

# ALTIBASE 전환가이드

대상 DB : Oracle

(주)알티베이스

## 개정이력

| 버전  | 변경일        | 변경내용          | 작성자 |
|-----|------------|---------------|-----|
| 1.0 | 2014.09.01 | 오라클 전환가이드 1.0 | 공공팀 |
|     |            |               |     |
|     |            |               |     |
|     |            |               |     |
|     |            |               |     |
|     |            |               |     |
|     |            |               |     |
|     |            |               |     |

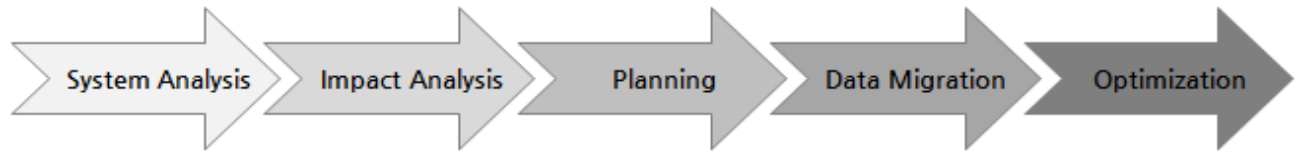
## 목 차

|   |    |
|---|----|
| 개요.....                                       | 5  |
| 1. 환경 구성.....                                 | 6  |
| 가. 지원 버전.....                                 | 6  |
| 나. 사전 점검 항목.....                              | 6  |
| 1) 환경 분석.....                                 | 6  |
| 2) Oracle DBMS 분석.....                        | 6  |
| 3) 비즈니스 분석.....                               | 6  |
| 1.2 환경설정.....                                 | 7  |
| 가. OBJECT 변환.....                             | 7  |
| 1) 데이터 타입.....                                | 7  |
| 2) Object 비교.....                             | 8  |
| 3) CREATE TABLESPACE.....                     | 9  |
| 4) CREATE TABLE.....                          | 10 |
| 5) CREATE INDEX.....                          | 14 |
| 6) CREATE VIEW.....                           | 16 |
| 7) CREATE TRIGGER.....                        | 16 |
| 8) CREATE SEQUENCE.....                       | 17 |
| 9) CREATE SYNONYM.....                        | 17 |
| 10) ALTER TABLE.....                          | 17 |
| 나. SQL 변환.....                                | 18 |
| 1) Math Function.....                         | 18 |
| 2) String Function.....                       | 20 |
| 3) Date Function.....                         | 22 |
| 4) Compare Function.....                      | 23 |
| 5) Convert Function.....                      | 24 |
| 6) Encode/Decode Function.....                | 24 |
| 7) Large Object Function.....                 | 25 |
| 8) Analyze Function.....                      | 25 |
| 다. STORED PROCEDURE/FUNCTION 변환.....          | 26 |
| 1) ALTIBASE AUTOCOMMIT 모드에서 Procedure 처리..... | 26 |
| 2) PARAMETER의 TYPE 과 RETURN TYPE.....         | 26 |
| 3) 파일 및 출력 처리.....                            | 27 |
| 4) REF CURSOR.....                            | 27 |
| 5) WHERE CURRENT OF 구문.....                   | 27 |
| 6) EXCEPTION.....                             | 28 |

|  |           |
|--|-----------|
| <b>1.3 전환절차 .....</b>  | <b>29</b> |
| 가. DATA 전환 및 APPLICATION 작업 절차 .....                           | 29        |
| 나. DATA 정합성 확인 .....   | 30        |
| APPLICATION 검증.....  | 30        |
| - 고객과의 사전 협의를 통하여 검증 방안 도출.....                                | 30        |
| 예) 온라인 화면 Capture 또는 장표 비교, 배치 Report 비교 등 .....               | 30        |
| DATA 검증 .....  | 30        |
| - Migration 결과 레포트를 통한 원본 DBMS와 ALTIBASE의 테이블 데이터 총 건수 비교..... | 30        |
| - 고객과의 사전 협의를 통하여 검증 방안 도출.....                                | 30        |
| 예) 데이터 총 건수 확인, 특정 칼럼 데이터 합산 값 비교, 특정 쿼리 수행 결과 비교 등.....       | 30        |
| 다. DATA 전환 장애 대처 방안.....                                       | 30        |
| <b>1.4 전환 소요시간 .....</b>                                       | <b>31</b> |
| <b>1.5 전환 도구 .....</b>   | <b>32</b> |
| 가. MIGRATION CENTER.....                                       | 32        |
| 1) Migration Center 이점.....                                    | 32        |
| 나. 사용법 .....   | 32        |
| 1) Add Database Connection.....                                | 32        |
| 3) Build Project.....  | 34        |
| 4) Reconcile Project.....                                      | 34        |
| 5) Run Project .....   | 35        |
| 6) PL/SQL Converter Tool.....                                  | 36        |
| 다. 다운로드.....   | 36        |
| 라. 문의사항.....   | 36        |

## 개요

본 문서는 Oracle DBMS를 ALTIBASE HDB v6.3 으로 전환하는 절차에 대한 문서입니다.



| 단계  |         | 내용   | 산출물      | 담당자                |
|-----|---------|--|----------|--------------------|
| 1단계 | 시스템 분석  | 요구사항 분석  | 시스템 분석서  | • DBA<br>• APP 개발자 |
|     |         | 환경 분석<br>• OS, HW, SW 등  |          | • DBA<br>• APP 개발자 |
|     |         | DBMS 분석<br>• Data Size (건, Bytes)<br>• Object 수 (Table, Index 등)<br>• Tablespace Size<br>• Function, Procedure |          | • DBA<br>• APP 개발자 |
|     |         | 비즈니스 분석<br>• 온라인성/배치성 업무의 비율<br>• 배치성 업무의 Query 복잡도  |          | • DBA<br>• APP 개발자 |
| 2단계 | 영향도 분석  | 위험 영향도 분석  | 영향도 보고서  | • DBA<br>• APP 개발자 |
|     |         | 테이블 사용과 관련 매트릭스 에 대한 어플리케이션과 데이터베이스간의 연관성 분석   |          | • DBA<br>• APP 개발자 |
| 3단계 | 일정계획 수립 | 일정 계획 수립   | 일정표      | • PM               |
|     |         | 자원 인력 계획 수립  |          |                    |
| 4단계 | 전환      | Migration Center를 활용한 전환<br>- Schema 및 Data 포함   |          | • DBA              |
|     |         | 비표준 Schema 및 SP 에 대한 수동 변환   |          | • 개발자              |
|     |         | 데이터 검증   |          | • DBA<br>• APP 개발자 |
| 5단계 | 최적화     | 파라미터 튜닝을 통한 구성 최적화   | 이관결과 보고서 | • DBA              |
|     |         | SQL 및 Procedure 튜닝을 통한 성능 최적화  |          | • DBA<br>• APP 개발자 |

# 1. 환경 구성

ALTIBASE HDB로 변경 가능한 Oracle의 버전 및 환경 구성을 위한 사전 점검 항목은 다음과 같습니다.

## 가. 지원 버전

- Oracle 9i – 11g

## 나. 사전 점검 항목

### 1) 환경 분석

- Oracle 버전
- Oracle DBMS 설치된 시스템의 OS 및 HW Spec(Cpu, Memory, Disk 등)
- ALTIBASE HDB 제품 및 버전
- ALTIBASE HDB 설치될 시스템의 OS 및 HW Spec(Cpu, Memory, Disk 등)
- DBMS 서버의 Storage 제품, 사양(Cpu, Memory Network) 및 구성 (RAID1/RAID5 등)
- DBMS 서버와 Storage 간의 Network 구성 및 속도

### 2) Oracle DBMS 분석

- 보유한 Tablespace 개수 및 사이즈
- 보유한 Table 개수 및 사이즈
- 보유한 Object 개수 및 사이즈
- 사용 중인 Physical Memory / Swap 사이즈
- 전체 Query 중 Function/Procedure의 비중

### 3) 비즈니스 분석

- 온라인성 업무와 배치성 업무의 비율 구성
- 배치성 업무의 Query 복잡도 (Join, Group By, Order By 등)
- 배치성 업무의 Query 중 동시에 수행 되는 수

## 1.2 환경설정

Oracle에서 ALTIBASE HDB로 변환 시 고려되어야 할 환경 설정에 대한 내용입니다.

### 가. OBJECT 변환

#### 1) 데이터 타입

| 분류        | Oracle                         | ALTIBASE          | 비고   |
|-----------|--------------------------------|-------------------|--|
| 문자<br>타입  | CHAR                           | CHAR              | 최대 32K   |
|           | VARCHAR2, VARCHAR              | VARCHAR2, VARCHAR | 최대 32K. DESC로 조회 시 VARCHAR로 조회   |
|           | NCHAR                          | NCHAR             | 문자 길이 최대16000B(UTF16), 문자 길이 최대10666B(UTF8)  |
|           | NVARCHAR2                      | NVARCHAR          | 문자 길이 최대16000B(UTF16), 문자 길이 최대 10666B(UTF8)   |
|           | LONG                           | CLOB              | 최대 2G  |
| LOB<br>타입 | BLOB                           | BLOB              | 최대 2G  |
|           | CLOB                           | CLOB              | 최대 2G  |
|           | NCLOB                          | CLOB              | 최대 2G  |
| 숫자<br>타입  | NUMERIC(p, s)                  | NUMERIC(p, s)     | 데이터가 SMALLINT, INTEGER, BIGINT, REAL, DOUBLE 등 native type으로 지정 가능 하다면 native type으로 지정하는 것이 좋다. 데이터 처리시 변환 비용에 따른 Overhead를 줄일 수 있고, 저장 공간의 효율성이 좋아지기 때문이다. |
|           | NUMBER (p, s)                  | NUMBER(p, s)      |  |
|           | DECIMAL(p, s)                  | DECIMAL(p, s)     |  |
|           | FLOAT(p), BINARY_FLOAT         | FLOAT(p)          |  |
|           | SMALLINT                       | SMALLINT          | 2 Byte 정수형 타입  |
|           | INT                            | INTEGER           | 4 Byte 정수형 타입  |
|           | REAL                           | REAL              | 4 Byte 실수형 타입  |
|           | BINARY_DOUBLE                  | DOUBLE            | 8 Byte 실수형 타입  |
| 날짜<br>타입  | DATE                           | DATE              | ALTIBASE HDB DATE 타입은 ORACLE DATE 타입의 표현 범위를 포함  |
|           | INTERVAL YEAR TO MONTH         | -                 | 지원 안됨  |
|           | INTERVAL DAY TO SECOND         | -                 |  |
|           | TIMESTAMP WITH TIME ZONE       | -                 |  |
|           | TIMESTAMP WITH LOCAL TIME ZONE | -                 |  |
|           | TIMESTAMP                      | TIMESATAMP        |  |

|          |            |      |       |
|----------|------------|------|-------|
| 이진<br>타입 | BFILE      | BLOB | 최대 2G |
|          | RAW (size) | BLOB |       |
|          | LONG RAW   | BLOB |       |

## 2) Object 비교

| Oracle                     | ALTIBASE  |
|----------------------------|---|
| CLUSTER                    | 지원하지 않음   |
| CONSTRAINT                 | 지원  |
| DATABASE LINK              | 지원  |
| DATABASE TRIGGER           | 지원  |
| DIMENSION                  | 지원하지 않음   |
| EXTERNAL PROCEDURE LIBRARY | 지원(C/C++만 가능)   |
| INDEX-ORGANIZED TABLE      | 지원하지 않음   |
| INDEX                      | B-TREE, R-TREE, Function based 지원(BITMAP, CLUSTER, Global Partitioned INDEX는 지원하지 않음) |
| INDEXTYPE                  | 지원하지 않음   |
| JAVA 관련 객체                 | 지원하지 않음   |
| MATERIALIZED VIEW          | 지원  |
| MATERIALIZED VIEW LOG      | 지원하지 않음   |
| OBJECT TABLE               | 지원하지 않음   |
| OBJECT TYPE                | 지원하지 않음   |
| OBJECT VIEW                | 지원하지 않음   |
| OPERATOR                   | 지원하지 않음   |
| PACKAGE                    | 지원  |
| SEQUENCE                   | 지원  |
| STORED FUNCTION/PROCEDURE  | 지원  |
| SYNONYM                    | 지원  |
| TABLE                      | 지원  |
| VIEW                       | 지원  |
| CONTEXT                    | 지원하지 않음   |
| DIRECTORY                  | 지원  |
| PARAMETER FILE             | 객체로서 지원하지 않음. altibase.properties 파일로 지원  |
| PROFILE                    | 지원하지 않음   |
| ROLE                       | 지원  |
| TABSPACE                   | MEMORY, DISK, VOLATILE, TEMPORARY, UNDO 지원  |
| USER                       | 지원  |



### 3) CREATE TABLESPACE

Oracle의 Data Tablespace는 모두 Disk Tablespace입니다. ALTIBASE HDB 전환 시 CREATE DISK TABLESPACE 구문 을 이용하여 Disk Tablespace를 생성해야 합니다.

#### 3.1 DATA TABLESPACE

| Oracle                               | ALTIBASE                       | 비고                                    |
|--------------------------------------|--------------------------------|---------------------------------------|
| BIGFILE SMALLFILE                    | -                              | ALTIBASE HDB는 지원하지 않으므로 변환 시 해당 옵션 삭제 |
| DATAFILE File Specification          | DATAFILE File Specification    |                                       |
| MINIMUM EXTENT                       | -                              | ALTIBASE HDB는 지원하지 않으므로 변환 시 해당 옵션 삭제 |
| BLOCKSIZE                            | -                              |                                       |
| LOGGING NOLOGGING                    | -                              |                                       |
| FORCE LOGGING                        | -                              |                                       |
| DEFAULT Storage 구문                   | -                              |                                       |
| ONLINE OFFLINE                       | -                              |                                       |
| EXTENT MANAGEMENT LOCAL DICTIONARY   | -                              |                                       |
| SEGMENT SPACE MANAGEMENT AUTO MANUAL | SEGMENT MANAGEMENT AUTO MANUAL |                                       |
| FLASHBACK ON OFF                     | -                              | ALTIBASE HDB는 지원하지 않으므로 변환 시 해당 옵션 삭제 |

#### 3.2 DATA TABLESPACE 예제

| Oracle   | ALTIBASE  |
|--|---|
| CREATE TABLESPACE TESTDATA<br>DATAFILE 'testdata01.dbf'<br>SIZE 1024M<br>AUTOEXTEND ON<br>NEXT 50M<br>MAXSIZE UNLIMITED<br>LOGGING<br>ONLINE<br>EXTENT MANAGEMENT LOCAL AUTOALLOCATE<br>BLOCKSIZE 8K<br>SEGMENT SPACE MANAGEMENT MANUAL<br>FLASHBACK ON; | CREATE TABLESPACE TESTDATA<br>DATAFILE 'testdata01.dbf'<br>SIZE 1024M<br>AUTOEXTEND ON<br>NEXT 50M<br>MAXSIZE UNLIMITED<br><br>SEGMENT MANAGEMENT MANUAL; |

- ✓ SEGMENT 절은 SEGMENT MANAGEMENT AUTO|MANUAL 절로 수정하고 그 외의 모든 절은 지원하지 않으므로 삭제

### 3.3 TEMPORARY TABLESPACE

| Oracle                             | ALTIBASE | 비고                                    |
|------------------------------------|----------|---------------------------------------|
| TABLESPACE GROUP                   | -        | ALTIBASE HDB는 지원하지 않으므로 변환 시 해당 옵션 삭제 |
| EXTENT MANAGEMENT LOCAL DICTIONARY | -        |                                       |

### 3.4 TEMPORARY TABLESPACE 예제

| Oracle  | ALTIBASE  |
|---|---|
| <pre>CREATE TEMPORARY TABLESPACE tbs_temp_02 TEMPFILE 'temp02.dbf' SIZE 5M AUTOEXTEND ON TABLESPACE GROUP tbs_grp_01;</pre> | <pre>CREATE TEMPORARY TABLESPACE tbs_temp_02 TEMPFILE 'temp02.dbf' SIZE 5M AUTOEXTEND ON;</pre> |

✓ TABLESPACE GROUP 절은 지원하지 않으므로 삭제

### 3.5 UNDO TABLESPACE

ALTIBASE는 시스템에 의해 자동으로 Undo Tablespace를 관리 합니다. Undo Tablespace에 데이터 파일의 추가 및 크기 변경만 가능합니다.

| Oracle  | ALTIBASE   |
|---|--|
| <pre>CREATE UNDO TABLESPACE UNDOTBS2 DATAFILE 'undotbs2.dbf' SIZE 500M AUTOEXTEND ON NEXT 5M MAXSIZE UNLIMITED;</pre> | <pre>ALTER TABLESPACE SYS_TBS_DISK_UNDO ADD DATAFILE 'undotbs2.dbf' SIZE 500M AUTOEXTEND ON NEXT 5M MAXSIZE UNLIMITED;</pre> |

## 4) CREATE TABLE

ALTIBASE HDB는 Temporary Table, Object Table, XMLType Table을 제공하지 않습니다. ALTIBASE HDB는 Memory Table을 제공합니다. 기존의 Table 의 특성을 파악하여 Memory Table 또는 Disk Table로 분리할 수 있습니다. 그러나 Memory Table 생성 시 Oracle에서 사용했던 CREATE TABLE 구문에 사용한 옵션들은 사용할 수 없습니다. 또한 TABLE 생성 시 Segment 관련 내용을 지정할 경우에는 TABLESPACE 지정 -> PCTFREE/PCTUSED 지정 -> INITRANS/MAXTRANS 지정 -> Storage 절 -> Logging 절 순으로 지정해 줘야 합니다.

#### 4.1 COLUMN DEFINITION 절

| Oracle                         | ALTIBASE                         | 비고   |
|--------------------------------|----------------------------------|--|
| SORT                           | -                                | ALTIBASE HDB는 지원하지 않으므로 변환 시 해당 옵션 삭제  |
| DEFAULT                        | DEFAULT                          |  |
| ENCRYPT                        | ENCRYPT                          |  |
| Constraint 구문                  | Constraint 구문                    | ALTIBASE HDB는 Constraint 지정 시 ENABLE/DISABLE 옵션, References 절의 ON DELETE SET NULL(ON DELETE CASCADE는 지원) 옵션을 제공하지 않으므로 삭제.<br>ALTIBASE HDB는 PRIMARY KEY, UNIQUE 지정 시 Using Index 절에는 Tablespace 절, Parallel 절, Logging 절, Force 절만 지정 가능하다. 즉 Index 이름 및 CREATE INDEX 절은 제공하지 않는다. |
| Ref Constraint 구문              | -                                | ALTIBASE HDB는 REF컬럼을 지원하지 않으므로 변환 시 해당 옵션 삭제   |
| ORGANIZATION                   | -                                | ALTIBASE HDB는 지원하지 않으므로 변환 시 해당 옵션 삭제  |
| CLUSTER                        | -                                |  |
| COMPRESS NOCOMPRESS            | COMPRESS NOCOMPRESS              |  |
| Column의 크기 지정 시 Byte 키워드 명시 가능 | Column의 크기 지정 시 Byte 키워드 제공하지 않음 | Oracle은 Column의 크기 지정 시 Byte 키워드를 명시할 수 있지만, ALTIBASE HDB는 크기 지정 시 Byte를 제공하지 않음.<br>ex) ORACLE : c1 VARCHAR2(10 Byte) => ALTIBASE : c1 VARCHAR2(10)   |

- ✓ PRIMARY KEY, UNIQUE Constraint을 지정할 때 USING INDEX 절을 이용하여 INDEX 속성을 지정할 때 ALTIBASE HDB는 TABLESPACE 절, PARALLEL/NOPARALLEL 절, LOGGING/NOLOGGING 절만 지정이 가능합니다. 즉 storage 관련 속성은 지정할 수 없습니다.

#### 4.2 SEGMENT ATTRIBUTES 절

| Oracle     | ALTIBASE   | 비고 |
|------------|------------|----|
| TABLESPACE | TABLESPACE |    |
| PCTFREE    | PCTFREE    |    |
| PCTUSED    | PCTUSED    |    |

|                   |                   |   |
|-------------------|-------------------|---|
| INITRANS          | INITRANS          |   |
| MAXTRANS          | MAXTRANS          | 255를 120으로 변경한다.Oracle의 MAXTRANS는 Deprecate 되었고, 항상 그 값은 255이다. 반면에 ALTIBASE의 MAXTRANS 값을 최대 120까지 지정할 수 있기 때문에 120으로 변경한다. |
| LOGGING NOLOGGING | LOGGING NOLOGGING |   |

#### 4.3 STORAGE 절

| Oracle      | ALTIBASE     | 비고                                    |
|-------------|--------------|---------------------------------------|
| INITIAL     | INITEXTENTS  | Bytes -> extent 개수로 변경                |
| NEXT        | NEXTTEXTENTS | Bytes -> extent 개수로 변경                |
| MINEXTENTS  | MINEXTENTS   |                                       |
| MAXEXTENTS  | MAXEXTENTS   |                                       |
| PCTINCREASE | -            | ALTIBASE HDB는 지원하지 않으므로 변환 시 해당 옵션 삭제 |
| FREELISTS   | -            |                                       |
| FREELIST    | -            |                                       |
| OPTIMAL     | -            |                                       |
| BUFFER POOL | -            |                                       |

#### 4.4 LOB STORAGE 절

| Oracle     | ALTIBASE   | 비고  |
|------------|------------|---|
| TABLESPACE | TABLESPACE | ALTIBASE의 LOB STORAGE절은 TABLESPACE만 지정 가능 |
| STORAGE    | -          | ALTIBASE는 지원하지 않으므로 변환 시 해당 옵션 삭제         |
| CHUNK      | -          |   |
| PCTVERSION | -          |   |
| RETENTION  | -          |   |
| FREEPOOLS  | -          |   |

|                   |                   |  |
|-------------------|-------------------|--|
| CACHE             | -                 |  |
| STORAGE IN ROW    | -                 |  |
| LOGGING NOLOGGING | LOGGING NOLOGGING |  |

#### 4.5 TABLE PARTITION 절

| Oracle                    | ALTIBASE           | 비고                                   |
|---------------------------|--------------------|--------------------------------------|
| PARTITION BY RANGE        | PARTITION BY RANGE |                                      |
| PARTITION BY HASH         | PARTITION BY HASH  |                                      |
| PARTITION BY LIST         | PARTITION BY LIST  |                                      |
| Composite partitioning 구문 | -                  | ALTIBASE는 지원하지 않으므로<br>변환 시 해당 옵션 삭제 |

#### 4.6 TABLE PROPERTIES 절

| Oracle                             | ALTIBASE                    | 비고                                   |
|------------------------------------|-----------------------------|--------------------------------------|
| ENABLE DISABLE ROW MOVEMENT        | ENABLE DISABLE ROW MOVEMENT | Partitioned TABLE에만 지원               |
| NOPARALLEL PARALLEL                | NOPARALLEL PARALLEL         |                                      |
| ENABLE DISABLE VALIDATE NOVALIDATE | -                           | ALTIBASE는 지원하지 않으므로<br>변환 시 해당 옵션 삭제 |

#### 4.7 TABLE 변환 예제

| Oracle   | ALTIBASE   |
|--|--|
| <pre>CREATE TABLE "SCOTT"."EMP" (   "EMPNO" NUMBER(4,0),   "ENAME" VARCHAR2(10 Byte),   "JOB" VARCHAR2(9 Byte),   "MGR" NUMBER(4,0),   "HIREDATE" DATE,   "SAL" NUMBER(7,2),   "COMM" NUMBER(7,2),   "DEPTNO" NUMBER(2,0),   CONSTRAINT "PK_EMP" PRIMARY KEY</pre> | <pre>CREATE TABLE "SCOTT"."EMP" (   "EMPNO" NUMBER(4,0),   "ENAME" VARCHAR2(10),   "JOB" VARCHAR2(9),   "MGR" NUMBER(4,0),   "HIREDATE" DATE,   "SAL" NUMBER(7,2),   "COMM" NUMBER(7,2),   "DEPTNO" NUMBER(2,0),   CONSTRAINT "PK_EMP" PRIMARY KEY</pre> |

|   |  |
|---|--|
| <pre> ("EMPNO") USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)  TABLESPACE "USERS" ENABLE,          CONSTRAINT "FK_DEPTNO" FOREIGN KEY ("DEPTNO")          REFERENCES "SCOTT"."DEPT" ("DEPTNO") ENABLE ) PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255 NOCOMPRESS LOGGING  STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645  PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)  TABLESPACE "USERS" ; </pre> | <pre> ("EMPNO") USING INDEX TABLESPACE "USERS" ,          CONSTRAINT "FK_DEPTNO" FOREIGN KEY ("DEPTNO")          REFERENCES "SCOTT"."DEPT" ("DEPTNO") )  TABLESPACE "USERS"  PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 120  STORAGE(INITEXTENTS 1 NEXTEXTENTS 1 MINEXTENTS 1 MAXEXTENTS 2147483645)  LOGGING ;  "ENAME" VARCHAR2(10 Byte)Byte는 삭제  USING INDEX TABLESPACE "USERS"          PK 지정 시 USING INDEX 절에는 TABLESPACE 절,         PARALLEL/NOPARALLEL 절,         LOGGING/NOLOGGING 절만 지정 가능하므로         TABLESPACE 절 이외의 다른 옵션은 삭제          REFERENCES "SCOTT"."DEPT" ("DEPTNO")          FK 지정 시 ENABLE 옵션은 지원하지 않으므로 삭제  PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 120          MAXTRANS의 최대 값은 120 이므로 120으로 수정  STORAGE(INITEXTENTS 1 NEXTEXTENTS 1 MINEXTENTS 1 MAXEXTENTS 2147483645)          INITIAL과 NEXT는 각각 INITEXTENTS,         NEXTEXTENTS로 변경해야 하며 그 값도 extent 의         개수로 수정  TABLESPACE "USERS"          PCTFREE 10 PCTUSED 40 INITRANS 1         MAXTRANS 120          STORAGE...          LOGGING  segment 관련 내용을 지정할 경우에는 TABLESPACE 지정 -&gt; PCTFREE/PCTUSED 지정 -&gt; INITRANS/MAXTRANS 지정 -&gt; Storage 절 -&gt; logging 절 순으로 지정 </pre> |
|---|--|

## 5) CREATE INDEX

ALTIBASE는 BTREE와 RTREE INDEX만 제공하고 BITMAP, CLUSTER, Function based, REVERSE,

Global partitioned INDEX를 제공하지 않습니다. 또한, ALTIBASE INDEX 생성 시 segment 관련 내용 지정 할 경우에는 TABLESPACE 지정 -> PARALLEL/ NOPARALLEL 지정 -> LOGGING/NOLOGGING 지정 -> storage 절 순으로 지정해야 합니다.

| Oracle              | ALTIBASE              | 비고                                   |
|---------------------|-----------------------|--------------------------------------|
| TABLESPACE          | TABLESPACE            |                                      |
| LOGGING NOLOGGING   | LOGGING NOLOGGING     |                                      |
| NOPARALLEL PARALLEL | NOPARALLEL PARALLEL   |                                      |
| COMPUTE STATISTICS  | -                     | ALTIBASE는 지원하지 않으므로<br>변환 시 해당 옵션 삭제 |
| REVERSE             | -                     |                                      |
| SORT NOSORT         | -                     |                                      |
| ONLINE              | -                     |                                      |
| COMPRESS NOCOMPRESS | -                     |                                      |
| PCTFREE, PCTUSED,   | -                     |                                      |
| INITRANS            | INITRANS              |                                      |
| MAXTRANS            | MAXTRANS              | ALTIBASE는 최대 30까지 지원                 |
| Storage 구문          | TABLE의 Storage 구문과 동일 |                                      |

## 5.1 INDEX 변환 예제

| Oracle   | ALTIBASE  |
|--|---|
| CREATE INDEX "SCOTT"."EMP_IDX1" ON<br>"SCOTT"."EMP" ("DEPTNO", "SAL")<br><br><b>PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE<br/>           STATISTICS</b><br><br>STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS<br>1 MAXEXTENTS 2147483645<br><br><b>PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1<br/>           BUFFER_POOL DEFAULT)</b><br><br>TABLESPACE "USERS" ; | CREATE INDEX "SCOTT"."EMP_IDX1" ON<br>"SCOTT"."EMP" ("DEPTNO", "SAL")<br><br>TABLESPACE "USERS"<br><br>INITRANS 2 MAXTRANS 30<br><br>STORAGE(INITEXTENTS 1 NEXTEXTENTS 1<br>MINEXTENTS 1 MAXEXTENTS 2147483645) ;<br><br>INITRANS 2 MAXTRANS 30<br><br>MAXTRANS는 30까지 지원하므로 30으로 변경<br><br>STORAGE(INITEXTENTS 1 NEXTEXTENTS 1<br>MINEXTENTS 1 MAXEXTENTS 2147483645) ;<br><br>STORAGE 절에는 INITEXTENTS, NEXTEXTENTS,<br>MINEXTENTS, MAXEXTENTS 만 지정가능 하고,<br>INITIAL과 NEXT는 각각 INITEXTENTS,<br>NEXTEXTENTS로 변경해야 하며 그 값도 extent 의<br>개수로 수정해야 함 |

|  |   |
|--|---|
|  | <p>TABSPACE "USERS"</p> <p>INITRANS 2 MAXTRANS 30</p> <p>STORAGE ... ;</p> <p>segment 관련 내용 지정할 경우에는 TABLESPACE 지정<br/>-&gt; PARALLEL/NOPARALLEL 지정 -&gt;<br/>LOGGING/NOLOGGING 지정 -&gt; storage 절 순으로 지정</p> |
|--|---|

## 6) CREATE VIEW

ALTIBASE의 VIEW는 Oracle의 VIEW 생성 구문과 동일하게 CREATE OR REPLACE VIEW 구문으로 생성합니다.

| Oracle            | ALTIBASE       | 비고   |
|-------------------|----------------|--|
| WITH READ ONLY    | WITH READ ONLY | ALTIBASE는 WITH READ ONLY 옵션의 VIEW만 제공하므로 해당 옵션이 default임 |
| [NO] FORCE        | [NO] FORCE     |  |
| WITH CHECK OPTION | -              | ALTIBASE는 지원하지 않으므로 변환 시 해당옵션 삭제                         |
| XMLType view 구문   | -              |  |
| Object view 구문    | -              |  |

## 7) CREATE TRIGGER

ALTIBASE의 TRIGGER는 Oracle의 TRIGGER 생성 구문과 동일하게 CREATE OR REPLACE TRIGGER 구문으로 생성합니다. 이중화로 인해 반영되는 데이بل 데이터의 변경은 트리거 동작을 발생시키지 않습니다. ALTIBASE는 TRIGGER 대상 TABLE에 LOB 컬럼을 포함할 수 없습니다.

| Oracle                    | ALTIBASE                  | 비고                             |
|---------------------------|---------------------------|--------------------------------|
| CREATE OR REPLACE TRIGGER | CREATE OR REPLACE TRIGGER |                                |
| BEFORE AFTER INSTEAD OF   | BEFORE AFTER              | ALTIBASE는 INSTEAD OF는 지원하지 않음  |
| DML 이벤트 구문                | DML 이벤트 구문                | DML 이벤트 구문은 Oracle과 동일         |
| DDL 이벤트 구문                | -                         | ALTIBASE는 DDL TRIGGER를 제공하지 않음 |
| WHEN 조건                   | WHEN 조건                   |                                |



| FOR EACH ROW    | FOR EACH ROW    |  |
|-----------------|-----------------|--|
| REFERENCING     | REFERENCING     | DELETE 이벤트에서 NEW는 REFERENCING 할 수 없고, INSERT 이벤트에서는 OLD는 REFERENCING 할 수 없다. OLD/NEW에 alias를 줄 경우 OLD/NEW는 키워드이므로 alias로 사용할 수 없다.   |
| Trigger body 구문 | Trigger body 구문 | ALTIBASE Trigger body 구문은 AS BEGIN 구문으로 시작해야 한다. 또한 Oracle의 Trigger body 구문은 DECLARE 절로 시작할 수 있지만, ALTIBASE의 Trigger body는 AS 절에 선언 부분을 지정해야 하고, DECLARE 절을 명시할 수 없다.<br><br>OLD/NEW row 사용 시 ':' 사용할 수 없다. Ex) :old (x) -> old(o) |

## 8) CREATE SEQUENCE

ALTIBASE의 CREATE SEQUENCE 구문은 Oracle과 동일합니다. 다만, Oracle에서 제공하는 ORDER, NOORDER 옵션은 지원하지 않습니다

또한, ORACLE SEQUENCE의 maxvalue 값은 28자리 정수까지 지정할 수 있지만, ALTIBASE SEQUENCE의 maxvalue는 (-9223372036854775807)부터 9223372036854775806까지의 범위 내에서 지정할 수 있습니다.

## 9) CREATE SYNONYM

ALTIBASE의 CREATE SYNONYM 구문은 Oracle과 동일합니다.

## 10) ALTER TABLE

ALTIBASE는 Constraint 추가 시 한번에 1개의 Constraint만 추가 가능합니다.

Oracle에서 “ALTER TABLE ADD (CONSTRAINT constraint\_name constraint\_type,...);” 문으로 여러 개의 Constraint를 추가하는 문장은 Constraint별로 나눠서 “ALTER TABLE ADD CONSTRAINT constraint\_name constraint\_type;” 문으로 Constraint 별로 나눠서 실행해야 합니다.

Oracle은 PRIMARY KEY와 UNIQUE를 지정 시 INDEX를 미리 생성한 후 지정이 가능하지만, ALTIBASE는 Constraint를 지정하는 시점에 내부적으로 INDEX를 생성하기 때문에 동일 컬럼에 대해 PK Constraint,

UNIQUE Constraint, INDEX 생성 중 하나만 가능합니다. 즉, INDEX를 생성한 컬럼에 PK Constraint를 지정할 수 없습니다.

## 나. SQL 변환

### 1) Math Function

| Oracle     | ALTIBASE   | 비고  |
|------------|------------|---|
| ABS        | ABS        | 절대값을 반환   |
| ACOS       | ACOS       | n의 역코사인(arc cosine)   |
| ASIN       | ASIN       | n의 역사인(arc sine)  |
| ATAN       | ATAN       | n의 역탄젠트(arc tangent)  |
| ATAN2      | ATAN2      | ATAN2(n,m)은 n/m의 역탄젠트(arc tangent)값을 반환   |
| AVG        | AVG        | Row의expr 값의 평균값을 구한다  |
| BITAND     | BITAND     | 인수1과 인수2의 비트에 대한 and 연산을 수행하여 정수를 반환  |
| CEIL       | CEIL       | 인수에서 지정한 수치를 올림 하여 정수를 반환   |
| COS        | COS        | n의 코사인값을 반환   |
| COSH       | COSH       | n의 쌍곡 코사인 값을반환  |
| COUNT      | COUNT      | Row의 개수를 세는 함수, 분석함수이다  |
| DENSE_RANK | DENSE_RANK | DENSE_RANK 함수는 RANK 함수 처럼 결과 집합 또는 파티션의 특정 멤버를 기준으로 순위를 매긴다. 그러나 중복 순위 발생 후에 건너뛰지 않고 다음 순위가 매겨진다. 반환 값의 타입은 |

|            |            |   |
|------------|------------|---|
|            |            | BIGINT 이다.  |
| EXP        | EXP        | e의n제곱 값을 반환   |
| FLOOR      | FLOOR      | 지정한 숫자보다 작거나 같은 가장 큰 정수 반환  |
| LAG        | LAG        | 정렬된 각 파티션 내에서 현재 행을 기준으로 이전의 offset 번째 행의 값을 구하는 함수이다   |
| LN         | LN         | 입력한 자연 로그값 구한다  |
| LOG(N,M)   | LOG(N,M)   | 밑을 n으로 하는 m의 로그값을 반환  |
| MOD        | MOD        | n2을 n1으로 나눈 나머지 값을 반환   |
| NANVL      | N/A        | 입력 값n2가 수치가 아니면 대체값n1을 반환   |
| NTH_VALUE  | NTH_VALUE  | 파티션 또는 윈도우 내에서 offset 번째 행의 값을 구하는 함수이다   |
| POWER      | POWER      | n2의 1n1승 값을 반환  |
| ROUND      | ROUND      | n값을 소수점 이하를 integer를 기준으로 반올림하여 반환  |
| ROW_NUMBER | ROW_NUMBER | ROW_NUMBER 함수는 결과 집합 또는 파티션의 특정 멤버를 기준으로 연속적인 유일한 수를 부여한다. 1 부터 시작하여 ORDER BY 식에 명시된 순서대로 부여된다. |
| SIGN       | SIGN       | n의 부호를 반환   |
| SIN        | SIN        | n의 사인값을 반환  |
| SINH       | SINH       | n의 쌍곡선 사인을 반환   |
| SQRT       | SQRT       | n의 제곱근을 반환  |
| STDDEV     | STDDEV     | STDDEV 는 입력된 expression 들의 표준편차를 반환한다.  |
| TAN        | TAN        | N의탄젠트 값을반환  |

|          |          |   |
|----------|----------|---|
| TANH     | TANH     | N의쌍곡선탄젠트(hyperbolic tangent)값 반환        |
| VARIANCE | VARIANCE | VARIANCE 는 입력된 expression 들의 변동량을 반환한다. |

## 2) String Function

| Oracle  | ALTIBASE | 비고  |
|---------|----------|---|
| CONCAT  | CONCAT   | char1과 char2를 연결하여 반환   |
| CHR     | CHR      | number에 해당하는 문자를 구한다.   |
| NCHR    | NCHR     | 유니코드 문자를 변환한다   |
| LOWER   | LOWER    | 입력된 문자열을 소문자로 변환한다  |
| UPPER   | UPPER    | 입력된 문자열을 대문자로 반환  |
| LPAD    | LPAD     | 지정된 자릿수N으로부터EXPR1을 채우고, 왼편에 남은 공간에 expr1을 채운다                   |
| LTRIM   | LTRIM    | 문자열 char 좌측으로부터set 으로 지정된 모든 문자를 제거한다.                          |
| RTRIM   | RTRIM    | 문자열 char 우측으로부터set 으로 지정된 모든 문자를 제거한다.                          |
| RPAD    | RPAD     | 인수 expr1 오른쪽으로 인수 expr2로 지정한 문자를 길이 필요에 따라 반복하여n만큼 붙여준다         |
| RPAD    | RPAD     | 인수 expr1 오른쪽으로 공백을 채워준다   |
| SOUNDEX | N/A      | char의 음성표현을 가지는 문자열을 반환   |
| SUBSTR  | SUBSTR   | 문자열 Char에서Position 문자위치로부터Substring_Length 문자 길이만큼 문자열을 추출하여 반환 |

|                       |              |  |
|-----------------------|--------------|--|
| REPLACE               | REPLACE2     | Parameter로 주어지는 첫번째 문자열에서, 두번째 문자열을 모두 세번째 문자열로바꾼 후 결과를 반환 |
| REGEXP_REPLACE/SUBSTR | N/A          | 지정한 정규 표현을 만족하는 부분을 치환, 반환                                 |
| INITCAP               | INITCAP      | 입력 문자열 중에서 각 단어의 첫 글자를 대문자로 변환한다                           |
| NLS_INITCAP           | N/A          | 입력문자열 중에서 각 단어의 첫 글자를 대문자로 변환한다(다국언어 지원)한다                 |
| NLSSORT               | N/A          | 입력 문자열을 sorting하여 string을 반환                               |
| TRANSLATE             | TRANSLATE    | from_string에서 각 문자를 to_string 안의 대응하는 문자로 치환하여 expr을 반환    |
| TO_CHAR               | TO_CHAR      | CHAR 타입으로 반환   |
| TREAT                 | N/A          | 인수의 선언형을 변경해준다   |
| NLS_CHARSET_DECL_LEN  | N/A          | NCHR열의 선언된 폭을 반환   |
| NLS_CHARSET_ID        | N/A          | 문자셋 이름에 상응하는 ID번호를 반환                                      |
| ASCII                 | ASCII        | 주어진 char의 첫 문자의 아스키 값에 상응하는10진수 값을 반환                      |
| INSTR                 | INSTR        | 문자열 중에서 지정한 문자가 처음 나타나는 위치를 숫자로 반환                         |
| LENGTH                | LENGTH       | 인수 char의 길이를 반환  |
| LENGTHB               | LENGTHB      | char대신 바이트 단위로 길이를 계산한다                                    |
| REGEXP_INSTR          | REGEXP_INSTR | 지정한 조건(정규표현)을 만족하는 부분의 최초의 위치를 반환                          |

### 3) Date Function

| Oracle            | ALTIBASE          | 비고  |
|-------------------|-------------------|---|
| (use +)           | (use +)           | 일자 date에서 integer를 더한 날짜가 반환                |
| (use -)           | (use -)           | 일자 Date 에서 integer 를 뺀 날짜가 반환               |
| ADD_MONTHS        | ADD_MONTHS        | 일자 date에 특정 개월 수integer를 더한 값을 반환           |
| CURRENT_DATE      | CURRENT_DATE      | 현재 세션의 날짜 정보를 date 형으로 반환                   |
| CURRENT_TIME      | N/A               | 현재 세션의 시간대를 기준으로 현재 시간을 출력한다.               |
| CURRENT_TIMESTAMP | CURRENT_TIMESTAMP | 현재 세션의 날짜와 시간정보를 반환(* CURRENT_DATE의 alias임) |
| LAST_DAY          | LAST_DAY          | 해당날짜가 속한 달의 마지막 날짜를 반환                      |
| NEW_TIME          | CONV_TIMEZONE     | date,zone1시간대를 zone2 시간대로 출력한다              |
| NEXT_DAY          | NEXT_DAY          | 해당일을 기준으로 명시된 요일의 다음 날짜를 반환                 |
| ROUND             | ROUND             | 지정된 단위로 반올림한 날짜 반환                          |
| TRUNC             | TRUNC             | 지정된 단위로 잘라서 날짜 반환                           |
| EXTRACT(datetime) | EXTRACT(datetime) | 특정 날짜, 시간 값이나 날짜 값 표현식으로부터 지정된 날짜 영역의       |

|                      |                      |   |
|----------------------|----------------------|---|
|                      |                      | 값을 추출 후 반환  |
| LOCALTIMESTAMP       | DB_TIMEZONE          | timestamp의 현재 날짜와 시각을 출력  |
| SYSTIMESTAMP         | SYSTIMESTAMP         | 시스템의 날짜를 반환   |
| MONTHS_BETWEEN       | MONTHS_BETWEEN       | 일자 date1과 date2 사이의 시간대로 출력   |
| TO_CHAR(datetime)    | TO_CHAR(datetime)    | 사용자가 지정한 품을 갖는 varchar2 형식의 Data로 변환  |
| TO_NUMBER(TO_CHAR()) | TO_NUMBER(TO_CHAR()) | 지정된 형식인 number형식으로 변환   |
| TO_DATE              | TO_DATE              | 지정된 형식인 date형식으로 변환   |
| TO_DSINTERVAL        | DATEDIFF로 대체가능       | interval year to month 형태로 변경   |
| SYSDATE              | SYSDATE              | Database가 있는OS의 일자와 시간을 반환  |
| N/A                  | DATENAME             | 입력 date에서 date_field_name에 해당하는 값만 반환                                       |
| N/A                  | DATEDIFF             | enddate에서 startdate를 뺀 값(즉, enddate-startdate)을 date_field_name에 명시한 단위로 반환 |
| N/A                  | DATEPART             | 입력 date에서 date_field_name에 해당하는 값만 반환                                       |
| N/A                  | DATEADD              | date의 date_field_name부분을 number 만큼 증가시켜 그 결과를 반환                            |

#### 4) Compare Function

| Oracle   | ALTIBASE | 비고                           |
|----------|----------|------------------------------|
| GREATEST | GREATEST | 하나 이상의 인수 중에서 가장 큰 값을 반환     |
| LEAST    | LEAST    | 인수 expr의 리스트 중에서 가장 작은 값을 반환 |

|  |  |  |
|--|--|--|
|  |  |  |
|--|--|--|

## 5) Convert Function

| Oracle           | ALTIBASE | 비고   |
|------------------|----------|--|
| CAST             | CAST     | Data형식을 변환                                       |
| CONVERT          | CONVERT  | 지정된 문자 세트를 변환                                    |
| RAWTOHEX         | N/A      | RAW를 16진수의 문자로 변환                                |
| HEXTORAW         | N/A      | 16진수를 raw값으로 변환                                  |
| RAWINTOCHAR      | N/A      | rowid 값을 VARCHAR2형식으로 변환                         |
| SCN TO TIMESTAMP | N/A      | 시스템 변경 번호(SCN) 로 변환                              |
| TO_BINARY_DOUBLE | N/A      | 배정밀도 부동소수점 변환                                    |
| TO_CLOB          | N/A      | nclob값을 clob값으로 변환                               |
| TO_LOB           | N/A      | long또는 long raw 값을 lob값으로 변환                     |
| COLEASE          | COLEASE  | 나열된 값을 순차적으로 체크하여 NULL 이 아닌 첫번째 인수를 반환           |
| LNNVL            | N/A      | 조건의 한쪽 또는 양쪽 연산자가 NULL이 존재할 경우에, 조건문을 평가하기 위한 방법 |
| NULLIF           | NULLIF   | expr1과 expr2 가 같으면, null 값을 반환                   |
| NVL              | NVL      | 쿼리의 결과에서 NULL (공백으로 반환)값을 치환                     |
| NVL2             | NVL2     | expr1이 NULL이 아니면 expr2를 반환하고 NULL 이면 expr3를 반환   |

## 6) Encode/Decode Function



| Oracle | ALTIBASE | 비고                                       |
|--------|----------|--|
| DECODE | DECODE   | 프로그래밍언어의 IF 문을 SQL , PL/SQL 안으로 끌어들이는 함수 |
| DUMP   | DUMP     | 지정한 Data의 위치와 길이등을 지정한 형식으로 반환           |
| VSIZE  | SIZEOF   | expr 의 내부 표현에서 바이트 수를 반환                 |

## 7) Large Object Function

| Oracle                | ALTIBASE | 비고   |
|-----------------------|----------|--|
| EMPTY_BLOB/EMPTY_CLOB | N/A      | LOB변수를 초기화하는 함수  |
| BFILENAME             | N/A      | Server File System의 물리 LOB BinaryFile과 연관된 BFILE Locator를 반환 |

## 8) Analyze Function

| Oracle      | ALTIBASE    | 비고                                 |
|-------------|-------------|------------------------------------|
| MAX         | MAX         | 인수 중에 최대값을 반환                      |
| MIN         | MIN         | 인수 중에 최소값을 반환                      |
| SUM         | SUM         | Row의 expr에 대한 합계                   |
| RANK        | RANK        | 값의 그룹에서 값의 순위                      |
| LEAD        | LEAD        | 현재형 기준으로 이후 값을 참조                  |
| GROUP_ID    | N/A         | 지정된 GROUP BY 결과로부터 중복된 그룹을 구별      |
| FIRST_VALUE | FIRST_VALUE | 파티션 또는 윈도우 내에 첫 번째 행의 값을 구하는 함수이다. |
| LAST_VALUE  | LAST_VALUE  | 파티션 또는 윈도우 내에서 마지막                 |

|  |  |                    |
|--|--|--------------------|
|  |  | 번째 행의 값을 구하는 함수이다. |
|--|--|--------------------|

## 다. STORED PROCEDURE/FUNCTION 변환

### 1) ALTIBASE AUTOCOMMIT 모드에서 Procedure 처리

ALTIBASE가 AUTOCOMMIT 모드일 경우 PROCEDURE/FUNCTION 전체가 하나의 트랜잭션으로 처리되고 PROCEDURE 내에서의 COMMIT/ROLLBACK은 무시됩니다.

PROCEDURE/FUNCTION을 실행한 후에 자동으로 COMMIT됩니다. 따라서 PROCEDURE/FUNCTION에서 DML 문장을 실행한 후 결과가 ALTIBASE와 Oracle이 다를 수 있습니다.

ALTIBASE를 NON-AUTOCOMMIT 모드로 적용했다면 Oracle의 NON-AUTOCOMMIT 모드와 동일한 결과가 나타납니다.

| Oracle   | ALTIBASE   |
|--|--|
| <pre>CREATE OR REPLACE PROCEDURE t1_test( in_t IN INTEGER, in_v IN VARCHAR)ISBEGIN INSERT INTO t1 VALUES(in_t, in_v); ROLLBACK;END;/EXEC t1_test(4, '000004');SQL&gt; SELECT COUNT(*) FROM t1; COUNT(*)- ----- 0</pre> | <pre>CREATE OR REPLACE PROCEDURE t1_test( in_t IN INTEGER, in_v IN VARCHAR(20))ISBEGIN INSERT INTO t1 VALUES(in_t, in_v); ROLLBACK;END;/EXEC t1_test(4, '000004');;SQL&gt; SELECT COUNT(*) FROM t1;COUNT ----- 1</pre> |

### 2) PARAMETER의 TYPE 과 RETURN TYPE

PARAMETER의 TYPE과 RETURN TYPE의 CHAR, VARCHAR는 크기 지정해야 합니다.

ALTIBASE에서 PROCEDURE나 FUNCTION의 PARAMETER 혹은 RETURN TYPE을 Oracle처럼 CHAR, VARCHAR로 선언하면 CHAR(1), VARCHAR(1)과 동일한 의미입니다. 따라서, 하나의 문자가 아니라 문자열을 사용하고자 한다면, 반드시 그 크기를 지정해 줘야 합니다. 만약 지정 해주지 않고 문자열이 2문자 이상이면 다음과 같은 에러가 발생합니다.

|   |
|---|
| ERR-2100D : Invalid length of the data type |
|---|

### 3) 파일 및 출력 처리

ALTIBASE의 파일 및 출력에 관련된 PROCEDURE는 SYSTEM\_ 유저에 자동으로 생성이 되어 있고, PUBLIC SYNONYM으로 정의되어 있기 때문에 사용자는 PROCEDURE 이름만 호출하여 사용할 수 있습니다.

| 구분   | Oracle               | ALTIBASE   |
|------|----------------------|------------|
| 표준출력 | DBMS_OUTPUT.PUT      | PRINT      |
|      | DBMS_OUTPUT.PUT_LINE | PRINTLN    |
| 파일처리 | UTL_FILE.FOPEN       | FOPEN      |
|      | UTL_FILE.FCLOSE      | FCLOSE     |
|      | UTL_FILE.FCLOSE_ALL  | FCLOSE_ALL |
|      | UTL_FILE.FCOPY       | FCOPY      |
|      | UTL_FILE.FFLUSH      | FFLUSH     |
|      | UTL_FILE.FREMOVE     | FREMOVE    |
|      | UTL_FILE.FRENAME     | FRENAME    |
|      | UTL_FILE.GET_LINE    | GET_LINE   |
|      | UTL_FILE.IS_OPEN     | IS_OPEN    |
|      | UTL_FILE.NEW_LINE    | NEW_LINE   |

### 4) REF CURSOR

Oracle의 REF CURSOR는 보통 PACKAGE에 선언을 하고, 이를 PROCEDURE의 OUT PARAMETER로 선언하여 사용합니다. 하지만, ALTIBASE는 PACKAGE를 제공하지 않기 때문에 TYPESET으로 생성하여 사용해야 합니다.

| Oracle  | ALTIBASE   |
|---|--|
| <pre>CREATE OR REPLACE PACKAGE ref_cursor_pkg AS TYPE ref_type IS REF CURSOR; PROCEDURE ref_cursor_pro(v_result OUT ref_type, v_sql IN VARCHAR2);END;/ CREATE OR REPLACE PACKAGE BODY ref_cursor_pkg AS PROCEDURE ref_cursor_pro(v_result OUT ref_type, v_sql IN VARCHAR2) ASBEGIN OPEN v_result FOR v_sql [USING] [Bind Var];END;/</pre> | <pre>CREATE OR REPLACE TYPESET my_typeAS TYPE my_cur IS REF CURSOR;END;/CREATE OR REPLACE PROCEDURE opencursor( v_result OUT my_type.my_cur, v_sql IN VARCHAR(200) )ASBEGIN OPEN y_result FOR v_sql [USING] [Bind Var];END;/</pre> |

### 5) WHERE CURRENT OF 구문

ALTIBASE는 CURSOR를 이용한 WHERE CURRENT OF 구문을 지원하지 않습니다. 다만, 해당 TABLE에 PRIMARY KEY가 있다면 다음과 같이 변경이 가능합니다.

| Oracle   | ALTIBASE  |
|--|---|
| <pre>CREATE OR REPLACE PROCEDURE proc1IS CURSOR emp_list IS SELECT empno FROM employee WHERE empno = 1 FOR UPDATE; BEGIN FOR emplst IN emp_list LOOP UPDATE employee SET empjob = 'SALESMAN' WHERE CURRENT OF emp_list; END LOOP; END; /</pre> | <pre>CREATE OR REPLACE PROCEDURE proc1ASBEGIN DECLARE CURSOR cur1 IS SELECT empno FROM employee WHERE empno = 1; v_empjob VARCHAR(10); v_empno INTEGER; BEGIN OPEN cur1; LOOP FETCH cur1 INTO v_empno, v_empjob; EXIT WHEN cur1%NOTFOUND; UPDATE employee SET empjob = 'SALESMAN' WHERE emp_no = v_empno; //emp_no가 PK이어야 한다. END LOOP; CLOSE cur1; END;END;/</pre> |

## 6) EXCEPTION

Oracle과 ALTIBASE는 Stored PROCEDURE/FUNCTION에서 발생하는 EXCEPTION들을 미리 시스템에서 정의해 놓았습니다.

| Oracle              |         | ALTIBASE            |         |
|---------------------|---------|---------------------|---------|
| SQLERRM             | SQLCODE | SQLERRM             | SQLCODE |
| CURSOR_ALREADY_OPEN | -6530   | CURSOR_ALREADY_OPEN | 201062  |
| DUP_VAL_ON_INDEX    | -1      | DUP_VAL_ON_INDEX    | 201063  |
| INVALID_CURSOR      | -1001   | INVALID_CURSOR      | 201064  |
| INVALID_NUMBER      | -1722   | INVALID_NUMBER      | 201065  |
| NO_DATA_FOUND       | +100    | NO_DATA_FOUND       | 100     |
| PROGRAM_ERROR       | -6501   | PROGRAM_ERROR       | 201067  |
| STORAGE_ERROR       | -6500   | STORAGE_ERROR       | 201068  |
| TIMEOUT_ON_RESOURCE | -51     | TIMEOUT_ON_RESOURCE | 201069  |
| TOO_MANY_ROWS       | -1422   | TOO_MANY_ROWS       | 201070  |
| VALUE_ERROR         | -6502   | VALUE_ERROR         | 201071  |
| ZERO_DIVIDE         | -1476   | ZERO_DIVIDE         | 201072  |
| ACCESS_INTO_NULL    | -6530   | 지원하지 않음             |         |
| CASE_NOT_FOUND      | -6592   |                     |         |
| COLLECTION_IS_NULL  | -6531   |                     |         |
| LOGIN_DENIED        | -1017   |                     |         |

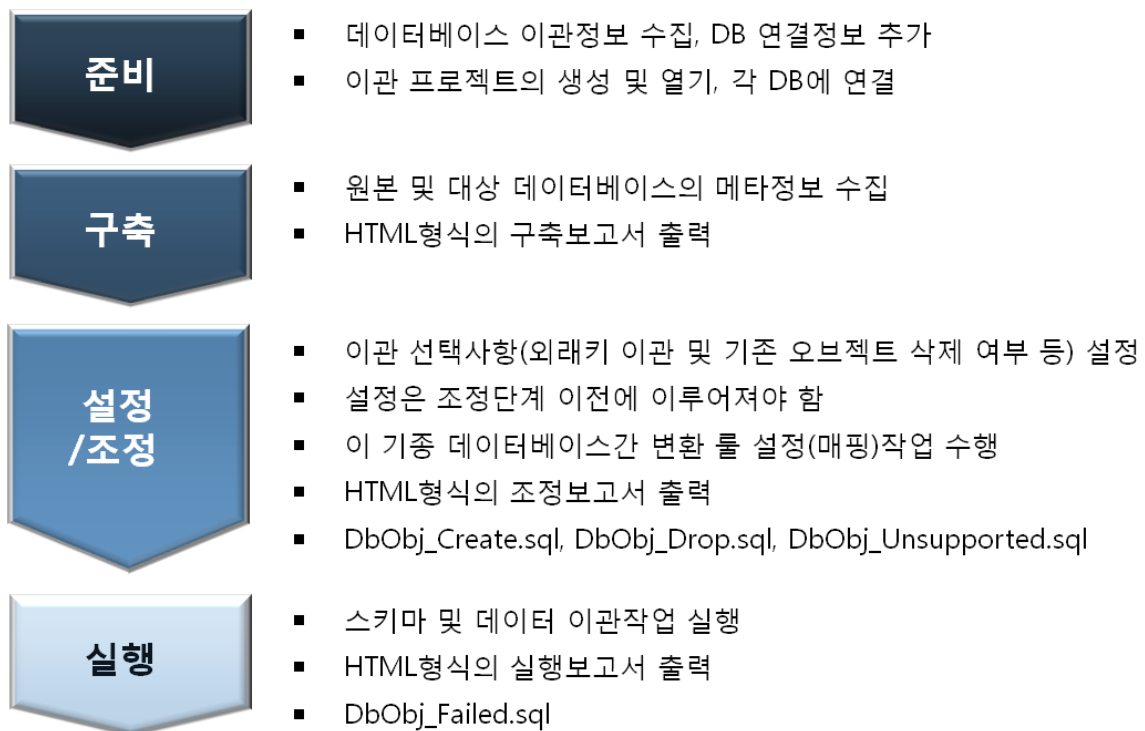
|                         |        |  |
|-------------------------|--------|--|
| NOT_LOGGED_ON           | -1012  |  |
| ROWTYPE_MISMATCH        | -6504  |  |
| SELF_IS_NULL            | -30625 |  |
| SUBSCRIPT_BEYOND_COUNT  | -6533  |  |
| SUBSCRIPT_OUTSIDE_LIMIT | -6532  |  |
| SYS_INVALID_ROWID       | -1410  |  |

## 1.3 전환절차

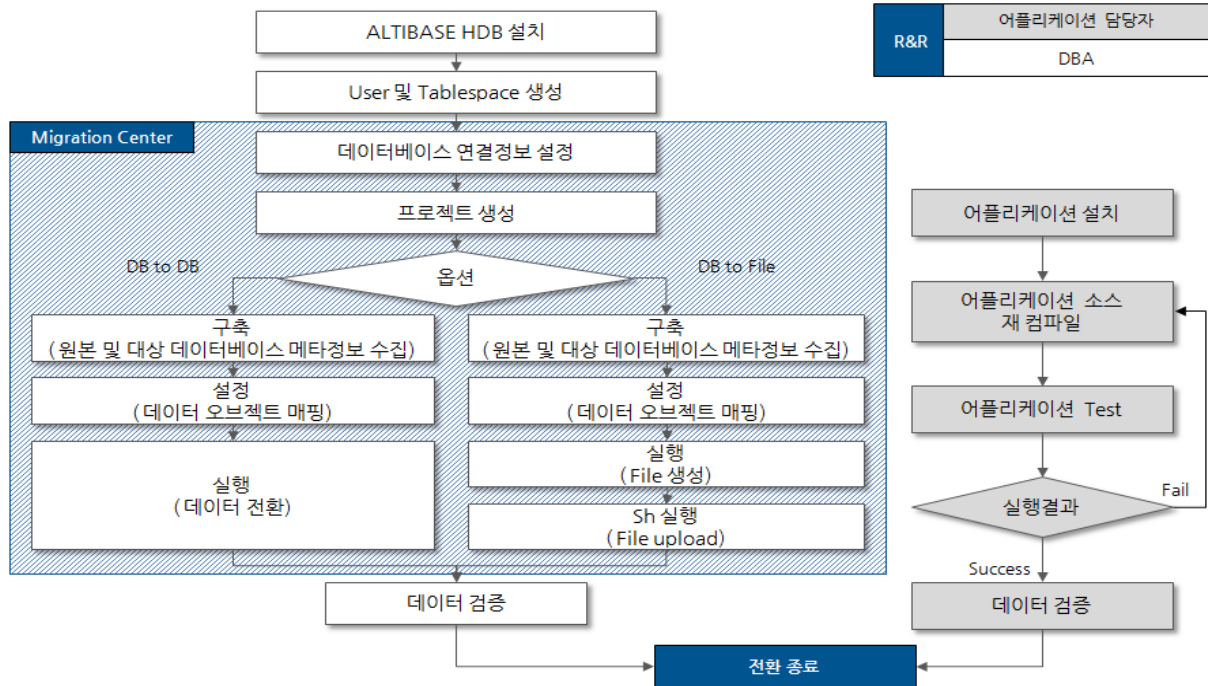
데이터의 전환 작업을 위해 자동화된 Migration Center라는 도구를 사용합니다.

전환 작업이 진행되기 이전에 맵핑 작업을 위하여 이전 DBMS에서 사용하는 환경에 맞게 user생성 과 Tablespace 생성이 필요합니다.

특정 DB 종속적인 기능이 적용된 경우에는 수동 변환 작업을 수행합니다.



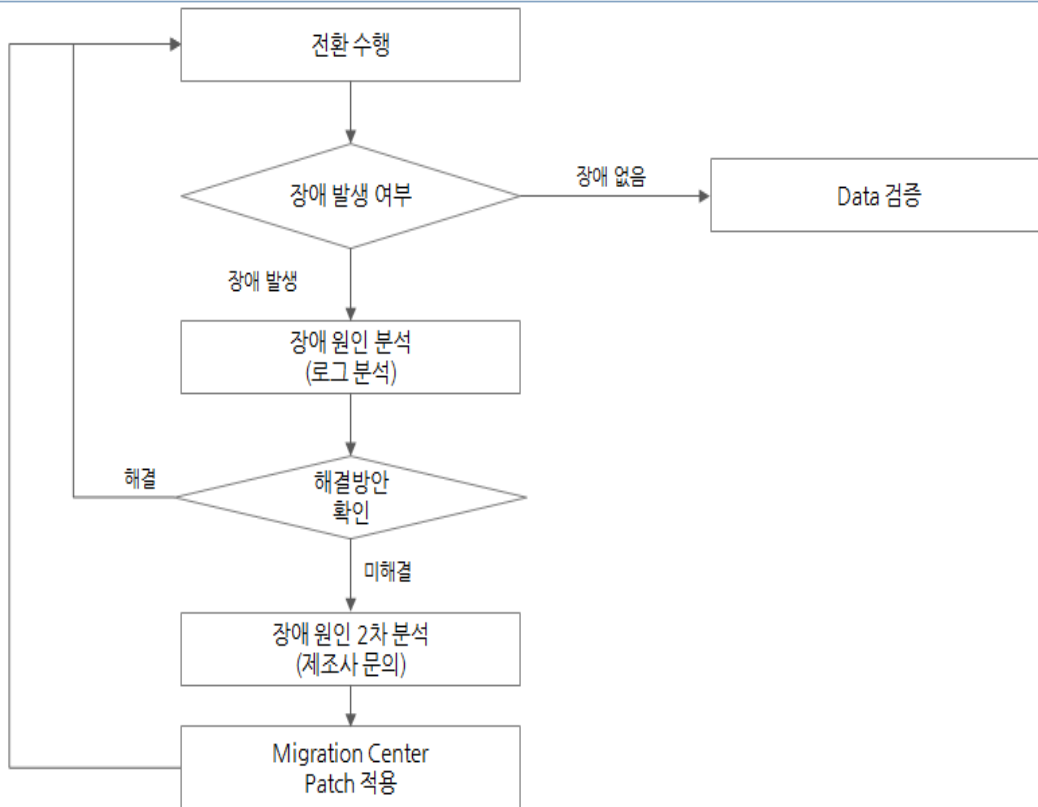
### 가. DATA 전환 및 APPLICATION 작업 절차



## 나. DATA 정합성 확인

|                |   |
|----------------|---|
| APPLICATION 검증 | <p><b>검증 방안 도출</b></p> <p>- 고객과의 사전 협의를 통하여 검증 방안 도출<br/>예) 온라인 화면 Capture 또는 장표 비교, 배치 Report 비교 등</p>   |
| DATA검증         | <p><b>Migration Center Report</b></p> <p>- Migration 결과 레포트를 통한 원본 DBMS와 ALTIBASE의 테이블 데이터 총 건수 비교</p> <p><b>검증 방안 도출</b></p> <p>- 고객과의 사전 협의를 통하여 검증 방안 도출<br/>예) 데이터 총 건수 확인, 특정 칼럼 데이터 합산 값 비교, 특정 쿼리 수행 결과 비교 등</p> |

## 다. DATA 전환 장애 대처 방안



## 1.4 전환 소요시간

ALTIBASE에서 제공하는 전환 절차에 대한 소요 시간 추정치는 다음과 같습니다.

| 단계      |       | 작업내용  | 소요기간                             |
|---------|-------|---|----------------------------------|
| 시스템 분석  |       | -소스DBMS 분석<br>DB 관리자와 인터뷰 수행<br>-시스템 분석서 작성 | -DBMS 1대 기준 1일 소요                |
| 영향도 분석  |       | -어플리케이션 변경에 따른 위험<br>영향도 분석<br>-영향도 보고서 작성  | -DBMS 1대 기준 1일 소요                |
| 계획수립    |       | -일정 및 지원 인력 계획서 작성                          | -1.5일 소요                         |
| 전환      | 데이터매핑 | -데이터 매핑 규칙 정의                               | -+a(Schema 복잡도에 따라 유동적)          |
|         | 실행    | -Migration Center를 활용                       | -1TB 기준 1일 소요                    |
| 시스템 최적화 |       | -성능 튜닝과 테스트                                 | - 1~3일 소요                        |
| 전체      |       |   | -5일 + a<br>어플리케이션 변경 영향도에 따라 유동적 |

## 1.5 전환 도구

### 가. MIGRATION CENTER

#### 1) Migration Center 이점

- GUI기반의 툴로 손쉽게 사용이 가능
- CLI 모드 지원으로 이관작업의 손쉬운 실행과 높은 성능
- 스키마와 함께 데이터도 모두 한 번의 조작으로 이관수행
- 서로 다른 데이터 형식에 대한 매핑작업을 통해 이관작업 수행
- 유연한 데이터 이관을 지원하기 위하여 사용자 정의 형식 지원
- 전환 결과에 대한 단계별 요약 보고서 제공

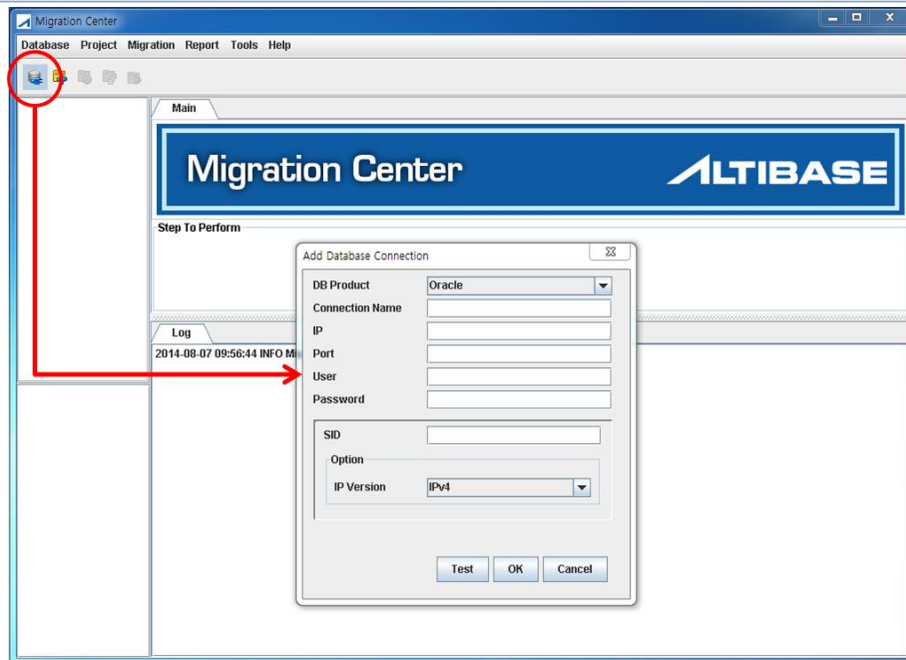
### 나. 사용법

#### 1) Add Database Connection

Source/Target DBMS의 정보를 입력합니다.

- Connection Name : 임의 지정 가능
- IP : DBMS Server IP
- Port : DBMS Server Port
- User : 연결 사용자 계정
- Password : 연결 사용자 비밀번호
- SID : 서비스 명
- IP Version : IPv4/IPv6



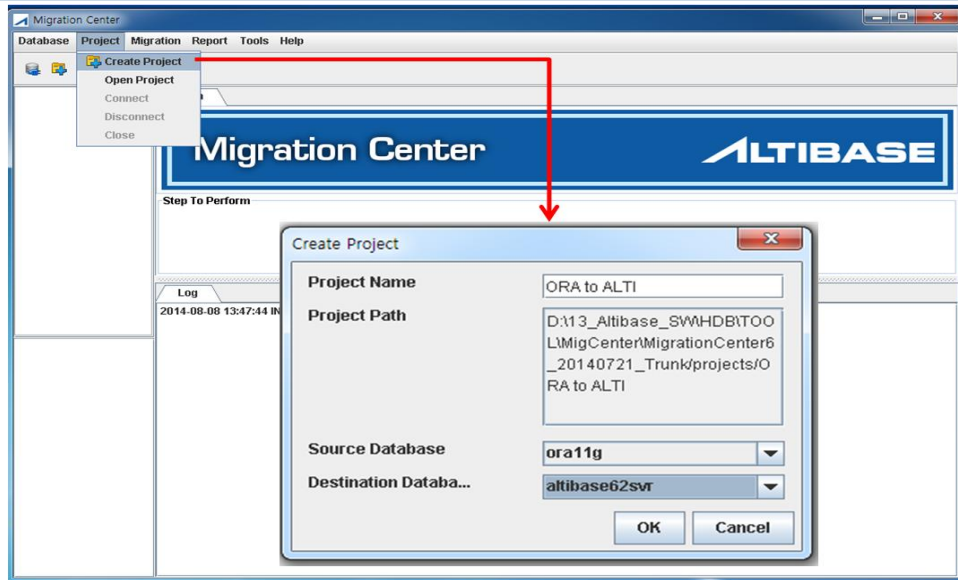


데이터베이스 접속을 위해 JDBC 드라이버를 사용하므로, 원본 데이터베이스와 대상 데이터베이스에 알맞은 JDBC 드라이버를 준비하는 것이 중요합니다. 사용자 편의를 위해 지원되는 데이터베이스에 적합한 몇 가지 JDBC 드라이버를 Migration Center와 함께 제공합니다.

## 2) Create Project

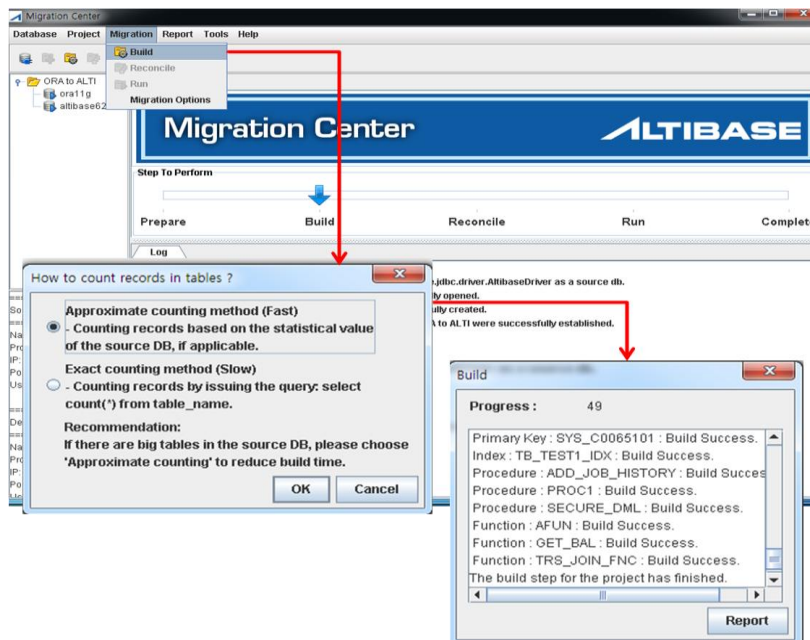
데이터 이관에 대한 프로젝트를 생성합니다.

- Project Name : 임의 지정 가능
- Project Path : 자동 생성
- Source Database : Oracle
- Destination Database : Altibase



### 3) Build Project

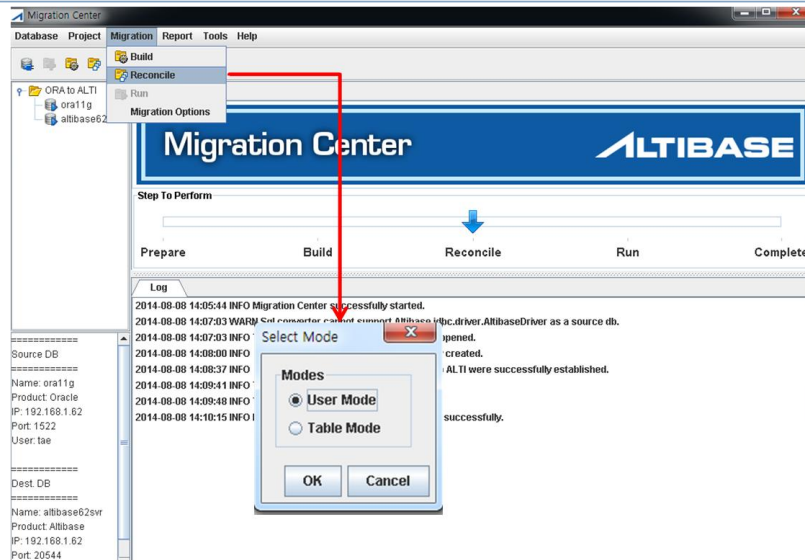
Source Database에서 테이블의 레코드 개수를 가져오는 방식(Fast/Slow)을 선택하고 진행합니다.



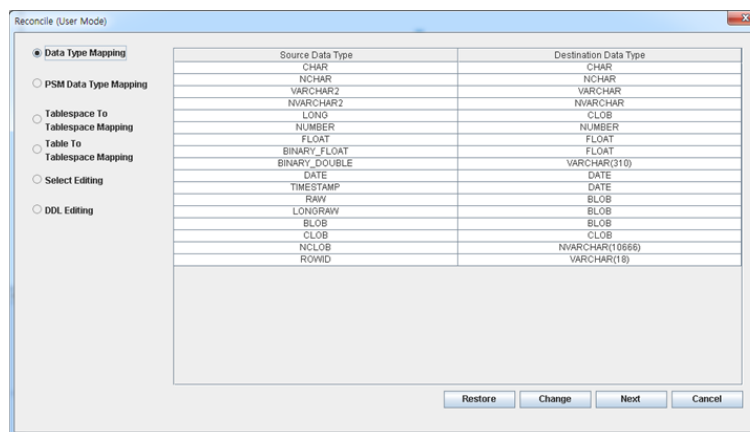
### 4) Reconcile Project

마이그레이션 수행 방법을 정의하는 단계로 마이그레이션 과정에서 가장 중요한 단계입니다.

- User Mode : Source Database 생성시 정의한 사용자 계정이 접근할 수 있는 모든 Table과 모든 데이터를 마이그레이션 대상으로 정의
- Table Mode : Source Database 생성시 정의한 사용자 계정이 접근할 수 있는 모든 Table 중에서 데이터를 마이그레이션 대상 Table을 선택 정의

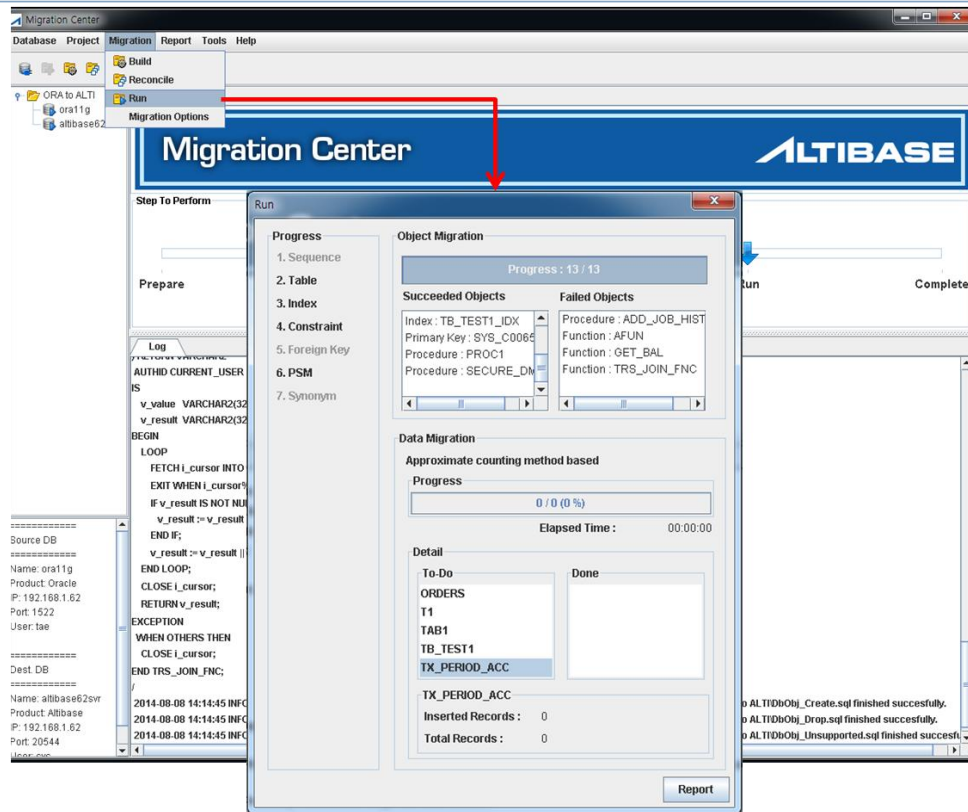


- Data Type Mapping : Oracle과 Altibase의 데이터타입 매핑
- PSM Data Type Mapping : Oracle PL/SQL과 Altibase PSM 의 데이터타입 매핑
- Tablespace to Tablespace Mapping : Oracle과 Altibase의 테이블스페이스 매핑
- Table to Tablespace Mapping : Altibase의 테이블을 저장할 테이블스페이스 지정
- Select Editing : Oracle 데이터 조회 조건 추가/수정
- DDL Editing : Altibase의 최종 DDL 수정



## 5) Run Project

데이터 마이그레이션을 수행합니다.



## 6) PL/SQL Converter Tool

파일로 저장된 Oracle PL/SQL을 Migration Center를 이용하여 데이터 타입에 대한 syntax를 변경합니다. 비즈니스 로직에 대한 변경은 자동 변환되지 않으므로, 비즈니스 로직을 구현한 개발자가 수동으로 변환해야 합니다.

### 다. 다운로드

Support.altibase.com의 다운로드 페이지를 통하여 다운로드 가능 합니다.

<http://support.altibase.com/kr/product>

### 라. 문의사항

<http://support.altibase.com>

☎ 02-2082-1114