

Altibase 7.1.0.8.2 Patch Notes

Table of Contents

- [Fixed Bugs](#)

- [BUG-49607 객체의 변경으로 PACKAGE SPEC 또는 TYPESET이 INVALID 상태로 변경되면, 이를 참조하는 객체도 INVALID 상태로 변경합니다.](#)
- [BUG-49851 SQLDriverConnect 함수에서 연결 속성 TIMEOUT의 값을 0으로 설정하면 Connection Timeout Error 에러가 발생할 수 있습니다.](#)
- [BUG-49913 Lateral view로 사용된 파티션드 테이블이 병렬 질의로 처리되어 발생하는 안정성 문제를 개선합니다.](#)
- [BUG-49923 재귀 호출 또는 참조 깊이의 수가 많을 때 PSM을 컴파일 시 발생하는 예외 처리를 개선합니다.](#)
- [BUG-49925 파티션드 테이블을 기반으로 생성한 뷰를 multiple update\(또는 delete\)에서 사용하고 참조 테이블에 트리거가 있을 때 발견된 안정성 문제를 개선합니다.](#)
- [BUG-49930 뷰를 갱신하는 multiple update 문에서 트리거에서 추가한 변경 컬럼을 찾지 못해 ERR-31058 : Column not found 에러가 발생합니다.](#)
- [BUG-49931 NEW ROW로 값을 변경하는 BEFORE UPDATE 트리거의 참조 테이블을 기반으로 생성한 뷰를 갱신할 때 발견된 정합성 오류를 개선합니다.](#)
- [BUG-49942 디스크 트랜잭션의 롤백과 핑퐁 체크포인트의 동시성 문제로 Altibase 서버 구동이 실패할 수 있습니다.](#)
- [BUG-49987 FROM 절에 인라인 뷰, GROUP BY 절에 문자 함수, SELECT 절과 분석 함수 ORDER BY 절에 GROUP BY 절의 키를 사용할 때 발생하는 안정성 문제를 개선합니다.](#)
- [BUG-49994 윈도우 함수의 window order clause 절에서 NULLS 및 window frame clause 절 사용 시 정합성 오류가 발생할 수 있습니다.](#)
- [BUG-49998 multiple update 구문에서 SET절에 같은 컬럼을 두 번 이상 사용해도 에러가 발생하지 않습니다.](#)
- [BUG-50002 RANGE USING HASH execute 시점에 shred tmlate 사용하는 부분 private tmlate로 변경](#)
- [BUG-50004 aku.conf.sample 파일 내 프로퍼티 값을 기본값으로 설정합니다.](#)
- [BUG-50017 APRE에서 CREATE QUEUE 구문에 DELETE 절을 사용하면 컴파일 시 구문 분석 오류가 발생합니다.](#)
- [BUG-50022 multiple update의 대상 테이블에 TIMESTAMP 타입 컬럼이 있을 때 발생하는 안정성 문제를 개선합니다.](#)
- [BUG-50032 oaUtility 유틸리티에 force 옵션을 추가합니다.](#)
- [BUG-50038 jdbcAdapter 및 oraAdapter 구동에서 예외 상황 발생 시 에러 메시지를 보완합니다.](#)
- [BUG-50048 aku에 세션 접근 제어 기능을 추가합니다.](#)
- [BUG-50050 뷰/인라인 뷰에 중복되는 조건식을 제거하는 기능을 추가합니다.](#)
- [BUG-50054 ALTER REPLICATION ~ SYNC가 30초 이상 지속하면 HeartBeat 스레드에 의해 이중화 SYNC가 실패합니다.](#)

- [Changes](#)

- [Version Info](#)
- [호환성](#)

- 프로퍼티
- 성능 분

Fixed Bugs

BUG-49607 객체의 변경으로 INVALID 상태의 PACKAGE SPEC 또는 TYPESET을 참조하는 객체를 수행할 때 발생하는 안정성 문제를 개선합니다.

module

qp-psm-trigger-pvo

Category

Fatal

재현 빈도

Always

설명

객체의 변경으로 PACKAGE SPEC 또는 TYPESET이 INVALID 상태로 변경되면, 이를 참조하는 객체를 실행할 때 Altibase 서버가 비정상 종료하는 버그를 수정합니다.

재현 방법

- 재현 절차

```
CREATE TABLE T1 ( I1 INTEGER, I2 INTEGER ) ;

CREATE OR REPLACE TYPESET TYPE1
AS
TYPE ARRI IS TABLE OF INTEGER INDEX BY INTEGER;
TYPE REC IS RECORD ( I1 T1.I1%TYPE);
END;
/

CREATE OR REPLACE PROCEDURE PROC1
AS VARRI TYPE1.ARRI;
R1 INTEGER; BEGIN VARRI(1) := 1;
SELECT VARRI(1) INTO R1 FROM DUAL ;
PRINTLN( R1 );
END;
/

ALTER TABLE T1 RENAME COLUMN I2 TO TYPE;

EXEC PROC1;
```

- 수행 결과

[ERR-91015 : Communication failure.] 발생하며 Altibase 서버 비정상 종료

- 예상 결과

1

Execute success.

Workaround

변경된 객체를 참조하는 객체를 다시 컴파일하거나 CREATE OR REPLACE를 수행합니다.

변경사항

- Performance view
- Property
- Compile Option
- Error Code

BUG-49851 SQLDriverConnect 함수에서 연결 속성 TIMEOUT의 값을 0으로 설정하면 Connection Timeout Error 에러가 발생할 수 있습니다.

module

mm-cli

Category

Functional Error

재현 빈도

Always

설명

Altibase CLI의 SQLDriverConnect 함수에서 연결 속성 TIMEOUT의 값을 0으로 설정하면 Connection Timeout Error 에러가 발생하는 현상을 수정합니다. 이 버그는 OS와 무관하게 노출되어 있으나, 버그 현상은 Windows 환경에서 Altibase 서버로 원격 접속할 때 항상 발생합니다.

이 버그를 적용하려면 Altibase CLI 라이브러리를 패치해야 합니다.

- Linux/Unix 클라이언트 : libodbccli.a, libodbccli_sl.so
- Windows 클라이언트 : altiodbc.dll

재현 방법

- 재현 절차

window에서 아래와 같이 접속 시 재현됩니다.

```
rc = SQLDriverConnect(dbc, NULL, (SQLCHAR
*)"Server=127.0.0.1;User=SYS;Password=MANAGER;TIMEOUT=0", SQL_NTS, NULL, 0,
NULL, SQL_DRIVER_NOPROMPT);
```

- 수행 결과

Connection Timeout Error

- 예상 결과

Connection success

Workaround

없음

변경사항

- Performance view
- Property
- Compile Option
- Error Code

BUG-49913 Lateral view로 사용된 파티션드 테이블이 병렬 질의로 처리되어 발생하는 안정성 문제를 개선합니다.

module

qp

Category

Fatal

재현 빈도

Always

설명

Lateral view로 사용된 파티션드 테이블을 병렬 질의로 처리되어 Altibase 서버가 비정상 종료합니다. Lateral view는 병렬 질의를 지원하는 대상이 아니므로 병렬 질의로 처리하지 않도록 수정합니다.

이 버그는 아래와 같은 조건에서 발생합니다.

- 파티션드 테이블에 PARALLEL 절을 지정하거나 PARALLEL 힌트를 사용
- 파티션드 테이블이 사용된 인라인 뷰를 Lateral View로 정의(LATERAL 또는 APPLY 키워드 사용)

재현 방법

- 재현 절차

```

CREATE TABLE T2( I1 INTEGER, I2 INTEGER )
PARTITION BY HASH(I1)
( PARTITION P1,
  PARTITION P2
) TABLESPACE SYS_TBS_MEM_DATA;

ALTER TABLE T2 PARALLEL 4;

CREATE TABLE T1 (I1 INTEGER, I2 INTEGER);
INSERT INTO T2 VALUES( 4, 4 );

SELECT * FROM T1 OUTER APPLY ( SELECT SUM(T1.I1) AS I1 FROM T2 INTERSECT
SELECT COUNT(T1.I2) AS I1 FROM T2 );

```

- 수행 결과

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

- 예상 결과

```

I1          I2          I1
-----
No rows selected.

```

Workaround

/*+ NOPARALLEL */ 힌트를 사용하여 버그를 회피할 수 있습니다.

```

SELECT * FROM T1 OUTER APPLY ( SELECT /*+ NOPARALLEL */ SUM(T1.I1) AS I1 FROM T2
INTERSECT SELECT /*+ NOPARALLEL */ COUNT(T1.I2) AS I1 FROM T2 );

```

변경사항

- Performance view
- Property
- Compile Option
- Error Code

BUG-49923 재귀 호출 또는 참조 깊이의 수가 많을 때 PSM을 컴파일 시 발생하는 예외 처리를 개선합니다.

module

```
qp-psm-trigger-execute
```

Category

```
Fatal
```

재현 빈도

Always

설명

PSM을 컴파일할 때 재귀 호출 또는 참조 깊이의 수가 많으면 Altibase 서버가 비정상 종료할 수 있는 버그를 개선합니다. 재귀 호출 또는 참조 깊이의 수가 많아지는 것을 방지하기 위해 Altibase 서버 프로퍼티를 추가합니다.

Altibase 7.1 에는 이 버그의 변경 사항이 비활성화되어 있습니다. 그래서 Altibase 7.1.0.8.2 이상에서 이 버그를 적용하려면 PSM_MAX_DDL_REFERENCE_DEPTH 프로퍼티의 값을 변경해야 합니다. 이 프로퍼티는 ALTER SYSTEM 으로 변경할 수 있습니다. 변경 값을 Altibase 서버에 영구적으로 적용하려면 altibase.properties 파일에 PSM_MAX_DDL_REFERENCE_DEPTH 프로퍼티를 추가해야 합니다.

PSM을 컴파일할 때 재귀 호출 또는 참조 깊이의 수가 PSM_MAX_DDL_REFERENCE_DEPTH 값을 초과하면 아래와 같은 에러 메시지가 반환됩니다.

에러 메시지 예시

```
[ERR-31459 : The attempt to recompile the object was aborted.
```

```
The number of recursive calls or references depth exceeded 128 in the PSM.  
0003 : VAR1 PKG2_SYM.T1_REC;  
      ^  ^
```

재현 방법

- 재현 절차

```
CREATE TABLE T1 (C1 INTEGER);  
CREATE TABLE T2 (C1 INTEGER);  
CREATE OR REPLACE PACKAGE PKG1 AS  
TYPE T1_REC IS RECORD(C1 T1.C1%TYPE);  
VAR1 NUMBER;  
PROCEDURE PROC1;  
END;  
/  
CREATE OR REPLACE PACKAGE BODY PKG1 AS  
PROCEDURE PROC1 AS  
BEGIN  
    PRINTLN('PKG1');  
END;  
BEGIN  
    VAR1 := 1;  
END;  
/  
CREATE SYNONYM PKG1_SYM FOR SYS.PKG1;  
CREATE OR REPLACE PACKAGE PKG2 AS  
TYPE T1_REC IS RECORD(C1 T2.C1%TYPE);  
VAR1 PKG1_SYM.T1_REC;  
PROCEDURE PROC1;  
END;  
/
```



```

CREATE OR REPLACE PACKAGE BODY PKG2 AS
PROCEDURE PROC1 AS
BEGIN
    PRINTLN('PKG2');
    PKG1_SYM.PROC1;
END;
BEGIN
    VAR1.C1 := 1;
END;
/
CREATE SYNONYM PKG2_SYM FOR SYS.PKG2;
CREATE OR REPLACE PACKAGE PKG3 AS
TYPE T1_REC IS RECORD(C1 T2.C1%TYPE);
VAR1 PKG2_SYM.T1_REC;
PROCEDURE PROC1;
END;
/
CREATE OR REPLACE PACKAGE BODY PKG3 AS
PROCEDURE PROC1 AS
BEGIN
    PRINTLN('PKG3');
    PKG2_SYM.PROC1;
END;
BEGIN
    VAR1.C1 := 1;
END;
/

DROP SYNONYM PKG1_SYM;
CREATE SYNONYM PKG1_SYM FOR SYS.PKG3;

EXEC PKG2.PROC1;

```

- 수행 결과

```

[ERR-31455 : Failed to work because an internal exception occurred from an
OS.[Contact Altibase's Support Center]]

```

- 예상 결과

```

[ERR-31459 : The attempt to recompile the object was aborted.
The number of recursive calls or references depth exceeded 128 in the PSM.
0003 : VAR1 PKG2_SYM.T1_REC;
      ^ ^
]

```

Workaround

없음

변경사항

- Performance view
- Property
 - PSM_MAX_DDL_REFERENCE_DEPTH
 - 데이터 타입
Unsigned Integer
 - 기본값
4294967295 (기능 비활성화)
 - 속성
변경 가능, 단일 값
 - 값의 범위
[64, 232-1]
 - 설명
PSM을 컴파일할 때 재귀 호출 또는 참조 깊이의 수를 제한합니다. 재귀 호출 또는 참조 깊이의 수가 설정 값을 초과하면 에러가 발생합니다.
- Compile Option
- Error Code

BUG-49925 파티션드 테이블을 기반으로 생성한 뷰를 multiple update(또는 delete)에서 사용하고 참조 테이블에 트리거가 있을 때 발견된 안정성 문제를 개선합니다.

module

qp-dm1-pvo

Category

Fatal

재현 빈도

Always

설명

파티션드 테이블을 기반으로 생성한 뷰를 multiple update(또는 delete)에서 사용하고 참조 테이블에 트리거가 있을 때 Altibase 서버가 비정상 종료하는 문제를 개선합니다.

재현 방법

- 재현 절차

```
CREATE TABLE T1( I1 INTEGER, I2 INTEGER, I3 INTEGER, I4 INTEGER ) PARTITION
BY HASH(I4) ( PARTITION P1, PARTITION P2 ) ;
```

```

CREATE VIEW V1 AS SELECT I1, I2 FROM T1 WHERE I1 < 5;
CREATE SYNONYM V2 FOR V1;

CREATE OR REPLACE TRIGGER TRIG1
BEFORE UPDATE OF I1 ON T1
REFERENCING NEW AS NEW_ROW
FOR EACH ROW
BEGIN
    NEW_ROW.I1 := NEW_ROW.I1 + 1;
    NEW_ROW.I2 := NEW_ROW.I2 + 1;
END;
/

DELETE FROM V1,V2 USING V1 RIGHT OUTER JOIN V2 ON V1.I1 = V2.I1 WHERE V1.I1 =
5 OR V2.I1 = 6;

```

- 수행 결과

[ERR-91015 : Communication failure.]

- 예상 결과

No rows deleted.

Workaround

없음

변경사항

- Performance view
- Property
- Compile Option
- Error Code

BUG-49930 뷰를 갱신하는 multiple update 문에서 트리거에서 추가한 변경 컬럼을 찾지 못해 ERR-31058 : Column not found 에러가 발생합니다.

module

qp-dml-pvo

Category

Functional Error

재현 빈도

Always

설명

뷰를 갱신하는 multiple update 문에서 트리거에서 추가한 변경 컬럼을 찾지 못해 ERR-31058 : Column not found 에러가 발생하는 버그를 수정합니다.

재현 방법

- 재현 절차

```
CREATE TABLE T1( I1 INTEGER, I2 INTEGER, I3 INTEGER, I4 INTEGER ) PARTITION
BY HASH(I4) ( PARTITION P1, PARTITION P2 ) TABLESPACE SYS_TBS_DISK_DATA;

CREATE VIEW V1 AS SELECT I1, I2 FROM T1 WHERE I1 < 5;
CREATE SYNONYM V2 FOR V1;

CREATE OR REPLACE TRIGGER TRIG1
BEFORE UPDATE OF I1 ON T1
REFERENCING NEW AS NEW_ROW
FOR EACH ROW
BEGIN
    NEW_ROW.I1 := NEW_ROW.I1 + 1;
    NEW_ROW.I2 := NEW_ROW.I2 + 1;
END;
/

UPDATE V1 RIGHT OUTER JOIN V2 ON V1.I1 = V2.I1 SET V1.I1 = V1.I1 + 1 WHERE
V1.I1 = 5 OR V2.I1 = 6;
```

- 수행 결과

```
UPDATE V1 RIGHT OUTER JOIN V2 ON V1.I1 = V2.I1 SET V1.I1 = V1.I1 + 1 WHERE
V1.I1 = 5 OR V2.I1 = 6;
[ERR-31058 : Column not found
0001 : UPDATE V1 RIGHT OUTER JOIN V2 ON V1.I1 = V2.I1 SET V1.I1 = V1.I1 + 1
WHERE V1.I1 = 5 OR V2.I1 = 6
      ^ ^
]
```

- 예상 결과

No rows updated.

Workaround

없음

변경사항

- Performance view
- Property
- Compile Option
- Error Code

BUG-49931 NEW ROW로 값을 변경하는 BEFORE UPDATE 트리거의 참조 테이블을 기반으로 생성한 뷰를 갱신할 때 발견된 정합성 오류를 개선합니다.

module

qp-dml-pvo

Category

Functional Error

재현 빈도

Always

설명

NEW ROW로 값을 변경하는 BEFORE UPDATE 트리거의 참조 테이블을 기반으로 생성한 뷰를 갱신할 때 발견된 결과 오류 버그를 수정합니다.

이 버그의 발생 조건은 아래와 같습니다.

1. NEW ROW로 값을 변경하는 BEFORE UPDATE 트리거
2. 이 트리거의 참조 테이블을 기반으로 생성한 뷰
3. 해당 뷰를 갱신하는 UPDATE 문

재현 방법

- 재현 절차

```
CREATE TABLE T1( I1 INTEGER, I2 INTEGER);
CREATE VIEW V1 AS SELECT I1, I2 FROM T1 WHERE I1 < 5;
CREATE OR REPLACE TRIGGER TRIG1
BEFORE UPDATE OF I1 ON T1
REFERENCING NEW AS NEW_ROW
FOR EACH ROW
BEGIN
    NEW_ROW.I1 := NEW_ROW.I1 + 1;
    NEW_ROW.I2 := NEW_ROW.I2 + 1;
END;
/
INSERT INTO V1 VALUES(1,1);
UPDATE V1 SET I1 = I1;
SELECT * FROM V1;
```

- 수행 결과

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

- 예상 결과

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

Workaround

없음

변경사항

- Performance view
- Property
- Compile Option
- Error Code

BUG-49942 디스크 트랜잭션의 롤백과 핑퐁 체크포인트의 동시성 문제로 Altibase 서버 구동이 실패할 수 있습니다.

module

sm-mem-recovery

Category

Fatal

재현 빈도

Always

설명

디스크 트랜잭션의 롤백과 핑퐁 체크포인트(ping-pong checkpoint)의 동시성 문제를 수정합니다. Altibase 서버 운용 중에 아래와 같은 상황이 발생하면 로그 앵커(Log Anchor) 파일에 안정적인(stable) 체크포인트 이미지 파일 정보가 변경되지 않습니다. 이에 따라 Altibase 서버 구동이 실패하거나 Altibase 서버 운용 중 예상할 수 없는 문제가 발생할 수 있습니다.

- 디스크 트랜잭션의 롤백과 메모리 DB의 핑퐁 체크포인트가 동시 수행
- 이후 Altibase 서버 중지

재현 방법

- 재현 절차
- 수행 결과
- 예상 결과

Workaround

없음

변경사항

- Performance view
- Property
- Compile Option
- Error Code

BUG-49987 FROM 절에 인라인 뷰, GROUP BY 절에 문자 함수, SELECT 절과 분석 함수 ORDER BY 절에 GROUP BY 절의 키를 사용할 때 발생하는 안정성 문제를 개선합니다.

module

qp-select

Category

Fatal

재현 빈도

Always

설명

FROM 절에 인라인 뷰, GROUP BY 절에 문자 함수, SELECT 절과 분석 함수 ORDER BY 절에 GROUP BY 절의 키를 사용할 때 Altibase 서버가 비정상 종료하는 현상을 수정합니다.

재현 방법

- 재현 절차

```
CREATE TABLE T1 ( YYYYMM VARCHAR(6) );
INSERT INTO T1 VALUES ('201901');
CREATE TABLE T2 ( YYYYMM VARCHAR(6) );
INSERT INTO T2 VALUES ('202001');
SELECT SUBSTR(B1.YYYYMM,1,4) || 'A' AS ACT_YYYY
      , ROW_NUMBER() OVER( ORDER BY SUBSTR(B1.YYYYMM,1,4)) AS PK_COL
FROM T1 B2
      , (SELECT A2.YYYYMM AS YYYYMM FROM T2 A2) B1
WHERE B2.YYYYMM (+)= B1.YYYYMM
GROUP BY SUBSTR(B1.YYYYMM,1,4);
```

- 수행 결과

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

- 예상 결과

```
ACT_YYYY  PK_COL
-----
2020A      1
1 row selected.
```

Workaround

없음

변경사항

- Performance view
- Property
- Compile Option
- Error Code

BUG-49994 윈도우 함수의 window_order_clause 절에서 NULLS 및 window_frame_clause 절 사용 시 정합성 오류가 발생할 수 있습니다.

module

qp-select

Category

Functional Error

재현 빈도

Always

설명

윈도우 함수의 window_order_clause 절에서 NULLS 및 window_frame_clause 절 사용 시 결과 오류가 발생하는 문제를 개선합니다.

재현 방법

- 재현 절차

```
CREATE TABLE T1 ( I1 INTEGER, I2 INTEGER, I3 INTEGER );
INSERT INTO T1 VALUES( 1, 1, 1);
INSERT INTO T1 VALUES( 1, 1, 1);
INSERT INTO T1 VALUES( 2, 1, 1);
INSERT INTO T1 VALUES( 2, 2, 1);
INSERT INTO T1 VALUES( 2, 1, 2);
```



```

INSERT INTO T1 VALUES( 2, 1, 1);
INSERT INTO T1 VALUES( 2, 3, 1);
INSERT INTO T1 VALUES( 3, 3, 1);
INSERT INTO T1 VALUES( 1, 1, 3);
INSERT INTO T1 VALUES( 1, 2, 3);
INSERT INTO T1 VALUES(NULL, 2, 3);
INSERT INTO T1 VALUES(NULL, NULL, 3);
INSERT INTO T1 VALUES(NULL, NULL, NULL);

SELECT COUNT(I3) OVER (PARTITION BY I1
                      ORDER BY I2 NULLS LAST ROWS UNBOUNDED PRECEDING) C1
, COUNT(I3) OVER (PARTITION BY I1
                  ORDER BY I2 NULLS LAST RANGE 1 PRECEDING) C7
FROM T1
GROUP BY I1,I2,I3
ORDER BY C1,C7;

```

- 수행 결과

c1	c7
1	1
1	1
2	1
2	1
1	2
2	2
1	2
2	2
4	2
3	3
3	3

11 rows selected.

- 예상 결과

c1	c7
1	1
1	1
1	2
1	2
2	1
2	1
2	2
2	2
3	3
3	3
4	2

11 rows selected.

Workaround

```
SELECT *
FROM (SELECT COUNT(I3) OVER (PARTITION BY I1
                             ORDER BY I2 NULLS LAST ROWS UNBOUNDED PRECEDING) C1
      , COUNT(I3) OVER (PARTITION BY I1
                             ORDER BY I2 NULLS LAST RANGE 1 PRECEDING) C7
FROM T1
GROUP BY I1,I2,I3 )
ORDER BY C1,C7;
```

변경사항

- Performance view
- Property
- Compile Option
- Error Code

BUG-49998 multiple update 구문에서 SET절에 같은 칼럼을 두 번 이상 사용해도 에러가 발생하지 않습니다.

module

qp-dml-pvo

Category

Functional Error

재현 빈도

Always

설명

multiple update 구문에서 SET 절에 같은 칼럼을 두 번 이상 사용해도 에러가 발생하지 않는 버그를 수정합니다. SQL Reference 매뉴얼 UPDATE 문 설명에 "SET 절에 같은 칼럼을 두 번 이상 사용할 수 없다." 는 주의 사항이 명시되어 있습니다.

재현 방법

- 재현 절차

```
CREATE SYNONYM V2 FOR V1;
CREATE TABLE T1( I1 INTEGER, I2 INTEGER, I3 INTEGER, I4 INTEGER );
CREATE VIEW V1 AS SELECT I1, I2 FROM T1 WHERE I1 < 5;
INSERT INTO T1 SELECT ROWNUM, ROWNUM + 1, ROWNUM, ROWNUM + 1 FROM DUAL
CONNECT BY LEVEL <= 20;

UPDATE /*+ no_plan_cache */ V1 RIGHT OUTER JOIN V2 ON V1.I1 = V2.I1 SET V1.I1
= V1.I1 - 1, V1.I1 = V1.I1 + 1 WHERE V1.I1 = 3 OR V2.I1 = 4;

SELECT * FROM T1 WHERE I1 <> I3 OR I2 <> I4;
```

- 수행 결과

2 rows updated.

- 예상 결과

[ERR-31072 : Duplicate columns specified in a SET clause.]

Workaround

없음

변경사항

- Performance view
- Property
- Compile Option
- Error Code

BUG-50002 RANGE USING HASH execute 시점에 shred tmlate 사용하는 부분 private tmlate로 변경

module

qp

Category

Fatal

재현 빈도

Always

설명

해시를 사용한 범위 파티션드 테이블이 USE_FULL_NL 조인 방법으로 수행될 때 Altibase 서버가 비정상 종료하는 버그를 수정합니다.

재현 방법

- 재현 절차

```
CREATE TABLE T1 ( I1 INTEGER, I2 VARCHAR(256) )
PARTITION BY RANGE_USING_HASH (I1)
( PARTITION P1 VALUES LESS THAN (400),
  PARTITION P3 VALUES DEFAULT
);

INSERT INTO T1 VALUES (1, 9);

SELECT /*+ NO_PLAN_CACHE USE_FULL_NL(V1, V2) */
FROM T1 V1
      , T1 V2
WHERE V1.I2 = V2.I1;
```

• 수행 결과

[ERR-91015 : Communication failure.] 발생하며 Altibase 서버 비정상 종료

• 예상 결과

I1	I2	I1	I2

No rows selected.			

Workaround

없음

변경사항

- Performance view
- Property
- Compile Option
- Error Code

BUG-50004 aku.conf.sample 파일 내 프로퍼티 값을 기본값으로 설정합니다.

module

aku

Category

Maintainability

재현 빈도

Always

설명

aku.conf.sample 파일에서 AKU_QUERY_TIMEOUT, SYNC_PARALLEL_COUNT 프로퍼티의 값을 기본값으로 변경합니다.

재현 방법

- 재현 절차
- 수행 결과
- 예상 결과

Workaround

없음

변경사항

- Performance view
- Property
- Compile Option
- Error Code

BUG-50017 APRE에서 CREATE QUEUE 구문에 DELETE 절을 사용하면 컴파일 시 구문 분석 오류가 발생합니다.

module

u1-odbc

Category

Functional Error

재현 빈도

Always

설명

APRE에서 CREATE QUEUE 구문에 DELETE 절을 사용하면 컴파일 시 아래와 같이 구문 분석 오류가 발생하는 버그를 수정합니다.

```
[ERR-000H : parse error, unexpected TR_DELETE, expecting TS_SEMICOLON]
```

이 버그를 적용하려면 APRE를 패치해야 합니다.

재현 방법

- 재현 절차
- 수행 결과
- 예상 결과

Workaround

없음

변경사항

- Performance view
- Property
- Compile Option
- Error Code

BUG-50022 multiple update의 대상 테이블에 TIMESTAMP 타입 컬럼이 있을 때 발생하는 안정성 문제를 개선합니다.

module

qp-dml-pvo

Category

Fatal

재현 빈도

Always

설명

multiple update의 대상 테이블에 TIMESTAMP 타입 컬럼이 있을 때 Altibase 서버가 비정상 종료하는 문제를 개선합니다.

재현 방법

- 재현 절차

```
CREATE TABLE T1 ( I1 INTEGER, I2 INTEGER ) ;
CREATE TABLE T2 ( I1 VARCHAR(100) );
ALTER TABLE T1 ADD COLUMN ( I3 TIMESTAMP VARIABLE );
UPDATE T1 LEFT OUTER JOIN T2 ON T1.I1 = T2.I1 SET T1.I1 = 100, T1.I2 = 100
WHERE T2.I1 = 6;
```

- 수행 결과

[ERR-91015 : Communication failure.] 발생하며 Altibase 서버 비정상 종료

- 예상 결과

No rows updated.

Workaround

없음

변경사항

- Performance view
- Property
- Compile Option
- Error Code

BUG-50032 oaUtility 유틸리티에 force 옵션을 추가합니다.

module

rp-oraAdapter

Category

Functionality

재현 빈도

Rare

설명

oaUtility 유틸리티에 force 옵션을 추가합니다. 이 옵션은 복제 대상 테이블에 프라이머리 키 제약 조건이 있는지 확인하지 않고 oraAdapter를 시작합니다.

재현 방법

- 재현 절차
- 수행 결과
- 예상 결과

Workaround

없음

변경사항

- Performance view
- Property
- Compile Option
- Error Code

BUG-50038 jdbcAdapter 및 oraAdapter 구동에서 예외 상황 발생 시 에러 메시지를 보완합니다.

module

rp-oraAdapter

Category

Functional Error

재현 빈도

Always

설명

jdbcAdapter 및 oraAdapter 구동에서 아래와 같은 예외 상황 발생 시 에러 메시지를 출력하도록 개선합니다.

- 초기화 실패
- 메모리 할당 실패
- Adapter 프로퍼티 파일 읽기 실패
- Adapter 트레이스 로그 파일 열기 실패
- Adapter 에러 메시지 파일 읽기 실패

재현 방법

- 재현 절차
- 수행 결과
- 예상 결과

Workaround

없음

변경사항

- Performance view
- Property
- Compile Option
- Error Code

BUG-50048 aku에 세션 접근 제어 기능을 추가합니다.

module

aku

Category

Functionality

재현 빈도

Always

설명

SLAVE AKU에서 수행하는 aku -p start 동작에 세션 접근 제어 기능을 추가합니다.

SLAVE AKU는 MASTER AKU에게 이중화 SYNC를 요청하기 전에 자신의 파드에 관리자 접근 모드를 설정하고 연결된 세션을 강제로 종료합니다. MASTER AKU에서 SLAVE AKU로 이중화 SYNC를 수행하고 SLAVE AKU에서 이중화 시작을 완료하면 관리자 접근 모드를 해제합니다.

재현 방법

- 재현 절차
- 수행 결과
- 예상 결과

Workaround

없음

변경사항

- Performance view
- Property
- Compile Option
- Error Code

BUG-50050 뷰/인라인 뷰에 중복되는 조건식을 제거하는 기능을 추가합니다.

module

qp

Category

Enhancement

재현 빈도

Always

설명

뷰/인라인 뷰에 중복되는 조건식을 제거하는 기능을 추가합니다.

뷰/인라인 뷰에서 아래에 해당하는 절에 중복되는 조건식이 있으면 제거됩니다. 이 기능이 적용되면 질의문의 실행 계획이 변경될 수 있습니다.

- FROM 절의 ON 절

- WHERE 절 (Outer Join 기호 (+)가 있을 때는 제외)
- HAVING 절

Altibase 7.1에서 이 기능을 사용하려면 비공개 프로퍼티 값을 변경하거나 비공개 힌트를 사용해야 합니다. 필요한 경우 Altibase 기술 지원 센터로 문의해주시기 바랍니다.

재현 방법

- 재현 절차

```
CREATE TABLE T1 (C1 INTEGER, C2 INTEGER, C3 CHAR(10), C4 CHAR(10));
ALTER TABLE T1 ADD CONSTRAINT T1_PK PRIMARY KEY (C2,C1);

CREATE TABLE T2 (C1 INTEGER, C2 INTEGER, C3 CHAR(10), C4 CHAR(10));
ALTER TABLE T2 ADD CONSTRAINT T2_PK PRIMARY KEY (C1);

CREATE INDEX T1_IDX_1 ON T1 ( C1,C2 );

ALTER SESSION SET EXPLAIN PLAN = ON;

SELECT *
  FROM (SELECT /*+ */
          A.C1, A.C2, B.C3
        FROM T1 A, T2 B
       WHERE 0 IS NULL
          OR 0 = A.C1
          AND A.C1 = B.C1 ) ;
```

- 수행 결과

```
c1          c2          c3
-----
No rows selected.
-----
PROJECT ( COLUMN_COUNT: 3, TUPLE_SIZE: 20, COST: 1290336.82 )
  JOIN ( METHOD: NL, COST: 1249804.63 )
    SCAN ( TABLE: SYS.T1 $$1_$NONAME_$A, FULL SCAN, ACCESS: 0, COST: 122.04 )
    SCAN ( TABLE: SYS.T2 $$2_$NONAME_$B, FULL SCAN, ACCESS: 0, COST: 116.76 )
-----
```

- 예상 결과

```
c1          c2          c3
-----
No rows selected.
-----
PROJECT ( COLUMN_COUNT: 3, TUPLE_SIZE: 20, COST: 487.20 )
  JOIN ( METHOD: INDEX_NL, COST: 81.88 )
    SCAN ( TABLE: SYS.T1 $$1_$NONAME_$A, INDEX: SYS.T1_IDX_1, RANGE SCAN,
ACCESS: 0, COST: 0.01 )
    SCAN ( TABLE: SYS.T2 $$2_$NONAME_$B, INDEX: SYS.T2_PK, RANGE SCAN, ACCESS:
0, COST: 116.76 )
-----
```

Workaround

없음

변경사항

- Performance view
- Property
- Compile Option
- Error Code

Changes

Version Info

altibase version	database binary version	meta version	cm protocol version	replication protocol version
7.1.0.8.2	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.

CM protocol Version

통신 프로토콜 버전은 변경되지 않았다.

Replication protocol Version

Replication 프로토콜 버전은 변경되지 않았다.

프로퍼티

추가된 프로퍼티

- PSM_MAX_DDL_REFERENCE_DEPTH

변경된 프로퍼티

삭제된 프로퍼티

성능 뷰

추가된 성능 뷰

변경된 성능 뷰

삭제된 성능 뷰