

Altibase 7.3.0.0.3 Patch Notes

Table of Contents

- [New Features](#)
 - [BUG-50654 로그 파일 준비 스레드\(Log File Prepare Thread\)의 병목현상 개선](#)
 - [BUG-50682 Windows client에서 OpenSSL 지원](#)
- [Fixed Bugs](#)
 - [BUG-50527 NVL EQUAL\(\), NVL NOT EQUAL\(\) 함수의 인자로 인덱스 컬럼에 대한 연산식을 적용할 경우 서버가 비정상 종료하는 경우가 있습니다.](#)
 - [BUG-50542 하이브리드 파티션드 테이블이면서 GEOMETRY 컬럼 또는 LOB 컬럼이 포함되고 update trigger로 설정 된 경우, multiple update 구문 수행 시 비정상 종료 발생 합니다.](#)
 - [BUG-50621 함수 기반 인덱스를 생성하는 수식이 지나치게 긴 경우, 서버 비정상 종료가 발생 합니다.](#)
 - [BUG-50659 ORDER BY 가 SELECT 절의 외부 참조가 있는 SUBQUERY를 alias로 참조 시 결과 오류가 발생합니다.](#)
 - [BUG-50663 LobCursorList 관리 함수의 동시성 문제를 수정합니다.](#)
 - [BUG-50670 OpenSSL을 이용하는 환경에서 오류 메시지 출력 부분에 발견된 오류를 수정합니다.](#)
 - [BUG-50686 V\\$TIME_ZONE NAMES에서 America/Porto Velho 타임존의 UTC OFFSET 값이 올바르지 않습니다.](#)
 - [BUG-50697 JDBC 에서 PreparedStatement를 이용하여 ping 쿼리 사용시 메모리 누수가 발생합니다.](#)
 - [BUG-50700 하이브리드 파티션드 테이블에서 컬럼 제약을 체크하는 로직에서 잘못된 row offset 정보로 인해 잘못된\(invalid\) 메모리 접근의 오류가 발생할 수 있습니다.](#)
- [Changes](#)
 - [Version Info](#)
 - [호환성](#)
 - [프로퍼티](#)
 - [성능 분](#)

New Features

BUG-50654 로그 파일 준비 쓰레드(Log File Prepare Thread)의 병목현상 개선

- **module** : sm
- **Category** : Enhancement
- **재현 빈도** : Always
- **설명** : 로그 파일 준비 쓰레드(Log File Prepare Thread)의 병목현상을 개선한 버그로, LOG_FILE_SIZE 의 기본값이 10MB에서 100MB로 변경되었습니다. 이로 인해 7.3.0.0.2에서 LOG_FILE_SIZE를 10M으로 설정한 경우, 7.3.0.0.3으로 패치시 DB를 재구성해야 합니다.
- **재현 방법**
- **재현 절차**
- **수행 결과**
- **예상 결과**
- **Workaround**
- **변경사항**
 - Performance view
 - Property
 - Compile Option
 - Error Code

BUG-50682 Windows 용 클라이언트에서 OpenSSL 지원

- **module** : mm
- **Category** : Portability
- **재현 빈도** : Always
- **설명** : Windows 용 클라이언트에서 OpenSSL을 사용할 수 있도록 기능이 개선되었습니다. 지원하는 OpenSSL의 버전은 3.1.3 이상입니다.
- **재현 방법**
 - **재현 절차**
 - **수행 결과**
 - **예상 결과**
- **Workaround**
- **변경사항**
 - Performance view
 - Property
 - Compile Option
 - Error Code

Fixed Bugs

BUG-50527 NVL_EQUAL(), NVL_NOT_EQUAL() 함수의 인자로 인덱스 컬럼에 대한 연산식을 적용할 경우 서버가 비정상 종료하는 경우가 있습니다.

- **module** : mt
- **Category** : Fatal
- **재현 빈도** : Always
- **설명** : NVL_EQUAL(), NVL_NOT_EQUAL() 함수의 인자로 인덱스 컬럼에 대한 연산식을 적용하는 경우, 서버가 비정상 종료하는 문제를 수정하였습니다.
- **재현 방법**
 - 재현 절차
 - 수행 결과
 - 예상 결과
- **Workaround**
- **변경사항**
 - Performance view
 - Property
 - Compile Option
 - Error Code

BUG-50542 하이브리드 파티션드 테이블이면서 GEOMETRY 컬럼 또는 LOB 컬럼이 포함되고 update trigger로 설정 된 경우, multiple update 구문 수행 시 비정상 종료 발생 합니다.

- **module** : sm
- **Category** : Fatal
- **재현 빈도** : Always
- **설명** : 아래의 경우를 모두 만족하는 특정한 상황에서 multiple update 구문 수행시 발생하는 비정상 종료 문제를 수정하였습니다.
 - 하이브리드 파티션드 테이블이면서 GEOMETRY 컬럼 또는 LOB 컬럼이 포함되어 있는 경우
 - update trigger event가 설정되어 있는 경우
- **재현 방법**
 - 재현 절차

```
DROP TABLE T1;
DROP TABLE T2;

CREATE TABLE T1 ( I1 INTEGER, I2 FLOAT, I3 GEOMETRY ) PARTITION BY
RANGE(I1) ( PARTITION P1 VALUES DEFAULT TABLESPACE SYS_TBS_disk_DATA )
TABLESPACE SYS_TBS_mem_DATA;
```

```
CREATE TABLE T2 (I1 INTEGER, I2 VARCHAR(1000) );

CREATE TRIGGER AFTER_UPDATE AFTER UPDATE ON T1
REFERENCING
OLD AS OLDROW
NEW AS NEWROW
FOR EACH ROW
AS
BEGIN
INSERT INTO T2 VALUES (7777, 'AFTER UPDATE');
END;
/

INSERT INTO T1(I2) VALUES (178);
UPDATE T1 LEFT OUTER JOIN T2 ON T1.I1 = T2.I1 SET T1.I1 = T1.I1 + 5,
T2.I1 = T2.I1 + 10;
```

- 수행 결과

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

- 예상 결과

```
[ERR-1105F : No more than one update cursor can be used on a table.
at "SYS.AFTER_UPDATE", line 8]
```

- Workaround

- 변경사항

- Performance view
- Property
- Compile Option
- Error Code

BUG-50621 함수 기반 인덱스를 생성하는 수식이 지나치게 긴 경우, 서버 비정상 종료가 발생합니다.

- module : qp-ddl-dcl-pvo
- Category : Fatal
- 재현 빈도 : Always
- 설명 : 함수 기반 인덱스를 생성하는 수식이 지나치게 긴 경우, 서버가 비정상 종료되는 문제를 수정합니다.
- 재현 방법
 - 재현 절차
 - 수행 결과
 - 예상 결과
- Workaround
- 변경사항

- Performance view
- Property
- Compile Option
- Error Code

BUG-50659 ORDER BY 가 SELECT 절의 외부 참조가 있는 SUBQUERY를 alias로 참조 시 결과 오류가 발생합니다.

- **module** : qp
- **Category** : Reliability
- **재현 빈도** : Always
- **설명** : ORDER BY 절에서 SELECT 절의 외부 참조가 있는 서브쿼리를 alias로 참조할 때 결과 오류가 발생하는 문제를 수정합니다.

이 버그는 아래의 4가지 조건이 모두 만족한 경우에만 발생합니다.

1. 디스크 테이블에서 ORDER BY시 디스크 템프를 사용한 경우
2. 서브쿼리의 alias를 ORDER BY에 사용한 경우
3. SELECT의 target에 서브쿼리가 2개 이상 사용되고, 그 중에 ORDER BY 절에 사용된 alias의 서브쿼리보다 앞에 있는 서브쿼리에서 외부참조 컬럼이 사용된 경우
4. 서브쿼리에 사용된 외부 참조 컬럼이 FROM 절의 인라인 뷰(Inline View)나 일반 뷰에 사용되면서, 뷰머지(View Merge)로 동작한 경우

- **재현 방법**
 - **재현 절차**

```
drop table tb1;
drop table tb2;
drop table tb3;
CREATE TABLE TB1 ( TB1_C1 VARCHAR(10) PRIMARY KEY,
                   TB1_C2 VARCHAR(10) NOT NULL,
                   TB1_C3 VARCHAR(10) NOT NULL )
tablespace sys_tbs_disk_data
;
insert into tb1 values ('aaa', 'aaa', 'aaa');
insert into tb1 values ('bbb', 'aaa', 'bbb');
insert into tb1 values ('ccc', 'aaa', 'ccc');
insert into tb1 values ('ddd', 'ccc', 'ddd');
insert into tb1 values ('eee', 'eee', 'eee');
CREATE TABLE TB2 ( TB2_C1 VARCHAR(10) PRIMARY KEY,
                   TB2_C2 VARCHAR(10) NOT NULL,
                   TB2_C3 VARCHAR(10) NOT NULL,
                   TB2_C4 NUMERIC(5,2),
                   TB2_C5 DATE DEFAULT SYSDATE )
tablespace sys_tbs_disk_data;
insert into tb2 values('aaa', 'aaa', 'aaa', 1, sysdate);
insert into tb2 values('bbb', 'aaa', 'bbb', 1, sysdate);
insert into tb2 values('ccc', 'bbb', 'ccc', 1, sysdate);
insert into tb2 values('eee', 'eee', 'eee', 1, sysdate);
insert into tb2 values('fff', 'eee', 'fff', 1, sysdate);
```

```
CREATE TABLE Tb3 ( TB3_C1 VARCHAR(10) PRIMARY KEY,
                   TB3_C2 VARCHAR(10) NOT NULL )
tablespace sys_tbs_disk_data;
insert into tb3 values ( 'aaa', 'aaa' );
insert into tb3 values ( 'bbb', 'bbb' );
insert into tb3 values ( 'ccc', 'ccc' );
insert into tb3 values ( 'fff', 'fff' );
SELECT
  (SELECT X.TB1_C3 FROM TB1 X WHERE X.TB1_C1 = A.TB2_C1)
,(SELECT X.TB1_C2 FROM TB1 X WHERE X.TB1_C1 = A.TB2_C1) AS AAAA
FROM TB2 A
WHERE EXISTS (SELECT 1 FROM TB3 X WHERE X.TB3_C2 = A.TB2_C1 )
ORDER BY AAAA ASC;
```

○ 수행 결과

```
aaa
ccc
bbb

4 rows selected.
```

○ 예상 결과

```
aaa      aaa
ccc      aaa
bbb      aaa

4 rows selected.
```

• Workaround

```
SELECT
  (SELECT X.TB1_C3 FROM TB1 X WHERE X.TB1_C1 = A.TB2_C1)
,(SELECT X.TB1_C2 FROM TB1 X WHERE X.TB1_C1 = A.TB2_C1) AS AAAA
FROM TB2 A
WHERE EXISTS (SELECT /*+ no_unnest */ 1 FROM TB3 X WHERE X.TB3_C2 = A.TB2_C1
)
ORDER BY AAAA ASC;
SELECT (SELECT X.TB1_C3 FROM TB1 X WHERE X.TB1_C1 =
A.TB2_C1),(SELECT X.TB1_C2 FROM TB1 X WHERE X.TB1_C1 = A.TB2_C1) AS AAAA
FROM TB2 A
WHERE EXISTS (SELECT /*+ no_unnest */ 1 FROM TB3 X WHERE X.TB3_C2 =
A.TB2_C1 )
ORDER BY AAAA ASC;
```

• 변경사항

- Performance view
- Property
- Compile Option
- Error Code

BUG-50663 LobCursorList 관리 함수의 동시성 문제를 수정합니다.

- **module** : sm
- **Category** : Fatal
- **재현 빈도** : Unknown
- **설명** : 내부적으로 LobCursorList를 관리하기 위해 사용되는 함수들의 동시성 문제가 발견되어, 동시성 제어를 위한 함수로 수정하였습니다.
- **재현 방법**
 - 재현 절차
 - 수행 결과
 - 예상 결과
- **Workaround**
- **변경사항**
 - Performance view
 - Property
 - Compile Option
 - Error Code

BUG-50670 OpenSSL을 이용하는 환경에서 오류 메시지 출력 부분에 발견된 오류를 수정합니다.

- **module** : cm
- **Category** : Functional Error
- **재현 빈도** : Always
- **설명** : OpenSSL을 활용한 프로그램에서 인증서 위치 경로가 잘못된 상황에 SQLDriverConnect를 연속으로 호출하는 경우, 잘못된 오류 메시지가 출력되는 문제를 수정합니다.
- **재현 방법**
 - 재현 절차
 - 수행 결과
 - 예상 결과
- **Workaround**
- **변경사항**
 - Performance view
 - Property
 - Compile Option
 - Error Code

BUG-50686 V\$TIME_ZONE_NAMES에서 America/Porto_Velho 타임존의 UTC_OFFSET 값이 올바르지 않습니다.

- **module** : mt
- **Category** : Functional Error
- **재현 빈도** : Always
- **설명** : 아래 TIMEZONE DATA를 수정합니다.
 - America/Porto_Velho -04:00
- **재현 방법**
 - 재현 절차

```
select * from v$time_zone_names where name='America/Porto_Velho';
```

- 수행 결과

NAME	UTC_OFFSET
America/Porto_Velho	04:00

- 예상 결과

NAME	UTC_OFFSET
America/Porto_Velho	-04:00

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

BUG-50697 JDBC 에서 PreparedStatement를 이용하여 ping 쿼리 사용시 메모리 누수가 발생합니다.

- **module** : mm-jdbc
- **Category** : Functional Error
- **재현 빈도** : Always
- **설명** : JDBC 에서 PreparedStatement를 이용하여 ping 쿼리 사용시 메모리 누수가 발생하는 문제를 수정하였습니다. 이 버그를 적용하려면 JDBC 드라이버를 패치해야 합니다.
- **재현 방법**
 - 재현 절차

```

Connection sConn = getConnection("20300");
PreparedStatement sStmt = sConn.prepareStatement("/ * PING */ SELECT 1
");
ResultSet sRs = sStmt.executeQuery();
if (sRs.next())
{
    sRs.close();
}
sStmt.executeQuery();
sStmt.executeQuery();
sStmt.executeQuery();
sStmt.close();
sConn.close();

```

- 수행 결과

메모리 사용량이 계속 증가함

- 예상 결과

메모리 사용량이 계속 증가하지 않음

- Workaround

ping 쿼리 대신 select 1 from dual 사용

- 변경사항

- Performance view
- Property
- Compile Option
- Error Code

BUG-50700 하이브리드 파티션드 테이블에서 컬럼 제약을 체크하는 로직에서 잘못된 row offset 정보로 인해 잘못된(invalid) 메모리 접근의 오류가 발생할 수 있습니다.

- module : qp
- Category : Memory Error
- 재현 빈도 : Always
- 설명 : 하이브리드 파티션드 테이블에서 컬럼 제약을 체크하는 로직에서 잘못된 row offset 정보로 인한 잘못된(invalid) 메모리 접근의 오류가 발생하지 않도록 수정합니다.
- 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
7.3.0.0.3	7.3.0	9.3.1	7.1.8	7.4.9

Altibase 7.3 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

