

Altibase 6.5.1.10.1 Patch Notes

- [New Features](#)
 - [BUG-50713 잘못된 패킷으로 Protocol header error가 발생한 경우, 에러메시지에 헤더 내용 출력하도록 개선합니다.](#)
- [Fixed Bugs](#)
 - [BUG-49356 인덱스 소유자가 테이블 소유자와 다른 경우, 전체 DB 모드로 aexport 수행 시 해당 인덱스 생성 스크립트를 추출하지 않습니다.](#)
 - [BUG-50686 V\\$TIME_ZONE_NAMES에서 America/Porto Velho 타임존의 UTC OFFSET 값이 올바르지 않습니다.](#)
 - [BUG-50694 디스크테이블에서 서브쿼리 내의 GROUP BY 칼럼이 메인 쿼리의 참조로 사용되는 경우 서버가 비정상 종료하는 문제를 수정합니다.](#)
 - [BUG-50710 External procedure를 호출하는 package를 생성하는 과정에서 서버가 비정상 종료할 수 있는 문제를 수정합니다.](#)
 - [BUG-50729 종료된 쓰레드의 스택 크기가 v\\$memstat 에서 계속 출력됩니다.](#)
 - [BUG-50758 APRE에서 double quote 문자에 대한 escape 처리 누락](#)
 - [BUG-50783 프로시저 실행 중 Table에 DDL 수행시, Invalid use of host variables error 가 발생하는 문제를 수정합니다.](#)
 - [BUG-50791 TO_CHAR 함수를 ALIAS로 조건절에 사용시 뷰에 대한 조건절 Pushdown 이 2번 이상 발생할 경우, 비정상 종료되는 문제를 수정합니다.](#)
 - [BUG-50844 ADO.net의 AltibaseBulkCopy 사용 후 메모리 사용량이 급증하는 문제를 수정합니다.](#)
- [Changes](#)
 - [Version Info](#)
 - [호환성](#)
 - [프로퍼티](#)
 - [성능 분](#)

New Features

BUG-50713 잘못된 패킷으로 Protocol header error가 발생한 경우, 에러메시지에 헤더 내용 출력하도록 개선합니다.

- **module** : cm
- **Category** : Functionality
- **재현 빈도** : Rare
- **설명** : 잘못된 패킷이 수신되어 Protocol header error가 발생했을 때, 에러메시지에 헤더 내용을 출력하여 문제 분석에 도움을 주도록 개선하였습니다.
 - 수정전 : ERR-7101d(errno=0) Protocol header error.(TCP 127.0.0.1:41462)
 - 수정후 : ERR-710cc(errno=0) Protocol header error.(TCP 127.0.0.1:41462, 0, 3132333435363738393031323334353637383930)
- **재현 방법**
- **재현 절차**
- **수행 결과**
- **예상 결과**
- **Workaround**
- **변경사항**
 - Performance view
 - Property
 - Compile Option
 - Error Code
 - 수정전 : ERR-7101d(errno=0) Protocol header error.(TCP 127.0.0.1:41462)
 - 수정후 : ERR-710cc(errno=0) Protocol header error.(TCP 127.0.0.1:41462, 0, 3132333435363738393031323334353637383930)

Fixed Bugs

BUG-49356 인덱스 소유자가 테이블 소유자와 다른 경우, 전체 DB 모드로 **aexport** 수행 시 해당 인덱스 생성 스크립트를 추출하지 않습니다.

- **module** : ux-aexport
- **Category** : Functional Error
- **재현 빈도** : Always
- **설명** : 인덱스 소유자가 테이블 소유자와 다른 경우, 전체 DB 모드로 aexport 수행 시 해당 인덱스 생성 스크립트를 추출하지 않는 문제를 수정하였습니다.
- **재현 방법**
 - **재현 절차**

```
CREATE USER alti IDENTIFIED BY test;
CONNECT alti/test
DROP TABLE t1;
CREATE TABLE t1 (c1 INTEGER, c2 CHAR(10), c3 CHAR(10));
CREATE INDEX t1_idx1 ON t1 (c1);

CONNECT sys/manager
CREATE INDEX alti.t1_idx2 ON alti.t1 (c2);
CREATE INDEX t1_idx3 ON alti.t1 (c3);
```

- **수행 결과**

T1_IDX3 인덱스 생성 스크립트는 추출되지 않는 문제가 있습니다.

```
$ aexport -s 127.0.0.1 -u sys -p manager
$ grep T1_IDX3 *
$
```

- **예상 결과**

ALL_CRT_INDEX.sql 파일에 T1_IDX3 인덱스 생성 스크립트를 확인할 수 있습니다.

```
$ grep T1_IDX3 *
ALL_CRT_INDEX.sql:create index "T1_IDX3" on "ALTI"."T1"("C3") tablespace
"SYS_TBS_MEM_DATA";
```

- **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** : V\$TIME_ZONE_NAMES에서 America/Porto_Velho 타임존의 UTC_OFFSET 값이 잘못된 값인 "04:00"으로 표시되고 있어서, 올바른 값인 "-04:00" 로 수정합니다.
- **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-50694 디스크테이블에서 서브쿼리 내의 GROUP BY 칼럼이 메인 쿼리의 참조로 사용되는 경우 서버가 비정상 종료하는 문제를 수정합니다.

- **module** : qp
- **Category** : Fatal
- **재현 빈도** : Always
- **설명** : 이 버그는 디스크 테이블에서만 재현됩니다. 서브쿼리 내의 GROUP BY 칼럼이 메인 쿼리의 참조로 사용되는 경우 서버가 비정상 종료하는 문제를 수정합니다.
- **재현 방법**
 - **재현 절차**

```
CREATE TABLE T1( I1 INTEGER, I2 INTEGER, I3 INTEGER ) TABLESPACE
SYS_TBS_DISK_DATA;
INSERT INTO T1 VALUES (2, NULL, NULL);
SELECT * FROM T1 WHERE I1 > ALL ( SELECT MIN(I2) FROM DUAL WHERE I1 = '1'
GROUP BY I1);
```

- 수행 결과

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

- 예상 결과

```
I1          I2          I3
-----
2
1 row selected.
```

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

BUG-50710 External procedure를 호출하는 package를 생성하는 과정에서 서버가 비정상 종료할 수 있는 문제를 수정합니다.

- **module** : qp-psm-trigger-pvo
- **Category** : Functional Error
- **재현 빈도** : Always
- **설명** : External procedure를 호출하는 package를 생성하는 과정에서 서버가 비정상 종료할 수 있는 문제를 수정합니다.
- **재현 방법**
 - 재현 절차
 - 수행 결과
 - 예상 결과
- Workaround
- 변경사항
 - Performance view
 - Property
 - Compile Option
 - Error Code

BUG-50729 종료된 쓰레드의 스택 크기가 v\$memstat 에서 계속 출력됩니다.

- **module** : id
- **Category** : Functional Error
- **재현 빈도** : Always
- **설명** : 쓰레드 종료 시 스택이 해제되어도 V\$MEMSTAT 조회에 반영되지 않는 문제를 수정합니다. 이로 인해, 실제 메모리 사용량보다 더 많은 사용량으로 표시되는 문제가 해결되었습니다.
- **재현 방법**
 - 재현 절차
 - 수행 결과
 - 예상 결과
- **Workaround**
- **변경사항**
 - Performance view
 - Property
 - Compile Option
 - Error Code

BUG-50758 APRE에서 double quote 문자에 대한 escape 처리 누락

- **module** : mm-apre
- **Category** : Functional Error
- **재현 빈도** : Always
- **설명** : EXEC SQL 구문에 "(double quote)문자가 포함된 경우, apre precompile후에 생성되는 .c, .cpp 파일에 escape 문자가 자동으로 추가되도록 수정하였습니다.

예)

.sc 파일 : EXEC SQL SELECT ' ' ' FROM DUAL;

==>

.c 파일 : ulpSqlstmt.stmt = (char *)"SELECT ' ' ' FROM DUAL";

- **재현 방법**
 - **재현 절차**
.sc 파일에서 아래와 같이 EXEC SQL 문을 작성한다.

```
int main()
{
    EXEC SQL SELECT ' ' ' FROM DUAL;
}
```

- **수행 결과**
.c 파일에서 아래와 같이 변환된다.

```

int main()
{
    /* SELECT ' ' ' FROM DUAL; */
    {
        struct ulpSqlstmt ulpSqlstmt;
        memset(&ulpSqlstmt, 0, sizeof(ulpSqlstmt));
        ulpSqlstmt.stmttype = 4;
        ulpSqlstmt.stmtname = NULL;
        ulpSqlstmt.ismt = 0;
        ulpSqlstmt.numofhostvar = 0;
        ulpSqlstmt.statusptr = NULL;
        ulpSqlstmt.errcodeptr = NULL;
        ulpSqlstmt.isatomic = 0;
        ulpSqlstmt.stmt = (char *)"SELECT ' ' ' FROM DUAL";
        ...
    }
}

```

- 예상 결과

.c 파일에서 아래와 같이 " 앞에 \이 추가된 것을 확인할 수 있다.

```

int main()
{
    /* SELECT ' ' ' FROM DUAL; */
    {
        struct ulpSqlstmt ulpSqlstmt;
        memset(&ulpSqlstmt, 0, sizeof(ulpSqlstmt));
        ulpSqlstmt.stmttype = 4;
        ulpSqlstmt.stmtname = NULL;
        ulpSqlstmt.ismt = 0;
        ulpSqlstmt.numofhostvar = 0;
        ulpSqlstmt.statusptr = NULL;
        ulpSqlstmt.errcodeptr = NULL;
        ulpSqlstmt.isatomic = 0;
        ulpSqlstmt.stmt = (char *)"SELECT ' \' ' FROM DUAL";
        ...
    }
}

```

- Workaround

- 변경사항

- Performance view
- Property
- Compile Option
- Error Code

BUG-50783 프로시저 실행 중 Table에 DDL 수행시, Invalid use of host variables error 가 발생하는 문제를 수정합니다.

- **module** : qp
- **Category** : Functional Error
- **재현 빈도** : Always
- **설명** : 프로시저 실행 중 테이블에 DDL 구문 수행시, ERR-3123B : Invalid use of host variables 에러가 발생하는 문제를 수정합니다.
- **재현 방법**
 - 재현 절차

```
-- session 1
CREATE TABLE T1 (I1 INTEGER);
INSERT INTO T1 VALUES(1);
CREATE OR REPLACE PROCEDURE PROC1 AS
VAR1 INTEGER;
VAR2 INTEGER;
BEGIN
    VAR1 := 1;
    FOR I IN 1 .. 10 LOOP
        SELECT VAR1 INTO VAR2 FROM T1;
        PRINTLN(VAR2);
        SLEEP(2);
        VAR1 := VAR1 + 1;
    END LOOP;
END;
/
EXEC PROC1;

-- session 2
ALTER TABLE T1 ADD (I2 INTEGER);
```

- 수행 결과

```
iSQL> EXEC PROC1;
1
2
3
4
[ERR-3123B : Invalid use of host variables
0001 : select :B0 from T1;
          ^  ^

at "SYS.PROC1", line 7]
```

- 예상 결과


```
iSQL> EXEC PROC1;
1
2
3
4
5
6
7
8
9
10
Execute success.
```

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

BUG-50791 TO_CHAR 함수를 ALIAS로 조건절에 사용시 뷰에 대한 조건절 Pushdown 이 2번 이상 발생할 경우, 비정상 종료되는 문제를 수정합니다.

- **module** : qp
- **Category** : Fatal
- **재현 빈도** : Always
- **설명** : TO_CHAR 함수를 alias로 조건절에 사용시 뷰에 대한 조건절 Pushdown이 2번 이상 발생할 경우, 서버가 비정상 종료되는 문제를 수정합니다. 이 버그는 아래의 4가지 조건이 모두 일치하는 경우에만 발생합니다.
 1. TO_CHAR 함수의 2번째 인자로 숫자 형식이나 날짜 형식이 있는 경우
 2. TO_CHAR 함수의 1번째 인자가 Numeric 타입이 아닌 경우
 3. TO_CHAR 함수의 1번째 인자의 컬럼이 서브쿼리의 뷰에 존재하며, 뷰 머징이 되지 않고 Pushdown되는 경우
 4. 서브쿼리의 타겟절에서 TO_CHAR 함수를 사용하고 ALIAS를 지정하며, 이 ALIAS를 메인쿼리의 조건절에서 사용할 때, TO_CHAR 함수가 사용된 뷰로 Pushdown 되는 경우
- **재현 방법**
 - **재현 절차**

```

select * from (
    select
        (select c.USER_NAME from SYSTEM_.SYS_COMMENTS_ c where
c.table_name = a.table_name and c.COLUMN_NAME is null limit 1) as usernm,
        a.table_name,
        (select c.comments from SYSTEM_.SYS_COMMENTS_ c where
c.table_name = a.table_name and c.COLUMN_NAME is null limit 1) as
tablenm,

to_char((b.fixed_alloc_mem+b.var_alloc_mem)/1024/1024, '999,999,999') ||
'MB' as alloc,

to_char((b.fixed_used_mem+b.var_used_mem)/1024/1024, '999,999,999') ||
'MB' as used
    from system_.sys_tables_ a, v$membtl_info b
    where a.table_oid = b.table_oid
        and a.user_id <> (select user_id from system_.sys_users_ where
user_name = 'SYSTEM_')
        and a.table_type = 'T'
    order by (b.fixed_alloc_mem+b.var_alloc_mem) desc)
    where alloc != 'OMB';

```

수행 결과

AIX 환경에서만 비정상 종료 합니다.

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

예상 결과

```
No rows selected.
```

• Workaround

NO_PUSH_SELECT_VIEW hint 사용

```

select /*+ NO_PUSH_SELECT_VIEW(AA) */ * from (
    select /*+ */
        (select /*+ */c.USER_NAME from SYSTEM_.SYS_COMMENTS_ c where
c.table_name = a.table_name and c.COLUMN_NAME is null limit 1) as usernm,
        a.table_name,
        (select /*+ */c.comments from SYSTEM_.SYS_COMMENTS_ c where
c.table_name = a.table_name and c.COLUMN_NAME is null limit 1) as tablenm,

to_char((b.fixed_alloc_mem+b.var_alloc_mem)/1024/1024, '999,999,999') || 'MB'
as alloc,

to_char((b.fixed_used_mem+b.var_used_mem)/1024/1024, '999,999,999')
|| 'MB' as used
    from system_.sys_tables_ a, v$membtl_info b
    where a.table_oid = b.table_oid
        and a.user_id <> (select /*+ */user_id from system_.sys_users_ where
user_name = 'SYSTEM_')
        and a.table_type = 'T'
    order by (b.fixed_alloc_mem+b.var_alloc_mem) desc) AA
    where alloc != 'OMB';

```

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

BUG-50844 ADO.net의 AltibaseBulkCopy 사용 후 메모리 사용량이 급증하는 문제를 수정합니다.

- **module** : ux-win-adonet
- **Category** : Memory Error
- **재현 빈도** : Always
- **설명** : ADO.net의 AltibaseBulkCopy 사용 후 메모리 사용량이 급증하는 문제를 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
6.5.1.10.1	6.3.1	8.1.1	7.1.3	7.4.5

Altibase 6.5.1 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