

- [Altibase 7.3.0.0.1 Release Notes](#)
 - [1. 시스템 요구사항](#)
 - [하드웨어 최저 사양](#)
 - [운영 체제 및 플랫폼](#)
 - [2. 릴리스 정보](#)
 - [2.1 Altibase 7.3 의 새로운 기능](#)
 - [2.1.1 AKU\(Altibase Kubernetes Utility\)의 지원](#)
 - [2.1.2 AltiShapeLoader 1.0제공](#)
 - [2.1.3 JDBC 4.2 스펙 지원](#)
 - [2.1.4 OpensSSL 3.0.8 지원](#)
 - [2.1.5 기능 개선 - SQL 확장](#)
 - [2.1.6 기능 개선 - Spatial SQL 개선](#)
 - [2.1.7 기능 개선 - 이중화 기능 개선](#)
 - [2.1.8 기능 개선 - 응용 프로그램 개발 인터페이스](#)
 - [2.1.9 기능 개선 - 내장패키지 및 함수](#)
 - [2.1.10 기능 개선 - 유틸리티](#)
 - [2.1.11 기능 개선 - JDBC Adapter, oraAdpater](#)
 - [2.1.12 성능 개선](#)
 - [2.1.13 고가용성](#)
 - [2.2 변경 사항](#)
 - [2.2.1 데이터베이스 버전](#)
 - [2.2.2 호환성](#)
 - [2.2.3 기타 변경사항](#)
 - [2.2.4 Altibase 서버 프로퍼티](#)
 - [2.2.5 메타 테이블](#)
 - [2.2.6 성능 분](#)
 - [2.3 패키지](#)
 - [2.4 다운로드](#)
 - [Package](#)
 - [Manual](#)
 - [설치](#)

Altibase 7.3.0.0.1 Release Notes

(2023.08)

1. 시스템 요구사항

하드웨어 최저 사양

- 1GB RAM (권장: 2GB)
- 1 CPU (권장: 2 CPUs)
- 4GB 하드 디스크 여유 공간 (권장: 12GB)

운영 체제 및 플랫폼

Altibase 7.3.0.0.1 는 아래 표에 나열된 운영체제와 플랫폼 상에서 운영 가능하다.

	Altibase 서버	Altibase 클라이언트	소프트웨어 요구사항
AIX on IBM Power Systems			
AIX 6.1	•	•	
Linux x86-64			
Red Hat Enterprise Linux 6 Red Hat Enterprise Linux 7 Red Hat Enterprise Linux 8	•	•	- GNU glibc 2.12 이상
Linux on Power			
Red Hat Enterprise Linux 6.5 이상	•	•	- GNU glibc 2.12 이상
Linux on Power (Little Endian)			
Red Hat Enterprise Linux 7.3 이상	•	•	- GNU glibc 2.17 이상
HP-UX Itanium (IA-64)			
HP-UX 11.31	•	•	

Altibase 서버/클라이언트 모두 64-bit 만 지원한다.

Red Hat Enterprise Linux 6, 7, 8 마이너 버전에 대해 호환성을 보장한다.

Java 버전: JDK 1.8 이상에서 호환된다.

2. 릴리스 정보

2.1 Altibase 7.3 의 새로운 기능

2.1.1 AKU(Altibase Kubernetes Utility)의 지원

AKU(Altibase Kubernetes Utility)는 쿠버네티스 환경에서 스케일링(scaling)할 때 파드(Pod)의 시작 및 종료에 따라 Altibase의 데이터를 동기화하거나 동기화 정보를 초기화하는 작업을 수행할 수 있게 도와주는 유틸리티이다.

2.1.2 AltiShapeLoader 1.0제공

altiShapeLoader는 셰이프파일[1]을 가져오기 내보내기를 수행하는 도구로 자바 기반의 오픈소스 GeoTools를 기반으로 개발되었다.

2.1.3 JDBC 4.2 스펙 지원

Altibase 7.3 에서 JDBC API Specification 4.2를 부분적으로 지원한다. Altibase 7.3 JDBC 드라이버에서 지원하는 JDBC 4.2 API는 [JDBC User's Manual - JDBC 4.2 API References](#) 에서 확인할 수 있다. 변경 사항 및 호환성 이슈는 이 문서의 [Altibase JDBC 7.3 변경 사항 및 호환성 이슈](#)에서 확인할 수 있다.

2.1.4 OpenSSL 3.0.8 지원

보안강화를 위해 OpenSSL의 최신버전 3.0.8 을 적용하여 지원하며, OpenSSL 1.0.x 버전은 더 이상 지원하지 않는다. 지원하는 프로토콜은 TLS 1.0, 1.2에 추가로 TLS 1.3을 지원한다. 만약 TLS 1.3의 특정 암호 알고리즘(CIPHER)을 사용하고자 하는 경우에는 Altibase 서버 프로퍼티 SSL_CIPHER_SUITES에 설정해야 한다. TLS 1.0, TLS 1.2의 경우는 기존 과 동일하게 SSL_CIPHER_LIST에 설정한다. 자세한 내용은 [Altibase SSL TLS User's Guide - 서버프로퍼티 설정](#) 을 참고한다.

추가로 FIPS 모듈의 사용을 지원하는데, 이 기능을 이용하기 위해서는 SSL_LOAD_CONFIG 프로퍼티를 1로 설정해야 한다. 자세한 내용은 [Altibase SSL TLS User's Guide - Step4 FIPS모듈을 사용할 경우](#) 를 참고한다.

2.1.5 기능 개선 - SQL 확장

VARRAY TYPE 지원

저장 프로시저 내에서 사용자 정의 타입으로 VARRAY가 추가되었다. VARRAY 타입은 동일한 데이터 타입의 연속된 데이터를 저장할 수 있는 ARRAY 형식의 사용자 정의 데이터 타입으로, 자세한 설명은 [Stored Procedures Manual - varray](#) 설명을 참고한다.

익명 블록(Anonymous Block) 지원

프로시저의 헤더 없이 바디블록으로 구성된 저장 프로시저로 DECLARE ... BEGIN... END;의 구조로 선언한다. 익명 블록은 PSM 객체를 생성하거나 데이터베이스에 저장하지 않고, RETURN 절의 값을 반환하지 않는 특징이 있다. 저장 프로시저와 달리 INPUT, OUTPUT, INOUTPUT 용도의 바인드 변수를 사용할 수 있다.

C/C++ External Procedure의 internal mode 프로시저 지원

Internal mode 프로시저는 에이전트 프로세스 없이 Altibase 서버에서 직접 동적 라이브러리를 로드하고 외부 프로시저를 직접 호출하는 방식으로 external mode에 비해 빠르게 동작한다. 자세한 설명은 [External Procedures Manual](#) 을 참고한다.

multiple update, delete 구문의 지원

multiple update, delete 구문을 지원한다. 자세한 내용은 [SQL Reference Manual - multiple delete](#) , [multiple update](#) 을 참고한다.

한글 검색 가능한 정규 표현식(Regular Expression) 지원

한글 검색 가능한 정규 표현식을 지원하기 위해 PCRE2 호환모드를 제공한다. PCRE2 호환 모드는 PCRE2 라이브러리의 정규 표현식 문법을 지원한다. 자세한 내용은 [SQL Reference Manual - a.부록: 정규 표현식](#) 을 참고한다.

fetch across rollback

CURSOR HOLD ON 기능을 이용하여 롤백할 때, Fetch out of sequence 에러가 발생하는 문제를 해결하기 위하여 fetch across rollback 기능을 지원한다.

CREATE QUEUE 및 ALTER QUEUE 구문에 DELETE 절 추가

큐(QUEUE) 테이블에 DELETE 문 허용 여부를 설정하는 DELETE 절이 추가되었다. 구문 사용 방법은 [SQL Reference Manual - CREATE QUEUE 구문](#) 을 참고한다. 관련하여 성능 뷰 [V\\$QUEUE DELETE OFF](#)가 추가되었다.

Sequence Restart 구문 지원

시퀀스를 초기화하기 위해서 ALTER SEQUENCE 문에서 RESTART 절을 지원한다. 자세한 설명은 [SQL Reference Manual - ALTER SEQUENCE 구문](#) 을 참고한다.

2.1.6 기능 개선 - Spatial SQL 개선

SRID(Spatial Reference Identifier) interface 지원

SRID(공간 참조 식별자)는 공간 객체를 구분하기 위해 지정하는 식별자로, 4바이트 범위 내 정수를 사용하며 GEOMETRY 칼럼에 적용할 수 있다. 테이블을 생성할 때 SRID를 지정할 수 있으며, ALTER TABLE 구문을 이용하여 SRID를 지정할 수 있다.

SRID의 지원으로 GEOMETRY 데이터타입 표현방법이 추가되었다.

- EWKT(Extended Well-Known Text) 형식: WKT 형식에 공간 객체를 표현하는 SRID(Spatial Reference Identifier) 정보가 추가된 것이다.
- EWKB(Extended Well-Known Binary) 형식: WKB 형식에 공간 객체를 표현하는 SRID(Spatial Reference Identifier) 정보가 추가된 것이다.

공간 함수의 추가

아래의 공간함수가 추가되었다.

- ASEWKT
- ASEWKB
- GEOMFROMEWKT
- GEOMFROMEWKB
- SETSRID
- SRID
- ST_Collect
- ST_IsCollection
- ST_LinestringFromWKB
- ST_MakeEnvelope
- ST_MakeLine
- ST_MakePoint
- ST_Point
- ST_PolygonFromText
- ST_Transform

Spatial API용 함수

아래의 Spatial API용 함수가 추가되었다.

- ACSTGetGeometrySRID

2.1.7 기능 개선 - 이중화 기능 개선

이중화 대상 테이블에 DDL 복제 기능 추가

이중화를 통하여 DDL 복제(Synchronization)가 가능하게 되었다. 이 기능을 사용하기 위해서는 각노드의 [REPLICATION_DDL_SYNC](#) 프로퍼티를 1로 설정해야 한다. 또한, 각 노드의 [REPLICATION_DDL_ENABLE](#) 프로퍼티를 1로 설정하고, [REPLICATION_DDL_ENABLE_LEVEL](#)이 동일하게 설정해야 한다.

DDL 복제를 사용하기 위해 다음의 제약 조건을 확인해야 한다.

- DDL 복제를 수행할 노드들의 이중화가 동작하고 있어야 한다.
- DDL 복제를 수행할 지역 노드와 원격 노드의 테이블 이름이 같아야 한다.
- DDL 복제를 수행할 지역 노드와 원격 노드의 테이블 파티션이름이 같아야 한다.
- DDL 복제를 수행할 이중화 대상 사용자의 이름이 같아야 한다.
- 한번에 하나의 노드에서만 DDL 복제를 수행해야 한다.
- DDL 복제를 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 복제를 허용하지 않는다.

이중화 수신 전용(RECEIVE_ONLY) 옵션 제공

이중화를 수신 전용 옵션으로 설정하여, 다른 노드로 변경 데이터를 전송하지 않는 기능을 제공한다. 수신 전용으로 이중화를 생성하면, 로그를 읽지 않으므로 네트워크 장애등의 이중화 이슈가 발생하여도 시스템에 영향을 주지 않는다. 자세한 설명은 [Replication Manual - 이중화 수신 전용 옵션](#) 을 참고한다.

2.1.8 기능 개선 - 응용 프로그램 개발 인터페이스

인피니밴드(InfiniBand) 지원

통신 성능 향상을 위해 RDMA(Remote Direct Memory Access) 통신 기반인 Infiniband를 지원한다.

JDBC 에 추가된 기능

- **Auto-loading of JDBC driver class**

명시적으로 Class.forName() 클래스를 로딩할 필요없이 META-INF/services/java.sql.Driver 파일을 이용한 자동 드라이버 로딩 기능 지원

- **Wrapper Pattern Support**

프록시에서 구현 객체에 대한 참조를 얻는 JDBC 4.0 표준 인터페이스를 지원한다. 커넥션풀 등에서 생성하는 프록시 객체에서 JDBC 객체를 획득할 수 있다.

```
try (Connection swrappedCon = dbPool.getConnection()) {
    if (swrappedCon.isWrapperFor(AltibaseConnection.class)) {
        AltibaseConnection connection =
            swrappedCon.unwrap(AltibaseConnection.class);
        ...
        ...
    }
}
```

- **National Character Set Support**

JDBC 4.0 스펙인 표준 다국어 처리 인터페이스 지원

- **Aborting Connections**

비동기적으로 데이터베이스와의 물리적 연결을 종료하는 Connection.abort() 인터페이스 지원

- **Standard Socket Network Timeout API Support**

데이터베이스 서버로부터 소켓 응답 대기 시간을 설정하는 표준 인터페이스 Connection.setNetworkTimeout() 지원

- **Connection Management Enhancements**

Validation Query없이 Connection 객체에서 유효성 검사를 수행하는 Connection.isValid() 지원

- **Large Update Counts Support**

대용량 레코드 업데이트를 위한 executeLargeUpdate(), executeLargeBatch() 지원

- **Set Client Information Support**

Connection.setClientInfo()를 이용한 클라이언트 어플리케이션 속성(name) 설정 지원

- **java.sql.SQLType interface Support**

JDBC 4.2 표준 인터페이스 java.sql.SQLType을 구현한 AltibaseJDBCType 지원

- **Try-with-resources** 구문을 통한 자동 JDBC 리소스 해제

```
try (Statement stmt = con.createStatement()) {
    ResultSet rs = stmt.executeQuery(query);
    while (rs.next()) {
        String coffeeName = rs.getString("aaa");
        int supplierID = rs.getInt("bbb");
    }
}
```

- **SQLException에 Enhanced for-each loop** 사용을 지원

```
catch(SQLException ex) {
    for(Throwable e : ex ) {
        LOG.error("Error occurred: " + e);
    }
}
```

2.1.9 기능 개선 - 내장패키지 및 함수

DBMS_STANDARD 패키지 제공

DBMS_STANDARD 패키지를 통해서 트리거 이벤트를 확인하는 함수를 제공한다.

DBMS_METADATA 패키지 제공

DBMS_METADATA 패키지는 데이터베이스 디렉터리로부터 객체 생성 DDL 구문 또는 권한 GRANT 구문을 추출하는 기능을 제공한다.

DBMS_SQL_PLAN_CACHE 패키지 제공

특정 실행 계획(Execution Plan)을 SQL Plan Cache에 유지하거나 삭제하는 기능을하는 저장 프로시저를 제공한다.

DBMS_OUTPUT 패키지에 print_enable/print_disable 프로시저 추가

PSM내에서 println 기능을 enable, disable 할수 있는 기능을 제공하며, 세션 단위로 수행된다.

DBMS_LOCK 패키지에 sleep2 프로시저 추가

마이크로초(micro second) sleep 을 지원하는 시스템 저장 프로시저 sleep2가 추가되었다.

SYS_SPATIAL 패키지

SPATIAL_REF_SYS 테이블에 Spatial Reference System 메타 데이터를 등록, 삭제하는 기능을 제공한다.

UTL_COPYSWAP 패키지

UTL_COPYSWAP 패키지는 테이블 스키마 복사, 데이터 복제, 테이블 교환 인터페이스를 제공한다.

2.1.10 기능 개선 - 유틸리티

Altimon의 AIX 7, Power Linux LE(Little endian)에서 동작 지원

AIX 7 버전 및 Power Linux LE에서도 altimon을 사용할 수 있다.

Alticom 커밋 카운트 설정 기능 추가

커밋(commit) 카운트를 설정할 수 있는 프로퍼티 COUNT_TO_COMMIT가 추가되었다. 관련 내용은 [Utilities Manual](#) 에서 확인할 수 있다.

2.1.11 기능 개선 - JDBC Adapter, oraAdpater

LOB데이터 타입 지원

LOB 데이터 타입의 지원을 위해 ADAPTER_LOB_TYPE_SUPPORT 프로퍼티가 추가되었다. LOB 데이터 타입 지원 기능을 사용하려면 ADAPTER_LOB_TYPE_SUPPORT 프로퍼티의 값을 1로 변경한 다음, ADAPTER를 재 시작 해야 한다.

오프라인 옵션 제공

어댑터(JDBC Adapter, oraAdapter)를 이용하여 Altibase에서 변경된 데이터를 타겟 데이터베이스에 적용할 때, Altibase 서버에서 장애가 발생하는 경우를 대비하기 위한 기능이다. 자세한 설명은 [Adapter for JDBC User's Manual - 오프라인 옵션](#) 및 [Adapter for Oracle User's Manual - 오프라인 옵션](#) 을 참고한다.

2.1.12 성능 개선

TABLE LOCK 병목구간 개선

테이블 잠금(TABLE LOCK) 관리자 타입을 지정하는 LOCK_MGR_TYPE 프로퍼티를 삭제하고, 새로운 테이블 잠금 모드(light mutex mode)를 적용하여 테이블 잠금 병목구간을 개선하였다.

TABLESPACE MANAGER MUTEX 병목구간 개선

테이블 스페이스 뮤텍스에서 불필요한 잠금(LOCK)을 제거하도록 개선하였다.

디스크 템프 테이블 성능 개선

디스크 템프 테이블을 사용하는 SQL의 성능을 향상시키고, 메모리 사용량을 개선하였다.

트랜잭션 로그 기록 성능 향상

로그 압축 알고리즘을 압축 속도가 빠른 LZ4 로 변경하였다.

OLTP Scalability 향상

- Linux x86-64 CPU 코어 수 24코어 이상에서 조회 트랜잭션 성능 저하 현상 개선
- 메모리 DB 삭제(DELETE) 트랜잭션 성능 향상을 위해 로깅 구조 개선
- 디스크 DB 변경 트랜잭션 성능 향상을 위해 In-place MVCC 동작 방식 개선
- INSERT/UPDATE 트랜잭션 처리 시 불필요한 트랜잭션 로그 기록을 제거
- 트랜잭션 로그파일 압축 시 메모리 할당/해제 병목 개선
- 커밋 병목 및 가비지 콜렉션 쓰레드 병목 개선
 - 트랜잭션 커밋 후 테이블 정보 업데이트 병목 개선
- 메모리 DB 트랜잭션 성능 향상
 - 디스크 읽기를 유발하는 함수의 병목을 제거
 - Group Commit Log 기능 추가

인덱스 성능 개선

- 서버 시작 시 POINTER BASE 인덱스 생성 시간 단축 및 메모리 사용량 개선
- 서버 시작 시 VALUE BASE 인덱스 생성 시간 단축 및 메모리 사용량 개선

데이터베이스 구동 성능 개선

서버 구동시 인덱스 빌드에 사용되는 쓰레드들의 관리를 개선하여, 서버 구동 성능이 향상되었다.

휘발성/비휘발성 메모리 DB 트랜잭션 성능 향상

메모리 테이블 객체 식별자 추적 단계를 간소화하여 휘발성/비휘발성 메모리 DB 트랜잭션 성능이 향상되었다.

DEQUEUE 병렬 수행시 성능 개선

병렬로 DEQUEUE 수행 시 발생하는 병목을 제거하여 성능을 개선하였다.

Common Subexpression Elimination의 PREPARE 시간 단축

CSE(Common Subexpression Elimination)는 조건절의 중복된 조건식을 찾아 제거하는 최적화 기능이다. CSE 수행 알고리즘을 개선하여 관련 쿼리의 성능을 개선하였다.

메모리 파티션드 테이블의 Simple query 최적화로 성능개선

기존에는 메모리 테이블에 대해서만 simple query 최적화를 지원하였으나, 메모리 파티션드 테이블의 경우도 지원하게 되었다. 메모리 파티션드 테이블의 simple query 최적화 지원으로 메모리 파티션드 테이블의 DML 성능이 개선되었다.

Row Filter 수행 성능 개선 - SERIAL FILTER 적용

Filter 연산자를 직렬화 및 함수 호출구조의 최적화를 통해 row filter 수행 성능을 개선하였다. 이 기능을 사용하기 위해 SERIAL_FILTER 힌트 및 SERIAL_EXECUTE_MODE 프로퍼티가 추가되었다. 실행 계획에서 FILTER SERIAL EXECUTE 를 확인할 수 있다.

스칼라 서브쿼리(Scalar Subquery) 성능 개선

스칼라 서브쿼리의 수행방식을 개선하여 수행 성능을 개선하였다.

PSM에서 for loop절의 성능개선

이중화 Sender 성능 향상

- 압축 로그에서 이중화에 필요한 로그만 압축 해제하는 기능 추가
- xLog 압축 알고리즘을 LZO에서 LZ4로 변경

마이그레이션 성능 개선

대용량 데이터 이전을 위한 데이터 삽입의 성능이 개선되었다. iloader의 성능 옵션에 -lightmod가 추가되었다. 자세한 설명은 [iLoader User's Manual - 성능옵션](#)에서 확인할 수 있다.

JDBC fetch 성능 개선

JDBC fetch 성능 향상을 위해 ResultSet 객체 사용방식을 개선하였다. 동일한 PreparedStatement 객체에서 여러개의 ResultSet 객체를 생성하는 경우, 첫번째 ResultSet 객체를 재사용한다. ResultSet 객체의 재사용을 하지 않으려면 JDBC 연결 속성 중 reuse_resultset 속성의 값을 false로 변경하면 된다.

2.1.13 고가용성

DDL PVO 안정성 향상

DDL PVO 단계에서의 예외처리 개선으로 안정성을 향상시켰다.

프로토콜 유효성 검증 개선

유효하지 않은 패킷(malformed packet) 전송으로 인한 서버에서의 비정상 종료 및 비정상 동작이 발생하지 않도록 개선되었다. 프로토콜 처리시 패킷의 유효성을 체크하여 비정상적인 경우, 클라이언트의 접속을 끊고 진단로그를 남기도록 개선되었다. 이를 위해 CM_MSGLOG_FLAG의 기본값이 3으로, SERVER_MSGLOG_FLAG의 기본값이 15로 변경되었다.

트랜잭션 안정성 개선 - Multiple Rollback Segment

동시에 수행가능한 최대 디스크 트랜잭션의 개수를 기존 512개에서 16384로 확장하였다.

언두(undo) 테이블스페이스 재사용 안정성 향상

언두 테이블스페이스와 디스크 인덱스의 불필요한 관계를 제거하여 버그 발생 위험 요소 제거하였다. 디스크 페이지 공간 효율 개선으로 관련 프로퍼티들의 기본값 및 최대값이 변경되었다.

- INDEX_INITTRANS 최대값이 30에서 50으로 변경
- INDEX_MAXTRANS 기본값과 최대값이 30에서 50으로 변경

2.2 변경 사항

DBA와 개발자가 알아야 할 추가, 변경, 제거된 기능을 아래에서 설명한다.

2.2.1 데이터베이스 버전

데이터베이스 구성 요소 별 버전

Altibase 버전	데이터베이스 바이너리 버전	메타 버전	통신 프로토콜 버전	이중화 프로토콜 버전
7.1.0.8.8	6.5.1	8.11.1	7.1.7	7.4.7
7.3.0.0.1	7.3.0	9.3.1	7.1.8	7.4.9

2.2.2 호환성

데이터베이스 바이너리 버전

데이터베이스 바이너리 버전은 데이터베이스 이미지 파일과 로그 파일의 호환성을 나타낸다.

로그 파일 로깅 구조 개선으로 데이터베이스 바이너리 버전이 변경되었다. **Altibase 7.3 이전 버전 데이터베이스와 호환되지 않으므로 Altibase 버전 업그레이드 시 마이그레이션 작업이 필요하다.**

메타 버전

메타 메이저 버전(META MAJOR VERSION)이 변경되었으므로, **Altibase 7.3 이전 버전에서 Altibase 7.3으로 업그레이드시 메타를 재구성해야 한다.**

통신 프로토콜 버전

Altibase 서버와 클라이언트 간 통신 규약 호환성을 의미하며 클라이언트 하위 호환성을 알 수 있다.

통신 프로토콜 버전 중 상위 두 자리는 같고 패치 버전이 변경되었다. 메이저 버전과 마이너 버전이 같으면 클라이언트 하위 호환성을 보장한다.

클라이언트 하위 호환성은 하위 버전 Altibase 라이브러리로 컴파일한 사용자 응용 프로그램(Altibase 클라이언트)이 상위 버전 Altibase 에서 정상 동작하는 것을 보장한다.

이중화 프로토콜 버전

이중화 프로토콜 버전은 Altibase 이중화 하위 호환성이나 이중화 부가기능 호환 여부를 나타낸다.

메이저 버전과 마이너 버전 변경이 없어 LAZY 모드 이중화는 Altibase 이중화 하위 호환성을 보장하지만 패치 버전 변경으로 이중화 부가기능은 호환되지 않는다.

Altibase 이중화 하위 호환성

Altibase 이중화 하위 호환성이란 이중화 프로토콜 버전이 낮은 버전에서 높은 버전으로 단방향 이중화가 가능함을 의미하며 이중화 프로토콜 버전에서 상위 두 자리(메이저와 마이너 버전)가 같은 경우 보장한다.

Altibase 이중화 하위 호환성은 LAZY 모드 이중화로 제한한다.

EAGER 모드 이중화는 하위 호환성을 보장하지 않는다.

DDL 복제는 이중화 프로토콜 버전 세 자리가 모두 일치해야하므로 하위 호환성을 보장하지 않는다.

오프라인 이중화를 포함한 이중화 부가기능은 하위 호환성을 보장하지 않는다.

2.2.3 기타 변경사항

aexport 변경사항

Altibase 7.3 aexport를 구동하기 위해서는 DBMS_METADATA 패키지를 반드시 설치해야 한다. 그렇지 않은 경우, 아래의 에러메시지가 출력된다.

```
[ERR-91144 : DBMS_METADATA package does not exist.]
```

Altibase JDBC 4.2 관련 변경 사항 및 호환성 이슈

Altibase JDBC 4.2는 Altibase JDBC 3.0 에 대해 하위 호환성을 보장하지만 일부 인터페이스의 경우 JDBC API Specification 4.2에 따라 동작이 변경되었다.

미지원 기능에 대한 예외 처리 클래스 변경

다음 인터페이스에 대한 예외 처리 클래스가 SQLException에서 SQLFeatureNotSupportedException으로 변경되었다. SQLFeatureNotSupportedException은 SQLException의 하위 클래스이므로 기존 사용자 프로그램은 수정없이 그대로 동작한다.

- Altibase.jdbc.driver.AltibaseConnection
 - setTypeMap(Map)
- Altibase.jdbc.driver.AltibaseStatement
 - setCursorName(String)
- Altibase.jdbc.driver.AltibasePreparedStatement
 - setArray(int, Array)
 - setRef(int, Ref)
 - setURL(int, URL)
 - setUnicodeStream(int, InputStream, int)
- Altibase.jdbc.driver.Blob
 - position(Blob, long)
 - position(byte[], long)
- Altibase.jdbc.driver.Clob
 - position(Clob, long)
 - position(String, long)
- Altibase.jdbc.driver.CallableStatement
 - getArray(int)
 - getObject(int, Map)

- getRef(int)
- getURL(int)
- Altibase.jdbc.driver.AltibaseDatabaseMetaData
 - getColumnPrivileges(String, String, String, String)
 - getUDTs(String, String, String, int[])
- Altibase.jdbc.driver.AltibaseResultSet
 - getCursorName()
 - getArray(int)
 - getObject(int, Map)
 - getRef(int)
 - getURL(int)
 - getUnicodeStream(int)
 - updateArray(int, Array)
 - updateRef(int, Ref)

DatabaseMetaData의 일부 인터페이스 결과에 항목 추가

getProcedures(), getProcedureColumns(), getFunctions(), getFunctionColumns() 인터페이스 결과에 SPECIFIC_NAME 컬럼이 추가되었다.

Altibase JDBC 7.3 에서 SPECIFIC_NAME은 다음과 같은 형태로 구현하였다.

```
ProcName(FuncName) + '_' + ouid
```

연결 속성 기본값 변경

- [reuse_resultset](#)
 - Altibase 7.3 기본값은 true로 ResultSet 객체를 재사용하지만, Altibase 7.1 기본값은 false로 재사용하지 않는다.
- [lob_null_select](#)
 - LOB 컬럼 값이 NULL일 때 getBlob(), getClob() 수행시 NULL을 반환할 수 있도록 JDBC 연결속성 lob_null_select가 추가되었다.
 - Altibase 7.3 기본값은 off로 NULL을 반환한다. Altibase 7.1 기본값은 on으로 LOB 객체를 반환한다.

Altibase JDBC 4.2만을 위한 JDBC 연결 속성 추가

- [getprocedures_return_functions](#)
 - DatabaseMetaData.getProcedures(), getProcedureColumns()의 결과에 function 결과를 포함할지 설정한다. JDBC API Specification 4.2 표준은 function 정보를 제외하지만 Altibase JDBC 4.2는 클라이언트 하위 호환성을 위해 하위 버전과 같게 유지한다. 표준에 따라 function정보를 제외하려면 속성값을 false로 설정한다.

CLIENT_TYPE 변경

Altibase 7.3 JDBC 세션의 CLIENT_TYPE은 NEW_JDBC42이다. Altibase 7.3 JDBC Driver 를 이용하여 컴파일 또는 실행한 경우 V\$SESSION의 CLIENT_TYPE 값은 NEW_JDBC42 로 조회해야 한다.

SQL 결과 및 실행 계획 변화

- 서브쿼리의 인라인 뷰에 ORDER BY절 사용 시 SQL 성능 개선
이 영향을 받는 SQL의 실행 계획에 변화가 있다. SUBQUERY FILTER 안에 SORT 플랜 노드 없어진다.
- 중첩된 LEFT OUTER JOIN 수행 방식을 최적화
이 영향을 받는 SQL에서 실행 계획 변경 및 SQL 수행 결과가 달라질 수 있다.
- Subquery Unnesting 관련 기능 변경 및 추가
이 영향을 받는 SQL에서 실행 계획이 변경될 수 있다.

신규 기능 관련 Altibase 이중화 제약사항

Altibase 7.1 과 Altibase 7.3 양방향 이중화 제약 사항

Altibase 7.1과 Altibase 7.3는 DDL 복제와 오프라인 이중화가 불가하다.

DDL 복제는 이중화 프로토콜 버전(replication protocol version) 세 자리가 모두 일치해야 하는 기능으로, 하위 호환성을 보장하지 않는다.

오프라인 이중화는 바이너리 데이터베이스 버전(binary db version) 세 자리가 모두 일치해야 하는 이중화 부가 기능으로 하위 호환성을 보장하지 않는다.

Altibase 6.5.1 과 Altibase 7.3 양방향 이중화 제약 사항

Altibase 이중화 하위 호환성 보장에 따라 Altibase 6.5.1와 Altibase 7.3 간 단방향 및 양방향 LAZY 모드 이중화는 가능하다. 단, 이중화 대상 테이블에 공간 데이터 타입 컬럼이 있는 경우 Altibase 7.3 에서 Altibase 6.5.1 로 이중화하는 경우 SRID 값을 가진 데이터를 Altibase 6.5.1 로 동기화할 수 없다.

2.2.4 Altibase 서버 프로퍼티

Altibase 7.3.0.0.1 에서 추가, 변경, 삭제된 Altibase 서버 프로퍼티들이다. 각 프로퍼티에 대한 자세한 내용은 [General Reference-1.Data Types & Altibase Properties](#)를 참고하기 바란다.

새로운 프로퍼티

- [DISK INDEX BUILD SORT AREA SIZE](#)
- [DBLINK GLOBAL TRANSACTION LEVEL](#)
- [IB_CONCHKSPIN](#)
- [IB_ENABLE](#)
- [IB_LATENCY](#)
- [IB_LISTENER DISABLE](#)

- [IB MAX LISTEN](#)
- [IB PORT NO](#)
- [INIT TOTAL WA SIZE](#)
- [IPCD A SEM KEY](#)
- [IPCD A SHM KEY](#)
- [IPC SHM KEY](#)
- [IPC SEM KEY](#)
- [JOB MSGLOG COUNT](#)
- [JOB MSGLOG FILE](#)
- [JOB MSGLOG FLAG](#)
- [JOB MSGLOG SIZE](#)
- [LISTAGG PRECISION](#)
- [MATHEMATICS TEMP MEMORY MAXIMUM](#)
- [NETWORK ERROR LOG FILE](#)
- [PSM MAX DDL REFERENCE DEPTH](#)
- [REGEXP MODE](#)
- [REPLICATION DDL SYNC](#)
- [REPLICATION DDL SYNC TIMEOUT](#)
- [REPLICATION GAP UNIT](#)
- [REPLICATION IB LATENCY](#)
- [REPLICATION IB PORT NO](#)
- [REPLICATION META ITEM COUNT DIFF ENABLE](#)
- [REPLICATION RECEIVER APPLIER YIELD COUNT](#)
- [REPLICATION SENDER IP](#)
- [SERIAL EXECUTE MODE](#)
- [SERVICE THREAD RECV TIMEOUT](#)
- [SSL CIPHER SUITES](#)
- [SSL LOAD CONFIG](#)
- [ST MSGLOG COUNT](#)
- [ST MSGLOG FILE](#)
- [ST MSGLOG FLAG](#)
- [ST MSGLOG SIZE](#)
- [VARRAY MEMORY MAXIMUM](#)

변경된 프로퍼티

- [ARCHIVE FULL ACTION](#)
읽기 전용에서 변경 가능으로 속성이 변경되었다. 기본값 변경은 없으나, 설정값 2가 추가되었다.
- [CM MSGLOG FLAG](#)
기본값이 3으로 변경되었다.
- [EXECUTE STMT MEMORY MAXIMUM](#)
기본값이 1073741824에서 2147483648로 변경되었다.
- [HASH AREA SIZE](#)
최소값이 512K 에서 3M로 변경되었다.
- [INDEX INITRANS](#)
최대값이 30에서 50으로 변경되었다.
- [INDEX MAXTRANS](#)
기본값과 최값이 30에서 50으로 변경되었다.
- [LOB CACHE THRESHOLD](#)
최대값이 8192에서 524288로 변경되었다.
- [MEMORY INDEX BUILD RUN SIZE](#)
기본값이 32768에서 131072으로 변경되었다.
- [MM MSGLOG FILE](#)
기본값이 1로 변경되었다.
- [PSM CHAR DEFAULT PRECISION](#)
기본값이 32767에서 32000으로 변경되었다.
- [PSM NCHAR UTF16 DEFAULT PRECISION](#)
기본값이 16383에서 16000으로 변경되었다.
- [PSM NCHAR UTF8 DEFAULT PRECISION](#)
기본값이 10921에서 10666으로 변경되었다.
- [PSM NVARCHAR UTF16 DEFAULT PRECISION](#)
기본값이 16383에서 16000으로 변경되었다.
- [PSM NVARCHAR UTF8 DEFAULT PRECISION](#)
기본값이 10921에서 10666으로 변경되었다.
- [PSM VARCHAR DEFAULT PRECISION](#)
기본값이 32767에서 32000으로 변경되었다.
- [REPLICATION EAGER PARALLEL FACTOR](#)
최소값이 1에서 2로 변경되었다.
- [SERVER MSGLOG FLAG](#)
기본값이 7에서 15로 변경되었다.
- [TOTAL WA SIZE](#)

최소값이 0으로 변경되었다.

- [TRANSACTION SEGMENT COUNT](#)

최대값이 512에서 16384로 변경되었다.

삭제된 프로퍼티

- GLOBAL_TRANSACTION_LEVEL
- LOCK_MGR_TYPE
- LOCK_MGR_SPIN_COUNT
- LOCK_MGR_MIN_SLEEP
- LOCK_MGR_MAX_SLEEP
- LOCK_MGR_DETECTDEADLOCK_INTERVAL
- TEMP_MAX_PAGE_COUNT
- TRANSACTION_START_MODE

2.2.5 메타 테이블

새로운 메타테이블

- [SYS GEOMETRIES](#)
- [SYS GEOMETRY COLUMNS](#)
- [SYS REPL RECEIVER](#)
- [SYS REPL TABLE OID IN USE](#)
- [USER SRS](#)

변경된 메타테이블

- [SYS REPLICATIONS](#)
 - REMOTE_LAST_DDL_XSN 컬럼이 추가되었다.
- [SYS REPL HOSTS](#)
 - CONN_TYPE 컬럼이 추가되었다.
 - IB_LATENCY 컬럼이 추가되었다.
- [SYS REPL OLD COLUMNS](#)
 - MT_SRID 컬럼이 추가되었다.
- [SYS REPL OLD ITEMS](#)
 - REMOTE_USER_NAME 컬럼이 추가되었다.
 - REMOTE_TABLE_NAME 컬럼이 추가되었다.
 - REMOTE_PARTITION_NAME 컬럼이 추가되었다.
 - PARTITION_COUNT 컬럼이 추가되었다.
 - PARTITION_METHOD 컬럼이 추가되었다.

- PARTITION_ORDER 컬럼이 추가되었다.
- PARTITION_MIN_VALUE 컬럼이 추가되었다.
- PARTITION_MAX_VALUE 컬럼이 추가되었다.
- INVALID_MAX_SN 컬럼이 추가되었다.

삭제된 메타테이블

아래의 메타 테이블이 삭제되었다.

- STO_COLUMNS_
- STO_DATUMS_
- STO_ELLIPSOIDS_
- STO_GEOCCS_
- STO_GEOGCS_
- STO_PRIMEMS_
- STO_PROJCS_
- STO_PROJECTIONS_
- STO_SRS_
- STO_USER_COLUMNS_

2.2.6 성능 뷰

아래의 성능 뷰 들이 추가되었다. 각 성능 뷰에 대한 자세한 내용은 [General Reference-2.The Data Dictionary](#)를 참고하기 바란다.

새로운 성능 뷰

- [V\\$LIBRARY](#)
- [V\\$PROCINFO](#)
- [V\\$QUEUE DELETE OFF](#)
- [V\\$REPL REMOTE META CHECKS](#)
- [V\\$REPL REMOTE META COLUMNS](#)
- [V\\$REPL REMOTE META INDEX COLUMNS](#)
- [V\\$REPL REMOTE META INDICES](#)
- [V\\$REPL REMOTE META ITEMS](#)
- [V\\$REPL REMOTE META REPLICATIONS](#)

삭제된 성능뷰

- V\$ST_ANGULAR_UNIT
- V\$ST_AREA_UNIT
- V\$ST_LINEAR_UNIT

2.3 패키지

OS	CPU	서버/클라이언트	패키지 인스톨러 이름
AIX	PowerPC	Altibase 서버	altibase- server-7.3.0.0.1-AIX-POWERPC-64bit-release.run
		Altibase 클라이언트	altibase- client-7.3.0.0.1-AIX-POWERPC-64bit-release.run
HP-UX	IA64	Altibase 서버	altibase- server-7.3.0.0.1-HPUX-IA64-64bit-release.run
		Altibase 클라이언트	altibase- client-7.3.0.0.1-HPUX-IA64-64bit-release.run
LINUX	x86-64	Altibase 서버	altibase-server-7.3.0.0.1-LINUX-X86-64bit-release.run
		Altibase 클라이언트	altibase-client-7.3.0.0.1-LINUX-X86-64bit-release.run
LINUX	PowerPC	Altibase 서버	altibase-server-7.3.0.0.1-LINUX-POWERPC-64bit-release.run
		Altibase 클라이언트	altibase-client-7.3.0.0.1-LINUX-POWERPC-64bit-release.run
LINUX	PowerPCLE (Little Endian)	Altibase 서버	altibase-server-7.3.0.0.1-LINUX-POWERPCLE-64bit-release.run
		Altibase 클라이언트	altibase-client-7.3.0.0.1-LINUX-POWERPCLE-64bit-release.run

2.4 다운로드

Package


<http://support.altibase.com>

Manual

https://github.com/ALTIBASE/Documents/blob/master/Manuals/Altibase_7.3/kor/README.md

설치

[Altibase Installation Guide](#) 참고

[1] 셰이프 파일(Shapefile) : 지리정보시스템 소프트웨어 개발사 ESRI에서 개발한 파일 형식으로 지리정보시스템(GIS) 분야에서 표준 파일이다. 셰이프 파일은 아래 3가지 파일들이 필수로 구성되어 있다. 

- shp : 벡터 형식으로 점, 선, 도형을 표현한 공간 데이터 정보를 가지고 있다.
- shx : 인덱스 파일. shp 파일에 담겨있는 도형 정보의 위치를 담고 있다.
- dbf : shp 파일의 도형 정보에 대한 속성 정보를 담은 dBASE 테이블 파일이다.

참고 : [Geoprocessing considerations for shapefile output](#)