

- Altibase 7.1.0.1.4 Patch Notes

- New Features

- BUG-45779 APRE에서 PSM 인자로 Array type을 지원해야 합니다.
- BUG-45701 PSM associative array parameter binding
- BUG-46032 bulk collect into 절에서 2-level associative array variable 지원
- BUG-45946 이중화를 통하여 DDL 동기화(DDL SYNCHRONIZATION)를 허용합니다.
- BUG-46308 Partition Merge / Split DDL 이중화 지원
- BUG-45972 Standalone 에서만 동작하는 DDL 구문의 처리
- BUG-45984 이중화에서 InfiniBand(IB) 통신 방법을 지원한다.
- BUG-46209 이중화 infiniband 통신방식 관련 추가된 구문을 APRE와 AEXPORT에도 추가 해야 합니다.
- BUG-46011 CLI Deferred Prepare 제약사항 제거
- BUG-46016 계층형 쿼리 (Hierarchy Query) 조인에서 Lob 제약을 제거합니다.
- BUG-46065 HASH 가 적용된 RANGE PARTITION 지원
- BUG-46074 Multiple trigger event 지원 및 DBMS\_STANDARD 패키지 추가
- BUG-46095 Adapter에 Propagation 기능이 필요 합니다.
- BUG-46151 failover 동작 변경: LoadBalance 추가
- BUG-46154 DBMS\_OUTPUT 패키지에 print\_enable/print\_disable 프로시저 추가
- BUG-46158 Plan cache내 plan을 keep 할 수 있는 기능을 제공
- BUG-46163 IPCDA에 Queue를 지원해야 합니다.
- BUG-46229 MERGE 구문에서의 메모리 재사용 개선
- BUG-46245 jdbcAdapter 에도 oraAdapter와 동일하게 실행시에 sql 스크립트를 실행할 수 있는 기능이 필요합니다.
- BUG-46273 LOCK TABLE 구문으로 특정 Partition에 Lock을 잡는 기능을 추가합니다.
- BUG-46067 replication sync시 lock 획득을 실패하는 메세지에 table 명이 없어서 분석이 어렵습니다.

- Fixed Bugs

- BUG-45053 병렬적용자(Parallel Applier) 옵션을 가진 이중화에서 스레드 시작 함수 에서 실패하는 경우, 스레드를 반납 하는 로직이 누락되어 있는 문제를 수정하였습니다.
- BUG-45060 offline replication start와 replication drop을 동시에 수행하는 경우, replication start가 완료되지 않았으면 replication drop을 수행하지 못하도록 수정하였습니다.
- BUG-45264 서버 시작시 temp 파일이 존재하지 않으면 자동으로 생성합니다.
- BUG-45643 암호화컬럼의 경우, 이중화 환경에서 DDL 수행시 Replication HandShake 가 실패하는 문제가 있어 수정하였습니다.
- BUG-45652 이중화에서 Active Server와 Standby Server의 List Partition 테이블의 범위 조건이 다른경우에 Handshake 가 성공되는 문제가 있어, 이를 수정하였습니다.

- BUG-45898 Fetch시 예외상황으로 인해 Hang이나 프로토콜이 잘못 해석되는 것을 예방해야 합니다.
- BUG-45948 클라이언트 시작 후 서버로 DDL 동기화(sync)를 수행하면 비정상 종료할 수 있습니다.
- BUG-45961 fetch across rollback 기능의 cursor 재사용 조건 확인 개선
- BUG-46010 HBT에서 Connection 진행중에 event가 발생하는 경우, 에러로 처리 하지 않도록 수정하였습니다.
- BUG-46023 select union all을 사용하고 view force 옵션으로 없는 컬럼의 table을 view로 생성시 비정상 종료가능성이 있습니다.
- BUG-46051 dblink 에서 savepoint rollback 실패 후, select 결과 다른 문제
- BUG-46068 DISK UNIQUE INDEX 이용중 재사용된 DATA에 접근하는 경우, 낮은 확률로 비정상 종료되는 오류를 수정하였습니다.
- BUG-46113 NCHAR, NVARCHAR 스키마에 SQLSetDescField()를 사용했을 때 Segmentation fault가 발생하는 문제가 있어, 이를 수정하였습니다.
- BUG-46123 JDBC에서 DatabaseMetaData.getTypeInfo() 함수의 결과의 타입 변환시 에러 발생
- BUG-46130 dumptrc 가 powerpc linux 에서 정상 동작하지 않습니다.
- BUG-46157 create replication시 propagation 관련 구문을 APRE와 AEXPORT에도 추가해야 합니다.
- BUG-46161 rdma library의 rshutdown()에서 Segmentation fault가 발생하는 문제를 수정합니다.
- BUG-46184 계층형 쿼리에서 order sibling by 절에 정상컬럼과 무의미한 컬럼을 함께 사용하는 경우, 비정상 종료되는 경우가 있어 수정하였습니다.
- BUG-46197 altilinker가 구동중이지 않은경우에 disconnect을 수행하면, altilinker를 다시 connect 하는 문제가 있어 수정하였습니다.
- BUG-46202 v\$event\_name에 중복된 항목이 있습니다.
- BUG-46230 iduMemory의 getStatus(), setStatus() 함수 에러메시지 세분화
- BUG-46239 DDL 동기화 중 Select 시, Lock 획득에 실패하는 경우 DDL 동기화가 실패할 수 있습니다.
- BUG-46242 DDL 동기화 시 Propagation 옵션이 켜져 있으면 실패해야 합니다.
- BUG-46249 group by 표현식에 있는 컬럼을 참조하는 집합 연산이 존재하지 않으면 결과 값 오류가 발생할 수 있습니다.
- BUG-46264 디스크테이블에서 window sort의 order by 절에 동일한 컬럼을 중복하여 나열하는 경우, hang이 발생할 수 있습니다.
- BUG-46265 Create Disk Temp Table 에서 Key column list가 순환 할 수도 있는지 검증합니다.
- BUG-46274 Partition swap 후 update 수행 시 column not found 에러가 발생합니다.
- BUG-46279 Disk temp 사용 시 grouping 데이터의 정렬이 subquery를 참조하는 경우 결과 값 오류가 발생합니다.

- [BUG-46286](#) insert ~ select 구문에서 테이블 이름 없이 parallel 힌트를 사용하면 에러가 발생합니다
- [Changes](#)
  - [Version Info](#)
  - [호환성](#)
  - [프로퍼티](#)
  - [성능 뷰](#)

# Altibase 7.1.0.1.4 Patch Notes

## New Features

### BUG-45779 APRE에서 PSM 인자로 Array type을 지원해야 합니다.

- **module** : mm-apre
- **Category** : Functionality
- **재현 빈도** : Always
- **증상** : APRE에서 psm 인자로 array type argument를 사용할 수 있는 기능이 추가되었습니다.
- **재현 방법**
  - 재현 절차
  - 수행 결과
  - 예상 결과
- **Workaround**
- **변경사항**
  - Performance view
  - Property
  - Compile Option
  - Error Code
    - 121, HY000, ulpERR\_ABORT\_Not\_Supported\_Host\_Var\_Type = Currently not supported host variable type.

# \*Cause: Currently not supported host variable type.

# \*Action: Refer to the Precompiler user's manual for available host variables.

- 122, HY000, ulpERR\_ABORT\_Column\_Value\_Is\_Null = The fetched result contains a NULL value. or Fetch column value is NULL.

# \*Cause: The fetched result contains a NULL value. or Fetch column value is NULL.

# \*Action: Verify that the stored procedure contains NULL.(Revise the cursor definition so that no columns possibly containing NULL values are retrieved.)

- 123, HY000, ulpERR\_ABORT\_Invalid\_PSM\_Array\_Version = Invalid psm array meta version.

# \*Cause: Invalid psm array meta version.

# \*Action: Contact Altibase's Support Center (<http://support.altibase.com>).

- 124, 07006, ulpERR\_ABORT\_Invalid\_PSM\_Array\_Type = The apre type and psm array type do not match.

# \*Cause: The apre host variable type and psm array type do not match.

# \*Action: Change the apre host variable types to the data types that are supported by psm array.

## BUG-45701 PSM associative array parameter binding

- **module** : qp-psm-trigger-pvo
- **Category** : Functionality
- **재현 빈도** : Always
- **증상** : Associative array 타입 인자를 가진 프로시저를 APRE 에서 직접적으로 호출할 수 있는 기능을 추가합니다.

```

create or replace package pkg1
as
type arr1 is table of integer index by integer;
end;
/

create or replace procedure proc1( a pkg1.arr1, b out pkg1.arr1 )
as
begin
println( a.count() );
b := a;
end;
/

EXEC SQL BEGIN DECLARE SECTION;
int args1[1024];
int args2[1024];
EXEC SQL END DECLARE SECTION;
EXEC SQL EXECUTE
BEGIN
    proc1(:args1 in, :args2 out);
END;
END-EXEC;

```

- **재현 방법**

- 재현 절차
- 수행 결과
- 예상 결과

- **Workaround**

- **변경사항**

- Performance view
- Property
- Compile Option
- Error Code
  - 1137,qpERR\_ABORT\_QSX\_ARRAY\_INDEX\_OUT\_OF\_RANGE = Index out of range for host language array.  
# \*Cause: An index in the array is either less than one or greater than the maximum size of the host language array.  
# \*Action: Verify an index in the array or size of the host language array.
  - 1138,qpERR\_ABORT\_QSX\_INVALID\_ARRAY\_BINDING\_PROTOCOL = Invalid host language array binding protocol.  
# \*Cause: Invalid host language array binding protocol.

# \*Action: Check the error number from the trace log and  
contact Altibase's Support Center(<http://support.altibase.com>).

## BUG-46032 bulk collect into 절에서 2-level associative array variable 지원

- **module** : qp-psm-trigger-execute
- **Category** : Functional Error
- **재현 빈도** : Always
- **증상** : PSM 에서 bulk collect into 절을 사용하는 select/select for update 구문과 execute immediate 구문에서 2-level associative array 변수를 지원합니다. 2-level associative array variable을 arrayVariableName[ index ] 또는 arrayVariableName( index ) 형태로 사용한 경우에도 수행될 수 있도록 수정하였습니다.

- **재현 방법**

- **재현 절차**

```
drop table t1;
create table t1 ( c1 int, c2 int, c3 int, c4 int, c5 int, c6 int, c7 int, c8 int, c9 int, c10 int );

insert into t1 select level, level, level, level, level, level, level, level, level, level from dual connect by level <= 10;

create or replace procedure proc1
as
type typ1 is table of int index by int;
type typ2 is table of typ1 index by int;
a typ2;
begin
select * bulk collect into a(1), a(2), a(3), a(4), a(5), a(6), a(7), a(8), a(9), a(10) from t1;
end;
/

exec proc1;
```

- **수행 결과**

```
iSQL> exec proc1;
[ERR-21008 : Data type module (type=1000003) not found.
at "SYS.PROC1", line 7]
```

- **예상 결과**

```
iSQL> exec proc1;
Execute success.
```

- **Workaround**

```

create or replace procedure proc1
as
    type typ1 is table of int index by int;
    type typ2 is table of typ1 index by int;
    a          typ2;
    n1 nocopy typ1;
    n2 nocopy typ1;
    n3 nocopy typ1;
    n4 nocopy typ1;
    n5 nocopy typ1;
    n6 nocopy typ1;
    n7 nocopy typ1;
    n8 nocopy typ1;
    n9 nocopy typ1;
    n10 nocopy typ1;
begin
    n1 := a(1);
    n2 := a(2);
    n3 := a(3);
    n4 := a(4);
    n5 := a(5);
    n6 := a(6);
    n7 := a(7);
    n8 := a(8);
    n9 := a(9);
    n10 := a(10);
    select * bulk collect into n1, n2, n3, n4, n5, n6, n7, n8, n9, n10 from t1;
end;
/

```

- **변경사항**

- Performance view
- Property
- Compile Option
- Error Code

## BUG-45946 이중화를 통하여 DDL 동기화(DDL SYNCHRONIZATION)를 허용합니다.

- **module** : rp
- **Category** : Other
- **재현 빈도** : Always
- **증상** : 이중화를 통하여 DDL 동기화(Synchronization)를 허용합니다. 이 기능을 사용하기 위해서는 각노드의 REPLICATION\_DDL\_SYNC 프로퍼티를 1로 설정해야 합니다. 또한, 각 노드의

REPLICATION\_DDL\_ENABLE 프로퍼티를 1로 설정하고, REPLICATION\_DDL\_ENABLE\_LEVEL 이 동일해야 합니다.

DDL 동기화를 사용하기 위해 다음의 제약 조건을 따릅니다.

- DDL 동기화를 수행할 노드들의 이중화가 동작하고 있어야 한다.
- DDL 동기화를 수행할 Local 노드와 Remote 노드의 Table 이름이 같아야 한다.
- DDL 동기화를 수행할 Local 노드와 Remote 노드의 Table 파티션이름이 같아야 한다.
- DDL 동기화를 수행할 이중화 대상 유저의 이름이 같아야 한다.
- 한번에 하나의 노드에서만 DDL 동기화를 수행해야 한다.
- DDL 동기화를 수행할 각 이중화 노드의 REPLICATION\_DDL\_ENABLE과 REPLICATION\_DDL\_ENABLE\_LEVEL 프로퍼티 값이 같아야 한다.
- **Altibase Patch 버전(5자리)이 동일해야한다.**

Propagation 옵션 사용시 DDL 동기화를 허용하지 않는다.

- **재현 방법**

- 재현 절차
- 수행 결과
- 예상 결과

- **Workaround**

- **변경사항**

- Performance view

- 변경

- V\$SESSION

컬럼추가)

- REPLICATION\_DDL\_SYNC : DDL 복제여부

0: DDL 복제를 하지 않음

1: DDL 복제를 한다.

- REPLICATION\_DDL\_TIMEOUT : DDL복제 타임아웃

- Property

- 추가

- REPLICATION\_DDL\_SYNC

- 데이터 타입 :Unsigned Integer

- 기본값 : 0

- 속성 :변경 가능, 단일 값

- 값의 범위 :[0, 1]

- 설명 :DDL 복제 여부를 나타낸다.

0 : DDL 복제를 허용하지 않음. DDL 수행시 Local 노드에서만 수행된다.

1 : DDL 복제를 허용함. DDL 수행시 이중화가 걸린 모든 노드에 DDL 이 복제된다.

- REPLICATION\_DDL\_SYNC\_TIMEOUT

- 데이터 타입 :Unsigned Integer



- 기본값 : 7200
  - 속성 : 변경 가능, 단일 값
  - 값의 범위 : [ 0, 2^32 - 1 ]
  - 설명 : DDL 복제시 수행 시간이 길어지면 DDL 을 수행하는 테이블의 서비스의 중지 시간이 길어지게 된다. 이런 서비스 중지 시간을 막기 위하여 이 값을 설정한다. Altibase 운영 중 ALTER SYSTEM 문 또는 ALTER SESSION 문을 이용하여 이 프로퍼티 값을 변경할 수 있다. 모든 노드의 Timeout 은 DDL 을 수행하는 Active 노드의 Timeout 값으로 적용된다.
- Compile Option
  - Error Code

## BUG-46308 Partition Merge / Split DDL 이중화 지원

- **module** : rp
- **Category** : Other
- **재현 빈도** : Always
- **증상** : DDL Synchronization 에 Partition Merge / Split 을 사용할수 있도록 지원합니다.  
주의사항 : 이중화 갭이 없는 상태에서 패치해야 함.
- **재현 방법**
  - 재현 절차
  - 수행 결과
  - 예상 결과
- **Workaround**
- **변경사항**
  - Performance view
  - Property
  - Compile Option
  - Error Code

## BUG-45972 Standalone 에서만 동작하는 DDL 구문의 처리

- **module** : rp
- **Category** : Other
- **재현 빈도** : Always
- **증상** : 기존에 DDL 동기화가 허용되지 않는 DDL 중 Active 에서만 동작하던 DDL 을 DDL 동기화 허용하도록 변경하였습니다.  
허용하는 DDL List - Aging , Compact, Drop Default, Rename Constraint, SetDefault
- **재현 방법**
  - 재현 절차

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

## BUG-45984 이중화에서 InfiniBand(IB) 통신 방법을 지원한다.

- **module** : rp-control
- **Category** : Efficiency
- **재현 빈도** : Always
- **증상** : 이중화에서 InfiniBand(IB) 통신 방법을 지원합니다. [이중화 생성 구문](#)에서 USING 키워드를 통해 지원합니다.
- **재현 방법**
  - 재현 절차
  - 수행 결과
  - 예상 결과
- **Workaround**
- **변경사항**
  - Performance view
  - Property
    - 추가
      - **REPLICATION\_IB\_PORT\_NO**
        - 데이터 타입 :Unsigned Integer
        - 기본값 : 0
        - 속성 : 읽기 전용, 단일 값
        - 값의 범위 : [ 0, 65535 ]
        - 설명 : 이중화 연결을 InfiniBand로 접속 할 때 지역 서버의 이중화 포트 번호이다. InfiniBand로 이중화를 사용하지 않으려면 이 프로퍼티를 0으로 설정한다. Altibase의 IB\_ENABLE이 1(enable) 이어야 이중화도 InfiniBand로 접속 가능하다.
      - **REPLICATION\_IB\_LATENCY**
        - 데이터 타입: Unsigned Integer
        - 기본값: 0
        - 속성: 읽기 전용, 단일 값

- 값의 범위: [0,1]
  - 설명: rsocket의 RDMA\_LATENCY 옵션값. 이 값이 1이면 CPU자원을 소모하더라도, 적은 대기시간을 사용한다.
- Compile Option
- Error Code
  - 394, rpERR\_ABORT\_RPC\_ROLE\_NOT\_SUPPORT\_IB = InfiniBand(IB) can not be used with 'FOR ANALYSIS' syntax.
  - # \*Cause:
  - # - InfiniBand(IB) can not be used with 'FOR ANALYSIS' syntax.
  - # \*Action:
  - # - Check the role of the replication. Do not use the InfiniBand(IB).

## BUG-46209 이중화 infiniband 통신방식 관련 추가된 구문을 APRE 와 AEXPORT에도 추가 해야 합니다.

- **module** : rp
- **Category** : Other
- **재현 빈도** : Always
- **증상** : infiniband 통신방식 지원으로 이중화 생성구문에 USING이 추가되었는데, 이 구문이 APRE 와 AEXPORT 에서도 동작하도록 적용합니다.
- **재현 방법**
  - 재현 절차
  - 수행 결과
  - 예상 결과
- **Workaround**
- **변경사항**
  - Performance view
  - Property
  - Compile Option
  - Error Code

## BUG-46011 CLI Deferred Prepare 제약사항 제거

- **module** : mm-cli
- **Category** : Enhancement
- **재현 빈도** : Always
- **증상** : deferred prepare로 여러개의 statement를 사용하면 낮은 확률로 제대로 동작하지 않는 문제가 있는데, 이를 수정하였습니다.
- **재현 방법**

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

## BUG-46016 계층형 쿼리 (Hierarchy Query) 조인에서 Lob 제약을 제거합니다.

- module : qp
- Category : Enhancement
- 재현 빈도 : Always
- 증상 : 계층형 쿼리 (Hierarchy Query) 조인에서 Lob이 사용될 경우, 에러를 출력하는 문제를 수정 하였습니다.
- 재현 방법
  - 재현 절차

```
create table t1 ( i1 integer, i2 integer, i3 clob, i4 blob );
insert into t1 values ( 2 , 1, 7 , '8');
insert into t1 values ( 3 , 2, 17, '9');
insert into t1 values ( 4 , 3, 27, '9');
insert into t1 values ( 5 , 4, 37, '9');
insert into t1 values ( 6 , 5, 47, '9');
```

```
create table t2 ( i1 integer, i2 integer);
insert into t2 values ( 1 , 47);
insert into t2 values ( 2 , 36);
insert into t2 values ( 3 , 26);
insert into t2 values ( 4 , 16);
insert into t2 values ( 47, 6);
```

```
select cast(t1.i3 as varchar(10)) from t1 left outer join t2 on t1.i1 = t2.i1 + 1 connect by
select t1.i3, sum(t1.i1 ) over () from t1 left outer join t2 on t1.i1 = t2.i1 + 1 connect by
```

- 수행 결과

```
iSQL> select cast(t1.i3 as varchar(10)) from t1 left outer join t2 on t1.i1 = t2.i1 + 1
[ERR-31318 : Unexpected errors may have occurred: qtc::fixAfterValidation: tuple row si
iSQL> select t1.i3, sum(t1.i1 ) over () from t1 left outer join t2 on t1.i1 = t2.i1 + 1
[ERR-31318 : Unexpected errors may have occurred: qtc::fixAfterValidation: tuple row si
```

- 예상 결과

```
iSQL> select cast(t1.i3 as varchar(10)) from t1 left outer join t2 on t1.i1 = t2.i1 + 1  
CAST(T1.I3ASVARCHAR(10))
```

-----

7

17

27

37

17

27

37

27

37

37

47

11 rows selected.

```
iSQL> select t1.i3, sum(t1.i1 ) over () from t1 left outer join t2 on t1.i1 = t2.i1 + 1  
I3
```

-----

SUM(T1.I1)OVER()

-----

7

46

17

46

27

46

37

46

17

46

27

46

37

46

27

46

37

46

37

46

47

46

11 rows selected.

- Workaround

- 변경사항

- Performance view
- Property

- Compile Option
- Error Code

## BUG-46065 HASH 가 적용된 RANGE PARTITION 지원

- **module** : qp
- **Category** : Enhancement
- **재현 빈도** : Always
- **증상** : 파티션드 객체에 RANGE\_USING\_HASH PARTITION 이 추가되었습니다. 해시를 사용한 범위 파티셔닝(RANGE\_USING\_HASH PARTITION)은 해당 해시(hash) 값을 기준으로 범위(range)를 정해서 분할하는 방법입니다.  
예) create table h1( i1 varchar(10) primary key, i2 varchar(10), i3 varchar(10) ) partition by range\_using\_hash(i1) (  
partition p1 values less than ('300'),  
partition p2 values less than ('700'),  
partition p3 values default  
);
- **재현 방법**
  - 재현 절차
  - 수행 결과
  - 예상 결과
- **Workaround**
- **변경사항**
  - Performance view
  - Property
  - Compile Option
  - Error Code

## BUG-46074 Multiple trigger event 지원 및 DBMS\_STANDARD 패키지 추가

- **module** : qp-psm-trigger-pvo
  - **Category** : Functionality
  - **재현 빈도** : Always
  - **증상** : 1. Trigger 생성구문에 trigger event 에 multiple trigger event를 지원합니다.  
2. DBMS\_STANDARD 패키지를 통해서 trigger event를 확인하는 함수를 제공합니다.  
DBMS\_STANDARD 패키지를 구성하는 프로시저와 함수는 아래의 표와 같이 제공합니다.
-

프로시저 및 함수	설명
DELETING	Trigger가 DELETE로부터 시작한 것인지를 반환한다.
INSERTING	Trigger가 INSERT로부터 시작한 것인지를 반환한다.
UPDATING	Trigger가 UPDATE로부터 시작한 것인지를 반환한다.
UPDATING	Trigger가 특정 컬럼의 UPDATE로부터 시작한 것인지를 반환한다.

## • 재현 방법

### ◦ 재현 절차

```
CREATE TABLE T1 (C1 INTEGER, C2 INTEGER, C3 INTEGER);
CREATE OR REPLACE TRIGGER TRIG1
BEFORE INSERT OR UPDATE OR DELETE ON T1
FOR EACH ROW BEGIN
    NULL;
END;
/
```

### ◦ 수행 결과

```
[ERR-31001 : SQL syntax error

line 2: parse error
BEFORE INSERT OR UPDATE OR DELETE ON T1
                ^^

]
```

### ◦ 예상 결과

```
iSQL> CREATE TABLE T1 (C1 INTEGER, C2 INTEGER, C3 INTEGER);
Create success.
iSQL> CREATE OR REPLACE TRIGGER TRIG1
BEFORE INSERT OR UPDATE OR DELETE ON T1
FOR EACH ROW BEGIN
    NULL;
END;
/
Create success.
```

## • Workaround

```
CREATE OR REPLACE TRIGGER TRIG1
BEFORE INSERT ON T1
FOR EACH ROW BEGIN
    NULL;
END;
/
```

```
CREATE OR REPLACE TRIGGER TRIG1
BEFORE UPDATE ON T1
FOR EACH ROW BEGIN
    NULL;
END;
/
```

```
CREATE OR REPLACE TRIGGER TRIG1
BEFORE DELETE ON T1
FOR EACH ROW BEGIN
    NULL;
END;
/
```

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

## BUG-46095 Adapter에 Propagation 기능이 필요합니다.

- **module** : rp-ala
- **Category** : Other
- **재현 빈도** : Always
- **증상** : **CREATE REPLICATION**시 이중화 수신자가 전송받은 로그를 복제하기 위해 **FOR PROPAGABLE LOGGING** 을 사용하여 로그를 기록한 후, 복제된 로그가 Adapter를 통해 다른 원격 서버로 전송하기 위해 **FOR ANALYSIS PROPAGATION**을 사용할 수 있다.
- **재현 방법**
  - 재현 절차
  - 수행 결과
  - 예상 결과
- **Workaround**
- **변경사항**
  - Performance view
  - Property



- Compile Option
- Error Code

## BUG-46151 failover 동작 변경: LoadBalance 추가

- **module** : mm
- **Category** : Functionality
- **재현 빈도** : Always
- **증상** : LoadBalance 지원
- **재현 방법**
  - 재현 절차
  - 수행 결과
  - 예상 결과
- **Workaround**
- **변경사항**
  - Performance view
  - Property
    - LoadBalance: on/off
  - Compile Option
  - Error Code

## BUG-46154 DBMS\_OUTPUT 패키지에 print\_enable/print\_disable 프로시저 추가

- **module** : qx
- **Category** : Functionality
- **재현 빈도** : Always
- **증상** : DBMS\_OUTPUT 패키지에 print\_enable/print\_disable 프로시저를 추가하여, PSM내에서 println 기능을 enable, disable 할수 있는 기능을 제공합니다. 세션단위로 수행됩니다.  
구문)  
DBMS\_OUTPUT.PRINT\_ENABLE  
DBMS\_OUTPUT.PRINT\_DISABLE
- **재현 방법**
  - 재현 절차
  - 수행 결과
  - 예상 결과
- **Workaround**
- **변경사항**
  - Performance view

- Property
- Compile Option
- Error Code

## BUG-46158 Plan cache내 plan을 keep 할 수 있는 기능을 제공

- **module** : mm-plancache
- **Category** : Functionality
- **재현 빈도** : Always
- **증상** : SQL PLAN CACHE에 PLAN을 keep/unkeep 할 수 있는 기능이 추가되었습니다.

이에 dbms\_sql\_plan\_cache 패키지와 plan\_cache\_keep 힌트가 추가되었습니다.

### 1. Package

```
create or replace package dbms_sql_plan_cache authid
current_user
as
procedure keep_plan( sql_text_id varchar(64) );
procedure unkeep_plan( sql_text_id varchar(64) );
end dbms_sql_plan_cache;
/
```

### 2. Hint

**PLAN\_CACHE\_KEEP 힌트**는 생성된 플랜이 Victim 선정 과정에서 제외되어 Plan cache내에 유지될 수 있도록 지시하는 힌트입니다.

PLAN\_CACHE\_KEEP 힌트는 Hardprepare 과정에서 적용됩니다.

사용자가 해당 Plan을 Unkeep으로 전환했을 때 Softprepare가 발생해도 다시 Keep 상태로 전환되지 않습니다.

예) select /\*+ plan\_cache\_keep \*/ \* from t1;

또한 v\$sql\_plan\_cache\_sqltext , v\$sql\_plan\_cache\_pco 에 PLAN CACHE KEEP 기능과 관련한 속성이 추가되었습니다.

- **재현 방법**
  - 재현 절차
  - 수행 결과
  - 예상 결과
- **Workaround**
- **변경사항**
  - Performance view
    - 변경
      - **V\$SQL\_PLAN\_CACHE\_SQLTEXT** 에 속성 추가됨  
PLAN\_CACHE\_KEEP | VARCHAR(6)

SQL\_TEXT\_ID에 해당하는 Plan cache 객체의 Keep 상태를 나타내며 'KEEP', 'UNKEEP'의 값을 가질 수 있다.

KEEP : PLAN이 Keep 되어 있는 상태로 Victim에 선정되지 않는다.

UNKEEP: PLAN이 Unkeep 되어 있는 상태로 Victim에 선정될 수 있다.

- **V\$SQL\_PLAN\_CACHE\_PCO**에 속성 추가됨

PLAN\_SIZE INTEGER

FIX\_COUNT INTEGER

PLAN\_CACHE\_KEEP VARCHAR(6)

PLAN\_SIZE : Plan cache 객체의 plan 크기를 나타낸다.

FIX\_COUNT : Plan cache 객체를 참조 중인 Statement 수를 나타낸다.

PLAN\_FIX\_COUNT가 1 이상이면 Victim에 선정되지 않는다.

PLAN\_CACHE\_KEEP : SQL\_TEXT\_ID에 해당하는 Plan cache 객체의 Keep 상태를 나타내며 'KEEP', 'UNKEEP'의 값을 가질 수 있다.

KEEP: PLAN이 Keep 되어 있는 상태로, Victim에 선정되지 않는다.

UNKEEP: PLAN이 Unkeep 되어 있는 상태로, Victim에 선정될 수 있다

- Property
- Compile Option
- Error Code

## BUG-46163 IPCDA에 Queue를 지원해야 합니다.

- **module** : mm-queue
- **Category** : Functionality
- **재현 빈도** : Always
- **증상** : ConnType=7(IPCDA)로 접속하였을 경우에도 Queue기능이 동작하도록 합니다.
- **재현 방법**
  - 재현 절차
  - 수행 결과
  - 예상 결과
- **Workaround**
- **변경사항**
  - Performance view
  - Property
  - Compile Option
  - Error Code

## BUG-46229 MERGE 구문에서의 메모리 재사용 개선

- **module** : qp-dml-execute

- **Category** : Maintainability
- **재현 빈도** : Always
- **증상** : Merge 함수에서 메모리 재사용 코드를 적합한 값으로 사용하도록 조정하였습니다.
- **재현 방법**
  - 재현 절차
  - 수행 결과
  - 예상 결과
- **Workaround**
- **변경사항**
  - Performance view
  - Property
  - Compile Option
  - Error Code

## BUG-46245 jdbcAdapter 에도 oraAdapter와 동일하게 실행시에 sql 스크립트를 실행할 수 있는 기능이 필요합니다.

- **module** : rp-jdbcAdapter
- **Category** : Other
- **재현 빈도** : Always
- **증상** : jdbcAdpater 시작 시 초기화 작업을 위하여 DB 관리자에 의해 생성된 전역 스크립트 파일인 oalogin.sql의 사용을 지원합니다.  
jdbcAdpater는 임의의 사용자가 jdbcAdpater를 구동할 때마다 다른 데이터베이스에 접속하여 해당 스크립트를 실행합니다. 스크립트 실행도중 실패하게되면 jdbcAdapter.trc 파일에 에러 내용을 기록하고 jdbcAdapter의 실행을 종료합니다. 스크립트 파일은 \$ORA\_ADAPTER\_HOME/conf 밑에 위치 해야하며 이  
파일이 존재하지 않아도 동작합니다.
- **재현 방법**
  - 재현 절차
  - 수행 결과
  - 예상 결과
- **Workaround**
- **변경사항**
  - Performance view
  - Property
  - Compile Option
  - Error Code

## BUG-46273 LOCK TABLE 구문으로 특정 Partition에 Lock을 잡는 기능을 추가합니다.

- **module** : qx
- **Category** : Functionality
- **재현 빈도** : Always
- **증상** : **LOCK TABLE** 구문에 특정 Partition 이름을 명시하여, Partition에 S/X Lock을 잡는 기능을 추가합니다.

구문

```
lock_table ::= LOCK TABLE [ user_name . ] tbl_name [PARTITION ( partition_name ) ] IN
```

```
lock_mode MODE
```

```
{ WAIT integer |NOWAIT } UNTIL NEXT DDL;
```

설명

partition\_name

잠금(LOCK)이 걸릴 파티션의 이름을 명시한다. 파티션의 이름을 명시하면, 파티션에 잠금 모드를 적용하고 테이블에는 ROW SHARE 또는 ROW EXCLUSIVE를 적용한다.

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

## BUG-46067 replication sync시 lock 획득을 실패하는 메시지에 table 명이 없어서 분석이 어렵습니다.

- **module** : rp-sender
- **Category** : Maintainability
- **재현 빈도** : Frequence
- **증상** : SYNC 도중 lock 실패 시 altibase\_rp.log에 찍히는 에러 메시지에 사용자와 테이블 이름이 출력되도록 수정하였습니다.

변경 전)

```
[SenderSyncParallel] lockTable() error in allocSCN()
```

변경 후)

```
[SenderSyncParallel] Failed to lock table for replication synchronization. (User:○○○, Table:○○○)
```

[SenderSyncParallel] Failed to lock table for replication synchronization. (User:○○○, Table:○○○, Partition:○○○) - 파티션 테이블테이블

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

## Fixed Bugs

**BUG-45053 병렬적용자(Parallel Applier) 옵션을 가진 이중화에서 스레드 시작 함수 에서 실패하는 경우, 스레드를 반납 하는 로직이 누락되어 있는 문제를 수정하였습니다.**

- **module** : rp
- **Category** : Fatal
- **재현 빈도** : Always
- **증상** : 병렬적용자(Parallel Applier) 옵션을 가진 이중화에서 한번에 여러 스레드를 시작시키다가 실패시, 스레드를 반납하는 로직이 누락되어 있었습니다. 이로 인해 서버가 내려갈때(종료될 때), 반납되지 않은 스레드들이 존재하여 서버가 종료되지 않고 계속 대기 할 수 있어, 이를 수정하였습니다.

스레드 시작함수에서 실패시, 시작이 성공한 스레드에 대해서 반납하도록 수정하였습니다.

- **재현 방법**
  - **재현 절차**
    1. rpxReceiver::startApplier 중 thread start 시작
    2. server stop
  - **수행 결과**

Hang
  - **예상 결과**

server stop 성공

- **Workaround**

없음

- **변경사항**

- Performance view
- Property
- Compile Option
- Error Code

**BUG-45060** offline replication start와 replication drop을 동시에 수행하는 경우, replication start가 완료되지 않았으면 replication drop을 수행하지 못하도록 수정하였습니다.

- **module** : rp
- **Category** : Fatal
- **재현 빈도** : Frequence
- **증상** : 오프라인 이중화에서 replication start와 replication drop을 동시에 수행하는 경우, offline replication start가 진행중일때 replication drop이 관련 자료구조를 삭제하여, 예기치 못한 오류가 발생할 수 있습니다. 이에 replication start가 완료되지 않았으면 replication drop을 수행하지 못하도록 수정하였습니다.
- **재현 방법**
  - 재현 절차
  - 수행 결과
  - 예상 결과
- **Workaround**
- **변경사항**
  - Performance view
  - Property
  - Compile Option
  - Error Code

**BUG-45264** 서버 시작시 temp 파일이 존재하지 않으면 자동으로 생성합니다.

- **module** : sm
- **Category** : Enhancement
- **재현 빈도** : Always
- **증상** : 서버 시작시 temp 파일이 존재하지 않으면 비정상 종료하는 문제가 있어, 이를 수정하였습니다.

서버 시작시 temp 파일이 존재하지 않으면, 자동으로 생성합니다.

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

## **BUG-45643 암호화컬럼의 경우, 이중화 환경에서 DDL 수행시 Replication HandShake 가 실패하는 문제가 있어 수정하였습니다.**

- **module** : rp-sender
- **Category** : Functional Error
- **재현 빈도** : Always
- **증상** : 이중화 환경에서 DDL 수행시, 암호화 컬럼의 경우가 고려되지 않아 Replication Handshake 실패하는 문제가 있습니다. 이에 암호화 컬럼이 있는 경우 이중화 환경에서 DDL 수행시, 암호화 컬럼에 대한 메타정보를 올바르게 재구축하여 Replication HandShake가 실패하지 않도록 수정하였습니다.
- **재현 방법**
  - 재현 절차
  - 수행 결과
  - 예상 결과
- **Workaround**
- **변경사항**
  - Performance view
  - Property
  - Compile Option
  - Error Code

## **BUG-45652 이중화에서 Active Server와 Standby Server의 List Partition 테이블의 범위 조건이 다른경우에 Handshake 가 성공되는 문제가 있어, 이를 수정하였습니다.**

- **module** : rp-receiver
- **Category** : Functional Error



- **재현 빈도** : Always
- **증상** : 이중화에서 Active Server와 Standby Server의 List Partition 테이블의 범위가 포함관계에 있는 경우, 두 partition의 범위가 달라도 HandShake 가 성공하는 문제가 있습니다. 이에, List Partition 테이블의 경우 Active Server와 Standby Server의 범위가 같은 경우에만 Handshake가 성공하도록 수정하였습니다.
- **재현 방법**
  - 재현 절차
  - 수행 결과
  - 예상 결과
- **Workaround**
- **변경사항**
  - Performance view
  - Property
  - Compile Option
  - Error Code

## BUG-45898 Fetch시 예외상황으로 인해 Hang이나 프로토콜이 잘못 해석되는 것을 예방해야 합니다.

- **module** : mm
- **Category** : Functional Error
- **재현 빈도** : Rare
- **증상** : LOB Fetch시 LOB Legth를 가져오는 함수에서 에러가 발생한 경우 Hang이나 프로토콜 해석이 잘못 될 수 있습니다. 에러가 발생하였을 때는 접속을 끊도록 수정 하였습니다.
- **재현 방법**
  - 재현 절차

```
CREATE TABLE LOB_T (C1 VARCHAR(10) PRIMARY KEY, C2 CLOB );
INSERT INTO LOB_T VALUES('A', '111');
INSERT INTO LOB_T VALUES('B', '222');
INSERT INTO LOB_T VALUES('C', '333');

PREFETCH_ROWS=1;
SELECT C1, C2 FROM LOB_T with STMT_A
{
    FETCH RET_C1, RET_C2;
    SELECT C2 FROM LOB_T WHERE C1 = 'RET_C1' with STMT_B
    GETLOB(C2)
    CLOSE STMT_B
}
```

STMT\_A에서 2번째 ROW Fetch시 Hang 발생.

- 수행 결과

Hang or Protocol errors.

- 예상 결과

All rows fetched

- Workaround

```
SELECT C1, C2 FROM LOB_T with STMT_A
{
    FETCH RET_C1, RET_C2;
    GETLOB(C2)
}
```

- 변경사항

- Performance view
- Property
- Compile Option
- Error Code

## BUG-45948 클라이언트 시작 후 서버로 DDL 동기화(sync)를 수행하면 비정상 종료할수 있습니다.

- module : rp
- Category : Fatal
- 재현 빈도 : Always
- 증상 : 클라이언트를 시작후 바로 서버로 DDL 동기화를 수행하면, 잘못된 DDL 동기화 정보를 가져와서 서버가 비정상 종료하는 문제가 있어 수정하였습니다.
- 재현 방법
  - 재현 절차
  - 수행 결과
  - 예상 결과
- Workaround
- 변경사항
  - Performance view
  - Property
  - Compile Option
  - Error Code

## BUG-45961 fetch across rollback 기능의 cursor 재사용 조건 확인 개선

- **module** : sm\_index
- **Category** : Functional Error
- **재현 빈도** : Frequence
- **증상** : fetch across rollback 기능은 내부적으로 rollback으로 인해 view가 깨지는지 확인하는데, 이 로직에 문제가 있어서, 실제 fetch across rollback이 가능함에도 불구하고 가능하지 않다고 판단하는 문제가 있었습니다. 이를 해결하기 위해, 확인 로직부분을 수정하였습니다.
- **재현 방법**
  - 재현 절차
  - 수행 결과
  - 예상 결과
- **Workaround**
- **변경사항**
  - Performance view
  - Property
  - Compile Option
  - Error Code

## BUG-46010 HBT에서 Connection 진행중에 event가 발생하는 경우, 에러로 처리 하지 않도록 수정하였습니다.

- **module** : dm
- **Category** : Functional Error
- **재현 빈도** : Always
- **증상** : HBT 에서 Connection 진행중에 event가 발생하는 경우, 에러로 처리하지 않도록 수정하였습니다.
- **재현 방법**
  - 재현 절차
  - 수행 결과
  - 예상 결과
- **Workaround**
- **변경사항**
  - Performance view
  - Property
  - Compile Option
  - Error Code

# BUG-46023 select union all을 사용하고 view force 옵션으로 없는 컬럼의 table을 view로 생성시 비정상 종료가능성이 있습니다.

- **module** : qp-ddl-dcl-execute
- **Category** : Fatal
- **재현 빈도** : Frequence
- **증상** : select union all을 사용하고 view force 옵션으로 없는 컬럼의 table을 view로 생성시 비정상 종료되는 문제를 수정하였습니다.
- **재현 방법**
  - **재현 절차**

```
-- 반복 해서 수행 시 재현
drop table ncmhem08;
drop view vw$akvital;
create table ncmhem08 (
ncm08status varchar(10),
ncm08idnoa varchar(10),
ncm08idnob varchar(10),
ncm08lwdat varchar(10),
ncm08endat varchar(10),
ncm08entime varchar(10),
ncm08bfr varchar(10),
ncm08bt varchar(10),
ncm08pr varchar(10),
ncm08bpmax varchar(10),
ncm08bpmin varchar(10),
ncm08hd varchar(10),
ncm08memo varchar(10),
ncm08r varchar(10),
ncm08uf varchar(10),
a varchar(10),--a clob,
ncm08df varchar(10) );
create or replace force view vw$akvital (ncm08status, ncm08idnoa, ncm08idnob, ncm08lwdat,
select ncm08status,ncm08idnoa,ncm08idnob,ncm08lwdat,ncm08endat,ncm08entime,ncm08bfr
select ncmhem08.* from ncmhem08                                where ncm08idnoa = '0507376')
```

- **수행 결과**

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

- **예상 결과**

```
iSQL> create or replace force view vw$akvital (ncm08status, ncm08idnoa, ncm08idnob, ncm
select ncm08status,ncm08idnoa,ncm08idnob,ncm08lwdat,ncm08endat,ncm08entime,ncm08bfr
select ncmhem08.* from ncmhem08                                where ncm08idnoa = '0507376'
Create success.
```

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

## **BUG-46051 dblink 에서 savepoint rollback 실패 후, select 결과 다른 문제**

- **module** : dblink
- **Category** : Functional Error
- **재현 빈도** : Always
- **증상** : dblink 에서 savepoint rollback 실패 시 내부적으로 잘못된 global transaction id를 할당받아 total rollback 이 수행되는것이 원인입니다. savepoint rollback 실패시, global transaction id가 잘못 세팅되던 문제를 수정하여 처리하였습니다.
- **재현 방법**
  - 재현 절차
  - 수행 결과
  - 예상 결과
- **Workaround**
- **변경사항**
  - Performance view
  - Property
  - Compile Option
  - Error Code

## **BUG-46068 DISK UNIQUE INDEX 이용중 재사용된 DATA에 접근하는 경우, 낮은 확률로 비정상 종료되는 오류를 수정하였습니다.**

- **module** : sm
- **Category** : Fatal
- **재현 빈도** : Always
- **증상** : 1. DISK UNIQUE INDEX 이용중 RECORD INSERT할때, 낮은 확률로 비정상 종료되는 오류를 수정하였습니다.  
2. 모든 DISK INDEX (NON-UNIQUE INDEX도 포함) SCAN 시에 낮은 확률로 에러메시지를 출력하고 실패하는 오류가 있어, 이를 수정하였습니다.
- **재현 방법**

## ◦ 재현 절차

```
DROP TABLE T;
CREATE TABLE T (ID VARCHAR(30), DATA VARCHAR(12000) ) TABLESPACE SYS_TBS_DISK_DATA PCTF
CREATE UNIQUE INDEX I ON T (ID);
CREATE OR REPLACE PROCEDURE LEGACYTEST()
AS
CURSOR C0 IS
SELECT ID FROM T ORDER BY 1;
V0 VARCHAR(30);
BEGIN
OPEN C0;
FETCH C0 INTO V0;
DELETE FROM T WHERE ID = V0;
COMMIT;
CLOSE C0;
END;
/
INSERT INTO T VALUES(LPAD('3',29), LPAD('D',6000));
INSERT INTO T VALUES(LPAD('1',29), LPAD('D',3000));
INSERT INTO T VALUES(LPAD('2',29), LPAD('D',3000));
AUTOCOMMIT OFF;
EXEC LEGACYTEST();
COMMIT;
AUTOCOMMIT ON;
EXEC SLEEP(5);
UPDATE T SET DATA = LPAD('A', 7500 ) WHERE ID = LPAD('2',29);
UPDATE T SET DATA = NULL WHERE ID = LPAD('2',29);
UPDATE T SET DATA = LPAD('A', 12000 ) WHERE ID = LPAD('3',29);
INSERT INTO T VALUES(LPAD('1',29), NULL);
```

## ◦ 수행 결과

```
iSQL> INSERT INTO T VALUES(LPAD('1',29), NULL);
[ERR-91015 : Communication failure.]
```

## ◦ 예상 결과

```
iSQL> INSERT INTO T VALUES(LPAD('1',29), NULL);
1 row inserted.
```

## • Workaround

### • 변경사항

- Performance view
- Property
- Compile Option
- Error Code

## BUG-46113 NCHAR, NVARCHAR 스키마에 SQLSetDescField()를 사용했을 때 Segmentation fault가 발생하는 문제가 있어, 이를 수정 하였습니다.

- **module** : mm-cli
- **Category** : Fatal
- **재현 빈도** : Always
- **증상** : NCHAR, NVARCHAR 스키마에 SQLSetDescField()를 사용했을 때 Segmentation fault가 발생하는 문제가 있어, 이를 수정하였습니다.
- **재현 방법**
  - 재현 절차
  - 수행 결과
  - 예상 결과
- **Workaround**
- **변경사항**
  - Performance view
  - Property
  - Compile Option
  - Error Code

## BUG-46123 JDBC에서 DatabaseMetaData.getTypeInfo() 함수의 결과의 타입 변환시 에러 발생

- **module** : mm-jdbc
- **Category** : Functional Error
- **재현 빈도** : Always
- **증상** : JDBC에서 DatabaseMetaData의 getTypeInfo() 의 결과값이 제대로 나오지 않던 문제를 수정하였습니다.
- **재현 방법**
  - 재현 절차

```
DatabaseMetaData metaData = connection.getMetaData();

sRS = metaData.getTypeInfo();

/* Fetch all data */
while( sRS.next() )
{
    System.out.println( " name="+sRS.getString(1) +", auto_increment: " + sRS.getBoole
    System.out.println();
}
```

- 수행 결과

```
ERROR MESSAGE : Data conversion failure: Cannot convert the value  to the SQL boolean d
java.sql.SQLException: Data conversion failure: Cannot convert the value  to the SQL bo
    at Altibase.jdbc.driver.ex.Error.throwSQLExceptionInternal(Error.java:171)
    at Altibase.jdbc.driver.ex.Error.throwSQLException(Error.java:151)
```

- 예상 결과

```
name=VARBIT, auto_increment: false
name=NCHAR, auto_increment: false
name=NVARCHAR, auto_increment: false
name=BIT, auto_increment: false
name=BIGINT, auto_increment: false
name=BINARY, auto_increment: false
name=CHAR, auto_increment: false
name=NUMERIC, auto_increment: false
name=INTEGER, auto_increment: false
```

- Workaround

- 변경사항

- Performance view
- Property
- Compile Option
- Error Code

## BUG-46130 dumptrc 가 powerpc linux 에서 정상 동작하지 않습니다.

- **module** : dm
- **Category** : Functional Error
- **재현 빈도** : Always
- **증상** : Power Linux 에서 dumptrc 가 제대로 동작하지 않는 경우가 있어, 수정하였습니다.
- **재현 방법**
  - 재현 절차
  - 수행 결과
  - 예상 결과
- **Workaround**
- **변경사항**
  - Performance view
  - Property
  - Compile Option
  - Error Code



## BUG-46157 create replication시 propagation 관련 구문을 APRE와 AEXPORT에도 추가 해야 합니다.

- **module** : rp
- **Category** : Other
- **재현 빈도** : Always
- **증상** : APRE와 AEXPORT에 create replication시 propagation 관련 구문을 추가해 주었습니다.
- **재현 방법**
  - 재현 절차
  - 수행 결과
  - 예상 결과
- **Workaround**
- **변경사항**
  - Performance view
  - Property
  - Compile Option
  - Error Code

## BUG-46161 rdma library의 rshutdown()에서 Segmentation fault가 발생하는 문제를 수정합니다.

- **module** : cm
- **Category** : Fatal
- **재현 빈도** : Always
- **증상** : rdma library의 rshutdown()에서 Segmentation fault가 발생하는 문제를 수정. (client only)  
본 버그가 재현될 수 있는 상황은 아래 2가지를 모두 만족해야 함.
  1. IB Connection시 Socket 생성 실패 (IB HCA가 없는 장비)
  2. System에 설치된 RDMA library (구 버전?) 이용 (최신 RDMA library에서는 문제 없음)
- **재현 방법**
  - 재현 절차
  - 수행 결과
  - 예상 결과
- **Workaround**
- **변경사항**
  - Performance view
  - Property
  - Compile Option
  - Error Code

# BUG-46184 계층형 쿼리에서 order sibling by 절에 정상컬럼과 무의미한 컬럼을 함께 사용하는 경우, 비정상 종료되는 경우가 있어 수정하였습니다.

- **module** : qp-dml-execute
- **Category** : Fatal
- **재현 빈도** : Always
- **증상** : 계층형 쿼리에서 order sibling by 절에 정상컬럼과 무의미한 컬럼을 함께 사용하는 경우, 비정상 종료되는 문제가 있어 수정하였습니다.  
무의미한 컬럼이 뒤쪽에 오는 경우에만 재현되며,  
회피방안으로는 order sibling by 절의 컬럼 순서를 바꾸거나, 무의미한 컬럼을 제거하는 경우 정상적으로 동작합니다.
- **재현 방법**

- **재현 절차**

```
drop table t1;
create table t1 ( i1 integer, i2 integer, i3 varchar(10), i4 varchar(10));
drop table t2;
create table t2( i1 varchar(10), i2 varchar(10));
insert into t1 values ( 1,1, 'AAA', 'AAA');
insert into t1 values ( 2,2, 'AAA', 'AAA');
insert into t1 values ( 3,3, 'AAA', 'AAA');
select
(SELECT i1 FROM t2 WHERE i2 = a.i3) AS AA
from
(
select i1, i2 , i3 , i4 from t1
) a
connect by a.i2 = prior a.i1
order siblings by i4, AA;
```

- **수행 결과**

```
iSQL> select
(SELECT i1 FROM t2 WHERE i2 = a.i3) AS AA
from
(
select i1, i2 , i3 , i4 from t1
) a
connect by a.i2 = prior a.i1
order siblings by i4, AA;
[ERR-91015 : Communication failure.]
```

- **예상 결과**

```

iSQL> select
      (SELECT i1 FROM t2 WHERE i2 = a.i3) AS AA
    from
      (
        select i1, i2 , i3 , i4 from t1
      ) a
    connect by a.i2 = prior a.i1
    order siblings by i4, AA;
[ERR-311A4 : Loop in hierarchical query detected.]

```

- **Workaround**

```

select
  (SELECT i1 FROM t2 WHERE i2 = a.i3) AS AA
from
  (
    select i1, i2 , i3 , i4 from t1
  ) a
connect by a.i2 = prior a.i1
order siblings by AA, i4;

```

- **변경사항**

- Performance view
- Property
- Compile Option
- Error Code

**BUG-46197** altilinker가 구동중이지 않은경우에 disconnect을 수행하면, altilinker를 다시 connect 하는 문제가 있어 수정하였습니다.

- **module** : dblink
- **Category** : Functional Error
- **재현 빈도** : Always
- **증상** : altilinker가 구동중이지 않은 경우에 disconnect을 수행하면, altiliner와 세션을 정리하기 위해 altilinker가 다시 connect를 시도합니다. 이로 인해 세션종료가 늦게 처리되는 문제가 있어, altilinker가 종료상태이면 connect을 시도하지 않도록 수정하였습니다.
- **재현 방법**
  - 재현 절차
  - 수행 결과
  - 예상 결과
- **Workaround**
- **변경사항**

- Performance view
- Property
- Compile Option
- Error Code

## BUG-46202 v\$event\_name에 중복된 항목이 있습니다.

- **module** : sm
- **Category** : Message Error
- **재현 빈도** : Always
- **증상** : v\$event\_name에 중복된 항목이 있어, 수정하였습니다.
- **재현 방법**
  - **재현 절차**

```
set vertical on;
select * from v$event_name where name in('latch free: drdb secondary bcb mutex');
```

### ◦ 수행 결과

```
iSQL> select * from v$event_name where name in('latch free: drdb secondary bcb mutex');
EVENT_ID      : 32
NAME          : latch free: drdb secondary bcb mutex
WAIT_CLASS_ID : 3
WAIT_CLASS    : Concurrency

EVENT_ID      : 33
NAME          : latch free: drdb secondary bcb mutex
WAIT_CLASS_ID : 3
WAIT_CLASS    : Concurrency

2 rows selected.
```

### ◦ 예상 결과

```
iSQL> select * from v$event_name where name in('latch free: drdb secondary bcb mutex');
EVENT_ID      : 32
NAME          : latch free: drdb secondary bcb mutex
WAIT_CLASS_ID : 3
WAIT_CLASS    : Concurrency

1 row selected.
```

- **Workaround**
- **변경사항**
  - Performance view
  - Property

- Compile Option
- Error Code

## BUG-46230 iduMemory의 getStatus(), setStatus() 함수 에러메시지 세분화

- **module** : id
- **Category** : Other
- **재현 빈도** : Always
- **증상** : iduMemory의 getStatus(), setStatus() 함수 에러메시지 세분화
- **재현 방법**
  - 재현 절차
  - 수행 결과
  - 예상 결과
- **Workaround**
- **변경사항**
  - Performance view
  - Property
  - Compile Option
  - Error Code

## BUG-46239 DDL 동기화 중 Select 시, Lock 획득에 실패하는 경우 DDL 동기화가 실패할 수 있습니다.

- **module** : rp
- **Category** : Other
- **재현 빈도** : Always
- **증상** : DDL 동기화 중에 DDL 동기화 대상 테이블에 내부적으로 Lock을 잡는데, Lock 획득에 실패하는 경우 DDL 동기화가 실패할 수 있습니다.
- **재현 방법**
  - 재현 절차
  - 수행 결과
  - 예상 결과
- **Workaround**
- **변경사항**
  - Performance view
  - Property
  - Compile Option

- Error Code

## **BUG-46242 DDL 동기화 시 Propagation 옵션이 켜져 있으면 실패해야 합니다.**

- **module** : rp
- **Category** : Other
- **재현 빈도** : Always
- **증상** : Propagation 기능이 켜져있을 때 DDL 동기화를 시도하면 DDL 동기화를 실패하도록 수정 합니다.
- **재현 방법**
  - 재현 절차
  - 수행 결과
  - 예상 결과
- **Workaround**
- **변경사항**
  - Performance view
  - Property
  - Compile Option
  - Error Code

## **BUG-46249 group by 표현식에 있는 컬럼을 참조하는 집합 연산이 존재하지 않으면 결과값 오류가 발생할 수 있습니다.**

- **module** : qp-select-pvo
- **Category** : Functional Error
- **재현 빈도** : Always
- **증상** : group by 표현식에 있는 컬럼(t1.c1)을 참조하는 집합 연산(min(c1))이 존재하지 않으면, case 구문에 포함된 substring 연산이 오동작해서 결과값 오류가 발생할 수 있습니다.  
수정 방법은 case 구문에 포함된 substring 연산이 group by 표현식을 참조하도록 변경합니다.
- **재현 방법**
  - 재현 절차

```

drop table t1;
create table t1 ( c1 varchar(15) );
insert into t1 values ( 'eee' );
insert into t1 values ( 'aaa' );
insert into t1 values ( 'bbb' );
insert into t1 values ( 'ddd' );
insert into t1 values ( 'bbb' );
insert into t1 values ( 'ccc' );
insert into t1 values ( 'aaa' );
insert into t1 values ( 'ccc' );
insert into t1 values ( 'eee' );
insert into t1 values ( 'ddd' );
select /*+ no_plan_cache */ substr(c1, 1, 3)
,count(case when substr(c1, 1, 3) = 'SPBK0' then 1 else 0 end) gg1
from t1 group by substr(c1, 1, 3);
select /*+ no_plan_cache */ substr(c1, 1, 3)
,min(c1)
,count(case when substr(c1, 1, 3) = 'SPBK0' then 1 else 0 end) gg1
from t1 group by substr(c1, 1, 3);

```

#### ○ 수행 결과

```

iSQL> select /*+ no_plan_cache */ substr(c1, 1, 3)
,count(case when substr(c1, 1, 3) = 'SPBK0' then 1 else 0 end) gg1
from t1 group by substr(c1, 1, 3);
SUBSTR(C1,1,3)    GG1
-----
ddd              1
ddd              1
ddd              1
ddd              1
ddd              1
ddd              3
ddd              1
ddd              1
8 rows selected.
iSQL> select /*+ no_plan_cache */ substr(c1, 1, 3)
,min(c1)
,count(case when substr(c1, 1, 3) = 'SPBK0' then 1 else 0 end) gg1
from t1 group by substr(c1, 1, 3);
SUBSTR(C1,1,3)    MIN(C1)          GG1
-----
aaa              aaa              2
bbb              bbb              2
ccc              ccc              2
ddd              ddd              2
eee              eee              2
5 rows selected.

```

#### ○ 예상 결과

```

iSQL> select /*+ no_plan_cache */ substr(c1, 1, 3)
        ,count(case when substr(c1, 1, 3) = 'SPBK0' then 1 else 0 end) gg1
        from t1 group by substr(c1, 1, 3);
select /*+ no_plan_cache */ substr(c1, 1, 3)
SUBSTR(C1,1,3)    GG1
-----
eee                2
aaa                2
bbb                2
ddd                2
ccc                2
5 rows selected.
iSQL> select /*+ no_plan_cache */ substr(c1, 1, 3)
        ,min(c1)
        ,count(case when substr(c1, 1, 3) = 'SPBK0' then 1 else 0 end) gg1
        from t1 group by substr(c1, 1, 3);
SUBSTR(C1,1,3)    MIN(C1)          GG1
-----
eee                eee                2
aaa                aaa                2
bbb                bbb                2
ddd                ddd                2
ccc                ccc                2
5 rows selected.

```

- **Workaround**

use group\_hash hint

- **변경사항**

- Performance view
- Property
- Compile Option
- Error Code

**BUG-46264 디스크테이블에서 window sort의 order by 절에 동일한 컬럼을 중복하여 나열하는 경우, hang이 발생할 수 있습니다.**

- **module** : qp-select-execute
- **Category** : Hang
- **재현 빈도** : Always
- **증상** : 디스크테이블에서 window sort의 order by 절에 동일한 컬럼을 중복하여 나열하는 경우, 중복을 제거하여 hang이 발생하지 않도록 수정하였습니다.
- **재현 방법**
  - 재현 절차



```

DROP TABLE T1;
CREATE TABLE T1 ( I0 integer , I1 integer ) TABLESPACE sys_tbs_disk_data;
INSERT INTO T1 VALUES ( 0, 0 );
INSERT INTO T1 VALUES ( 0, 0 );
SELECT I0,I1, ROW_NUMBER() OVER( ORDER BY I0, I1, I0 ) FROM t1;

```

#### ◦ 수행 결과

Hang

#### ◦ 예상 결과

```

iSQL> SELECT I0,I1, ROW_NUMBER() OVER( ORDER BY I0, I1, I0 ) FROM t1;
I0          I1          ROW_NUMBER()OVER(ORDERBYI0,I1,I0)
-----
0           0           1
0           0           2
2 rows selected.

```

#### • Workaround

```

SELECT I0,I1, ROW_NUMBER() OVER( ORDER BY I0, I1 ) FROM t1;
또는
SELECT /*+ TEMP_TBS_MEMORY */ I0,I1, ROW_NUMBER() OVER( ORDER BY I0, I1, I0 ) FROM t1;

```

#### • 변경사항

- Performance view
- Property
- Compile Option
- Error Code

## BUG-46265 Create Disk Temp Table 에서 Key column list가 순환 할 수도 있는지 검증합니다.

- **module** : sm-disk-resource
- **Category** : Hang
- **재현 빈도** : Always
- **증상** : Create Disk Temp Table 에서 Key column list에 중복이 있는경우, 무시하도록 수정하였습니다. 이로 인해 HANG이 발생하던 문제가 수정되었습니다.
- **재현 방법**
  - 재현 절차

```

drop table t1;
create table t1 ( I0 integer , I1 integer ) tablespace sys_tbs_disk_data;
insert into t1 values( 0, 0 );
insert into t1 values( 0, 0 );

SELECT I0,I1, ROW_NUMBER() OVER( ORDER BY I0, I1, I0 ) FROM t1;

```

#### ◦ 수행 결과

Hang

#### ◦ 예상 결과

```

iSQL> SELECT I0,I1, ROW_NUMBER() OVER( ORDER BY I0, I1 ) FROM t1;
I0          I1          ROW_NUMBER()OVER(ORDERBYI0,I1)
-----
0           0           1
0           0           2
2 rows selected.

```

#### • Workaround

```
SELECT I0,I1, ROW_NUMBER() OVER( ORDER BY I0, I1 ) FROM t1;
```

#### • 변경사항

- Performance view
- Property
- Compile Option
- Error Code

## BUG-46274 Partition swap 후 update 수행 시 column not found 에러가 발생합니다.

- **module** : qx
- **Category** : Functional Error
- **재현 빈도** : Always
- **증상** : Partition swap 후 update 수행 시 column not found 에러  
수정
- **재현 방법**
  - 재현 절차

```

CREATE TABLE RANGE_T1( I1 VARCHAR(10) PRIMARY KEY, I2 VARCHAR(10) )
PARTITION BY RANGE(I1)
(
    PARTITION P1 VALUES LESS THAN (300),
    PARTITION P2 VALUES LESS THAN (600),
    PARTITION PD VALUES DEFAULT
);
CREATE TABLE RANGE_T1_REBUILD( I1 VARCHAR(10) PRIMARY KEY, I2 VARCHAR(10) )
PARTITION BY RANGE(I1)
(
    PARTITION P1 VALUES LESS THAN (300),
    PARTITION P2 VALUES LESS THAN (600),
    PARTITION PD VALUES DEFAULT
);

update range_t1 set i2 =1;
ALTER TABLE RANGE_T1 REPLACE RANGE_T1_REBUILD PARTITION P1;
update range_t1 set i2 =1;

```

#### ◦ 수행 결과

```

iSQL> update range_t1 set i2 =1;
[ERR-31318 : Unexpected errors may have occurred: qmo::makePartUpdateColumnList: Column

```

#### ◦ 예상 결과

```

iSQL> update range_t1 set i2 =1;
No rows updated.

```

#### • Workaround

#### • 변경사항

- Performance view
- Property
- Compile Option
- Error Code

## BUG-46279 Disk temp 사용 시 grouping 데이터의 정렬이 subquery를 참조하는 경우 결과 값 오류가 발생합니다.

- **module** : qp-select-pvo
- **Category** : Functional Error
- **재현 빈도** : Always
- **증상** : Disk temp 사용 시 grouping 데이터의 정렬이 subquery를 참조하는 경우 결과 값 오류가 발생합니다.
- **재현 방법**

## ○ 재현 절차

```
DROP TABLE T1;
DROP TABLE T3;
CREATE TABLE T1 ( game_id int ) TABLESPACE sys_tbs_disk_data;
CREATE TABLE T3 ( game_nm int, game_id int ) TABLESPACE sys_tbs_disk_data;
INSERT INTO T1 SELECT LEVEL FROM DUAL CONNECT BY LEVEL <= 5;
INSERT INTO T3 SELECT LEVEL, LEVEL FROM DUAL CONNECT BY LEVEL <= 5;
SELECT (SELECT GAME_NM FROM T3 WHERE GAME_ID = a.GAME_ID) AS GAME_NM
FROM T1 a
GROUP BY a.GAME_Id
ORDER BY a.GAME_ID,
        GAME_NM
;
SELECT /*+ TEMP_TBS_MEMORY */ (SELECT GAME_NM FROM T3 WHERE GAME_ID = A.GAME_ID) AS GAM
FROM T1 A
GROUP BY A.GAME_ID
ORDER BY A.GAME_ID,
        GAME_NM
;
```

## ○ 수행 결과

```
iSQL> SELECT (SELECT GAME_NM FROM T3 WHERE GAME_ID = a.GAME_ID) AS GAME_NM
        FROM T1 a
        GROUP BY a.GAME_Id
        ORDER BY a.GAME_ID,
                GAME_NM
;
GAME_NM
-----
3
5
4
2
5 rows selected.
iSQL> SELECT /*+ TEMP_TBS_MEMORY */ (SELECT GAME_NM FROM T3 WHERE GAME_ID = A.GAME_ID)
        FROM T1 A
        GROUP BY A.GAME_ID
        ORDER BY A.GAME_ID,
                GAME_NM
;
GAME_NM
-----
1
2
3
4
5
5 rows selected.
```

- 예상 결과

```
isql> SELECT (SELECT GAME_NM FROM T3 WHERE GAME_ID = a.GAME_ID) AS GAME_NM
        FROM T1 a
        GROUP BY a.GAME_Id
        ORDER BY a.GAME_ID,
                GAME_NM
        ;
```

GAME\_NM

-----

1

2

3

4

5

5 rows selected.

```
isql> SELECT /*+ TEMP_TBS_MEMORY */ (SELECT GAME_NM FROM T3 WHERE GAME_ID = A.GAME_ID) .
        FROM T1 A
        GROUP BY A.GAME_ID
        ORDER BY A.GAME_ID,
                GAME_NM
        ;
```

GAME\_NM

-----

1

2

3

4

5

5 rows selected.

- Workaround

```
SELECT /*+ TEMP_TBS_MEMORY */ (SELECT GAME_NM FROM T3 WHERE GAME_ID = A.GAME_ID) AS GAME_NM
FROM T1 A
GROUP BY A.GAME_ID
ORDER BY A.GAME_ID,
        GAME_NM
;
```

- 변경사항

- Performance view
- Property
- Compile Option
- Error Code

**BUG-46286 insert ~ select 구문에서 테이블 이름 없이 parallel 힌트를 사용하면 에러가 발생합니다**

- **module** : qp-dml-pvo
- **Category** : Functional Error
- **재현 빈도** : Always
- **증상** : insert ~ select 구문에서 테이블 이름 없이 parallel 힌트를 사용하면 힌트를 무시하도록 변경합니다.
- **재현 방법**

- **재현 절차**

```
DROP TABLE T1;
CREATE TABLE T1 ( c1 int );
INSERT /*+ parallel(8) */ into T1 SELECT * FROM T1;
```

- **수행 결과**

```
iSQL> INSERT /*+ parallel(8) */ into T1 SELECT * FROM T1;
[ERR-31455 : Failed to work because an internal exception occurred from an OS.[Contact .
```

- **예상 결과**

```
iSQL> INSERT /*+ parallel(8) */ into T1 SELECT * FROM T1;
No rows inserted.
```

- **Workaround**

```
INSERT /*+ parallel(T1, 8) */ into T1 SELECT * FROM T1;
```

- **변경사항**

- Performance view
- Property
- Compile Option
- Error Code

## Changes

### Version Info

altibase version	database binary version	meta version	cm protocol version	replication protocol version	sharding version
7.1.0.1.4	6.5.1	8.6.1	7.1.6	7.4.3	2.1.0

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 버전이 8.5.1에서 8.6.1로 변경되었다. 7.1.0.1.3 이하에서 7.1.0.1.4로 패치하면, 자동으로 메타 업그레이드가 수행된다.

메타테이블 [SYS\\_REPL\\_HOSTS\\_](#)에 [CONN\\_TYPE](#), [IB\\_LATENCY](#) 컬럼이 추가되었다.

패치를 롤백하려는 경우, [메타다운그레이드](#)를 참고한다.

### CM protocol Version

통신 프로토콜 버전은 변경되지 않았다.

### Replication protocol Version

이중화 프로토콜 버전이 7.4.3에서 7.4.4로 변경되었지만, 하위 호환성을 보장한다.

### Sharding Version

샤딩 버전이 2.0.0에서 2.1.0으로 변경되었다. 샤딩에서 사용하는 메타 테이블 및 패키지(프로시저)가 변경되었으므로, 패치 후 샤딩 재구성해야 한다.

알티베이스 샤딩 프로토콜 및 메타는 상위, 하위 호환성을 보장하지 않는다. 즉, 샤딩 버전이 다른 경우, 재구성해야 한다. 샤딩 설정은 \*[Altibase Sharding 설치와 설정](#)\*을 참고한다.

## 프로퍼티

### 추가된 프로퍼티

- [REPLICATION\\_IB\\_PORT\\_NO](#)
- [REPLICATION\\_IB\\_LATENCY](#)
- REPLICATION\_DDL\_SYNC
- REPLICATION\_DDL\_SYNC\_TIMEOUT

## 성능 뷰

### 변경된 성능 뷰

- [V\\$SQL\\_PLAN\\_CACHE\\_SQLTEXT](#)
- [V\\$SQL\\_PLAN\\_CACHE\\_PCO](#)
- V\$SESSION