저장한 별도의 파일(txSegEntry.hdr)이 생성됩니다. 이 파일에 위치는 언두 테이블스페이스의 데이터 파일과 같은 곳에 있으며 변경할 수 없습니다. txSegEntry.hdr에 관한 보다 자세한 내용은 매뉴얼을 참

- [Administrator's Manual - 10. 백업 및 복구 - 데이터베이스 백업 - 범위에 따른 백업 분류](#)
- [Administrator's Manual - 10. 백업 및 복구 - 데이터베이스 백업 - 방식에 따른 백업 분류](#)

- **재현 방법**

- 재현 절차
- 수행 결과
- 예상 결과

- **Workaround**

- **변경사항**

- Performance view
- Property
  - TRANSACTION\_SEGMENT\_COUNT : 최대값이 900에서 16384로 증가
- Compile Option
- Error Code

# Fixed Bugs

**BUG-48065 DROP SEQUENCE 구문과 해당 시퀀스에 SELECT 구문을 동시에 수행하는 경우 Altibase 서버가 비정상 종료할 수 있습니다.**

- **module** : sm
- **Category** : Fatal
- **재현 빈도** : Always
- **설명** : DROP SEQUENCE 구문과 해당 시퀀스에 SELECT 구문을 동시에 수행하는 경우 Altibase 서버가 비정상 종료하는 현상을 수정합니다.

아래 조건 중 하나라도 해당하면 이 버그에 노출되어 있습니다.

- Altibase 7.1.0.1.2 ~ 7.1.0.7.0 버전
- altibase.properties에 `__CATALOG_SLOT_REUSABLE = 1`을 추가한 경우
- **재현 방법**
  - **재현 절차**

```
CREATE SEQUENCE S1;
-- SESSION 1
declare
  cursor c1 is select s1.nextval from dual connect by level < 5;
  var1 integer;
begin
  open c1;
  loop
    fetch c1 into var1;
    exit when c1%notfound;
    println(var1);
    sleep(10);
  end loop;
end;
/
-- SESSION 2
DROP SEQUENCE S1;
```

- **수행 결과**

```
1234
Execute success.
```

- **예상 결과**

```
1
[ERR-1107C : The sequence has already been dropped.at "SYS.ANONYMOUS_BLOCK", line
7]
```

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

## BUG-49505 복잡도가 높은 SQL문 수행 시 PREPARE\_STMT\_MEMORY\_MAXIMUM 초과로 SQL Plan Cache에 실행 계획을 등록하지 못한 경우 예외 처리를 추가합니다.

- **module** : qp
- **Category** : Fatal
- **재현 빈도** : Always
- **설명** : 복잡도가 높은 SQL문 수행 시 PREPARE\_STMT\_MEMORY\_MAXIMUM 초과로 SQL Plan Cache에 실행 계획을 등록하지 못한 경우 Altibase 서버가 비정상 종료하는 현상을 수정합니다.
- **재현 방법**
  - 재현 절차

```
DROP TABLE BUG-49505;

CREATE TABLE BUG-49505
(
    I1  INTEGER,
    I2  INTEGER,
    I3  INTEGER,
    I4  INTEGER,
    I5  INTEGER,
    I6  INTEGER,
    I7  INTEGER,
    I8  INTEGER,
    I9  INTEGER,
    I10 INTEGER,
    I11 INTEGER,
    I12 GEOMETRY
)
PARTITION BY RANGE( I1 )
(
    PARTITION P1 VALUES LESS THAN ( 10 ),
    PARTITION P2 VALUES LESS THAN ( 20 ),
    PARTITION P3 VALUES LESS THAN ( 30 ),
    PARTITION P4 VALUES DEFAULT
);

ALTER TABLE BUG-49505 MERGE PARTITIONS P2, P3 INTO PARTITION P3 ;
ALTER TABLE BUG-49505 DROP PARTITION P1;
```

[illegible]

```
SELECT * FROM BUG-49505 UNION ALL SELECT * FROM BUG-49505 UNION ALL SELECT * FROM  
BUG-49505 UNION ALL SELECT * FROM BUG-49505 UNION ALL SELECT * FROM BUG-49505 );
```

#### ◦ 수행 결과

Altibase 서버 비정상 종료

#### ◦ 예상 결과

동일 쿼리 첫 번째 수행 시

```
ERR-01067 : The memory size allocated for the statement has exceeded the maximum  
limit ( Name : Query_Prepere, Wanted Memory Size : 209719025, Max size :  
209715200 ).
```

동일 쿼리 재 수행 시

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

#### • Workaround

##### • 변경사항

- Performance view
- Property
- Compile Option
- Error Code

**BUG-49569 UNION ALL 연산자를 사용한 SELECT 문에서 select\_list 절에 LOB, GEOMETRY 또는 이진 데이터 타입이 사용된 경우 예외 처리를 추가합니다.**

- **module** : qp
- **Category** : Fatal
- **재현 빈도** : Always
- **설명** : UNION ALL 연산자를 사용한 SELECT 문에서 select\_list 절에 LOB, GEOMETRY, NIBBLE, BYTE, VARBYTE 데이터 타입 중 2가지를 같이 사용한 경우 Altibase 서버가 비정상 종료하거나 무한 대기하는 상황에 예외 처리를 추가하였습니다.

이 버그가 반영된 버전에서는 ERR-91022 : LOB and GEOMETRY type data cannot be displayed 또는 ERR-21043 : Unexpected errors may have occurred: mtnCalculate\_Blob2BlobLocator: not BLOB module 에러가 발생합니다.

#### • 제형 바버



```

DROP TABLE t1 CASCADE;
DROP TABLE t2 CASCADE;
CREATE TABLE t1( I1 BLOB );
CREATE TABLE t2 ( I1 BYTE(12) );
INSERT INTO t2 VALUES (1);
SELECT i1 FROM t1 UNION ALL SELECT i1 FROM t2;

```

#### ◦ 수행 결과

```
[ERR-91015 : Communication failure.]
```

#### ◦ 예상 결과

```
[ERR-91022 : LOB and GEOMETRY type data cannot be displayed]
```

### • Workaround

#### • 변경사항

- Performance view
- Property
- Compile Option
- Error Code

## BUG-49649 LOB 컬럼을 포함한 복제 트랜잭션 수행 시 Unable to find record in ~ 에러가 발생한 경우 수신자(Receiver)가 중지됩니다.

- **module** : rp-receiver
- **Category** : Functional Error
- **재현 빈도** : Always
- **설명** : LOB 컬럼을 포함한 복제 트랜잭션 수행 시 Unable to find record in ~ 에러가 발생한 경우 수신자(Receiver)가 중지되지 않고 정상적으로 데이터 충돌 처리하도록 수정합니다.

이 버그 발생 시 수신자, 송신자에서 확인할 수 있는 현상입니다.

#### ◦ 수신자 측

트레이스 로그에 아래와 같은 에러가 남습니다.

##### ■ altibase\_rp\_conflict.log

```
ERR-610f7(errno=62) [Receiver] Unable to find record in ...
```

##### ■ altibase\_rp.log

```
ERR-610f7(errno=62) [Receiver] Unable to find record in openLOBCursor()
function
ERR-61048(errno=62) [Receiver] replication_name receiver has recvXLog error
in run()
[Receiver] RepName:replication_name is processed at ~
[Receiver] RepName:replication_name is received at ~
ERR-6104b(errno=62) [Receiver] replication_name receiver is ended(by
thr_exit)
Error Stop!
[Receiver] Replication replication_name Stopped ...
```

#### ○ 송신자 측

송신자 측에서 REPLICATION\_SENDER\_SLEEP\_TIMEOUT 프로퍼티 설정에 따라 재시작을 반복하지만 수신자 측에서 Unable to find record in openLOBCursor() function 에러로 수신자가 정상적으로 시작되지 않습니다. 이로 인해 이중화 갭이 발생합니다.

트레이스 로그에 아래와 같은 에러가 남습니다.

#### ■ altibase\_rp.log

```
[SenderApply] Replication replication_name Start
[Sender] Replication replication_name:0 Start... at[93025988]
[SenderApply] Replication replication_name Got Error
ERR-620f0(errno=11) [Sender] Stop sender thread replication_name:0 at
[93027296], Restart SN[93025988]
ERR-7101a(errno=16) Connection closed
ERR-61022(errno=11) [Sender] Sender Sleep : 60 seconds
```

이 버그가 반영된 버전에서는 LOB 컬럼을 포함한 복제 트랜잭션 수행 시 존재하지 않는 프라이머리 키 값인 경우 수신자 측 트레이스 로그 altibase\_rp\_conflict.log에 아래와 같은 형태의 로그가 남습니다.

```
ERR-610f7(errno=16) [Receiver] Unable to find record in openLOBCursor() function
A failure occurred while opening LOB Cursor (Table : ) (PK : ) ; (TID : ) COMMIT (TID : )
```

#### ● 재현 방법

- 재현 절차
- 수행 결과
- 예상 결과

#### ● Workaround

#### ● 변경사항

- Performance view
- Property
- Compile Option
- Error Code

## BUG-49654 문자열을 포함한 CHECK 제약조건을 가진 테이블을 대상으로 이중화 객체 생성 후 이중화를 시작하면 ERR-31001 SQL syntax error 에러가 발생합니다.

- **module** : rp-sender
- **Category** : Functional Error
- **재현 빈도** : Always
- **설명** : 문자열을 포함한 CHECK 제약조건을 가진 테이블을 대상으로 이중화 객체 생성 후 이중화를 시작하면 ERR-31001 SQL syntax error 에러가 발생하는 현상을 수정합니다.

이 버그로 인한 ERR-31001 SQL syntax error 에러는 아래 3가지 상황에서 발생할 수 있습니다.

- 문자열을 포함한 CHECK 제약조건을 가진 테이블을 대상으로 이중화 객체 생성 후 이중화를 시작하는 경우

ALTER REPLICATION *replication\_name* START 수행 시 ERR-6104F : [Sender] Failed to initialize 에러 발생하며 altibase\_rp.log에 ERR-31001(errno=11) SQL syntax error 발생

- 문자열을 포함한 CHECK 제약조건을 가진 테이블을 이중화 객체에 추가하는 경우

ALTER REPLICATION *replication\_name* ADD TABLE 수행 시 ERR-31001 : SQL syntax error 에러 발생

- 이중화 대상 테이블에 문자열을 포함한 CHECK 제약조건을 추가하는 경우

ALTER TABLE *table\_name* ADD CONSTRAINT *constraint\_name* CHECK 수행 시 altibase\_rp.log에 ERR-31001(errno=62) SQL syntax error가 발생하며 이중화 송신자가 중지

이 상황이 발생하면 이중화 객체 삭제 후 재생성해야 합니다.

- **재현 방법**
  - **재현 절차**

```
CREATE TABLE RCSC_AUID
(
    TS TIMESTAMP,
    AUID VARCHAR(96) not null,
    AUID_USE SMALLINT default 1,
    XDMS VARCHAR(8),
    XDMS_USE SMALLINT default 1,
    PRIMARY KEY (AUID)
);

ALTER TABLE RCSC_AUID
ADD CONSTRAINT RCSC_RCSC_AUID_CHECK_01
CHECK( XDMS in ('GMS', 'CMS', 'IdMS', 'KMS', 'PROXY') );

ALTER REPLICATION REP1 START;
```

- **수행 결과**

ERR-6104F : [Sender] Failed to initialize 에러 발생하며 altibase\_rp.log에 ERR-

- **Workaround**

없음

- **변경사항**

- Performance view
- Property
- Compile Option
- Error Code

**BUG-49661 디스크 임시 테이블스페이스를 사용하는 SQL문 수행 시 특정 상황에서 ERR-11069 : Internal server error in the storage manager ([FAILURE] ERR-110C1(error=11) The data file containing page [129832016] does not exist. 에러가 발생합니다.**

- **module** : sm
- **Category** : Functional Error
- **재현 빈도** : Always
- **설명** : 디스크 임시 테이블스페이스를 사용하는 SQL문 수행 시 특정 상황에서 ERR-11069 : Internal server error in the storage manager ([FAILURE] ERR-110C1(error=11) The data file containing page [129832016] does not exist. 에러가 발생하는 현상을 수정합니다.

이 버그는 아래 조건을 모두 만족하는 경우 매우 낮은 확률로 발생합니다.

- Altibase 7.1.0.5.7 이상
- 디스크 템프 테이블 연산 중 HASH 연산 수행
- HASH\_AREA\_SIZE 부족으로 디스크 임시 테이블스페이스를 사용하는 경우

참고 : 디스크 테이블에 대한 질의 처리 과정에서 SORT/HASH 연산이 필요할 경우 빠른 연산을 위해 메모리에 일정 크기를 할당하여 사용한다. 만약 정해진 크기의 메모리를 모두 사용하고 SORT/HASH 연산을 위한 공간이 추가적으로 필요한 경우 디스크 임시 테이블스페이스를 사용한다.

- **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
7.1.0.7.3	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 부

삭제된 성능 부