

- [Altibase Hadoop Connector User's Manual](#)
 - [서문](#)
 - [이 매뉴얼에 대하여](#)
 - [1.Altibase 하둡 커넥터 소개](#)
 - [배경 지식](#)
 - [Altibase 하둡 커넥터란?](#)
 - [2.Altibase 하둡 커넥터 설치하기](#)
 - [소프트웨어 요구 사항](#)
 - [Altibase 하둡 커넥터 설치](#)
 - [실행 및 테스트](#)
 - [3.기능](#)
 - [커맨드 라인 옵션](#)
 - [Import](#)
 - [Export](#)
 - [list-databases](#)
 - [list-tables](#)
 - [A.부록: 데이터 타입](#)
 - [지원되는 데이터 타입](#)

Altibase® Tools & Utilities

Altibase Hadoop Connector User's Manual



Altibase Tools & Utilities Altibase Hadoop Connector User's Manual

Release 7.2

Copyright © 2001~2021 Altibase Corp. All Rights Reserved.

본 문서의 저작권은 (주)알티베이스에 있습니다. 이 문서에 대하여 당사의 동의 없이 무단으로 복제 또는 전용할 수 없습니다.

(주)알티베이스

08378 서울시 구로구 디지털로 306 대륭포스트타워Ⅱ 10층

전화: 02-2082-1114 팩스: 02-2082-1099

고객서비스포털: <http://support.altibase.com>

homepage: <http://www.altibase.com>

서문

이 매뉴얼에 대하여

이 매뉴얼은 Altibase 하둡 커넥터를 사용하여 Altibase와 하둡을 연결하는 방법을 기술한다.

대상 사용자

이 매뉴얼은 다음과 같은 사용자를 대상으로 작성되었다.

- 데이터베이스 관리자
- 데이터 분석가
- 시스템 및 응용 프로그램 개발자
- 시스템 관리자

다음과 같은 배경 지식을 가지고 이 매뉴얼을 읽는 것이 좋다.

- 컴퓨터, 운영 체제 및 운영 체제 유틸리티 운용에 필요한 기본 지식
- 관계형 데이터베이스 사용 경험 또는 데이터베이스 개념에 대한 이해
- 컴퓨터 프로그래밍 경험
- 데이터베이스 서버 관리, 운영 체제 관리 또는 네트워크 관리 경험

소프트웨어 환경

이 매뉴얼은 데이터베이스 서버로 Altibase 버전 6.0 이상을 사용한다는 가정 하에 작성되었다.

이 매뉴얼의 구성

이 매뉴얼은 다음과 같이 구성되어 있다.

- 제 1장 Altibase 하둡 커넥터 소개
이 장은 Altibase 하둡 커넥터가 무엇인지 소개하고 배경 지식을 설명한다.
- 제 2장 Altibase 하둡 커넥터 설치
이 장은 Altibase 하둡 커넥터의 설치 방법 및 이를 사용하기 위해 필요한 소프트웨어와 설치 방법도 함께 기술한다.
- 제 3장 기능
이 장은 Altibase 하둡 커넥터의 기능을 예제와 함께 설명한다.
- A. 부록: 지원하는 데이터 타입
이 부록은 Altibase 하둡 커넥터가 지원하는 Altibase의 데이터 타입에 대해 기술한다.

문서화 규칙

이 절에서는 이 매뉴얼에서 사용하는 규칙에 대해 설명한다. 이 규칙을 이해하면 이 매뉴얼과 설명서 세트의 다른 매뉴얼에서 정보를 쉽게 찾을 수 있다.

여기서 설명하는 규칙은 다음과 같다.

- 샘플 코드 규칙

샘플 코드 규칙

코드 예제는 SQL, Stored Procedure, iSQL, 또는 다른 명령 라인 구문들을 예를 들어 설명한다.

아래 테이블은 코드 예제에서 사용된 인쇄 규칙에 대해 설명한다.

| 규칙 | 의미 | 예제 |
|---------------|---|---|
| [] | 선택 항목을 표시 | VARCHAR [(size)][[FIXED \] VARIABLE] |
| { } | 필수 항목 표시. 반드시 하나 이상을 선택해야 되는 표시 | { ENABLE DISABLE COMPILE } |
| | 선택 또는 필수 항목 표시의 인자 구분 표시 | { ENABLE DISABLE COMPILE } [ENABLE DISABLE COMPILE] |
| ... | 그 이전 인자의 반복 표시 예제 코드들의 생략 되는 것을 표시 | SQL> SELECT ename FROM employee; ENAME ----- SWNO HJNO HSCHOI . . . 20 rows selected. |
| 그 밖에 기호 | 위에서 보여진 기호 이 외에 기호들 | EXEC :p1 := 1; acc NUMBER(11,2); |
| 기울 임 꼴 | 구문 요소에서 사용자가 지정해야 하는 변수, 특수한 값을 제공해야만 하는 위치 | SELECT * FROM <i>table_name</i> ; CONNECT <i>userID/password</i> ; |
| 소문 자 | 사용자가 제공하는 프로그램의 요소들, 예를 들어 테이블 이름, 칼럼 이름, 파일 이름 등 | SELECT ename FROM employee; |
| 대문 자 | 시스템에서 제공하는 요소들 또는 구문에 나타나는 키워드 | DESC SYSTEM.SYS_INDICES; |

관련 자료

자세한 정보를 위하여 다음 문서 목록을 참조한다.

- Installation Guide
- Getting Started Guide
- Administrator's Manual
- General Reference
- Error Message Reference

Altibase는 여러분의 의견을 환영합니다.

이 매뉴얼에 대한 여러분의 의견을 보내주시기 바랍니다. 사용자의 의견은 다음 버전의 매뉴얼을 작성하는데 많은 도움이 됩니다. 보내실 때에는 아래 내용과 함께 고객센터포털(<http://support.altibase.com/kr/>)로 보내주시기 바랍니다.

- 사용 중인 매뉴얼의 이름과 버전
- 매뉴얼에 대한 의견
- 사용자의 성함, 주소, 전화번호

이 외에도 Altibase 기술지원 설명서의 오류와 누락된 부분 및 기타 기술적인 문제들에 대해서 이 주소로 보내주시면 정성껏 처리하겠습니다. 또한, 기술적인 부분과 관련하여 즉각적인 도움이 필요한 경우에도 고객센터포털을 통해 서비스를 요청하시기 바랍니다.

여러분의 의견에 항상 감사드립니다.

1. Altibase 하둡 커넥터 소개

이 장은 Altibase 하둡 커넥터가 무엇인지 소개하고 배경 지식을 설명한다.

배경 지식

이 절은 스쿱을 사용하여 Altibase와 하둡간에 데이터를 이동하는 방법을 설명하기 위한 기본 개념을 설명한다.

하둡(Hadoop)

하둡(Hadoop)은 대용량 데이터의 관리와 분석에 적합한 시스템이다. '빅 데이터'에 대한 관리와 데이터웨어하우스의 병렬 처리에 대한 요구가 증가하고 클라우드와 분산 컴퓨팅이 유행하면서 가장 많이 언급되고 있는 솔루션이다.

하둡은 여러 컴퓨터로 구성된 클러스터에 걸쳐 있는 대용량 데이터를 분산 처리하기 위한 프레임워크로써 자바 기반의 오픈소스 소프트웨어이다. 하둡은 크게 하둡 분산 파일 시스템(HDFS, Hadoop Distributed File System)과 맵리듀스(MapReduce)로 구성된다.

스쿱(Sqoop)

스쿱(Sqoop)은 하둡과 관계형 데이터베이스 간의 데이터 전송을 위한 도구로써, 오픈소스 소프트웨어이다. 사용자는 스쿱을 사용하여 관계형 데이터베이스 시스템(RDBMS)에서 하둡 분산 파일 시스템(HDFS, Hadoop Distributed File System)으로 데이터를 가져오고(import) 다시 RDBMS로 내보낼(export) 수 있다.

스쿱은 가져올 데이터를 스키마로 표현하기 위해 데이터베이스에 의존하여 이 과정의 대부분을 자동화한다. 스쿱은 데이터 import와 export를 위해 맵리듀스를 사용한다. 맵리듀스는 장애 내구성(Fault Tolerance)과 병렬 작업(Parallel Operation)을 제공한다.

Altibase 하둡 커넥터란?

Altibase 하둡 커넥터(Altibase-Hadoop Connector)는 하둡과 Altibase 서버 사이의 효율적인 데이터 전송을 용이하게 하며, 운영 데이터는 Altibase에서, 데이터 분석은 하둡에서 처리할 수 있도록 해 준다. 즉, Altibase 하둡 커넥터는 하둡에서의 데이터 처리를 위해 사용자가 Altibase 서버에 접속하여 데이터를 HDFS 또는 Hive로 내보내도록 해 준다.

Altibase 하둡 커넥터는 스쿱 기반으로 동작하며 스쿱이 제공하는 거의 모든 기능을 지원한다. 또한, 스쿱과 유사한 커맨드 라인 인자 구조를 사용하기 때문에, 이전에 스쿱을 사용한 경험이 있는 사용자들은 수월하게 사용할 수 있을 것이다.

2. Altibase 하둡 커넥터 설치하기

이 장은 Altibase 하둡 커넥터의 설치 방법 및 이를 사용하기 위해 필요한 소프트웨어와 설치 방법도 함께 기술한다.

소프트웨어 요구 사항

Altibase 하둡 커넥터를 설치하고 실행하기 위해서는 아래의 소프트웨어를 먼저 설치해야 한다.

- 자바 실행 환경(JRE, Java Runtime Environment) 또는 자바 개발 키트(JDK, Java Development Kit) 버전 1.6 이상
- Hadoop 버전 1.0
- Sqoop 버전 1.4.4 이상
- Altibase 버전 5.0 이상

이 절에서는 Altibase 하둡 커넥터의 실행을 위해 필요한 하둡 및 스쿱의 설치 방법을 설명한다. 그리고 Altibase 서버와의 연동을 위해 하둡 및 스쿱 환경에 Altibase JDBC 드라이버를 설치하는 방법을 기술한다.

하둡 설치

아래의 순서대로 하둡을 설치하고 하둡 운영 환경을 설정한다.

1. <https://www.cloudera.com/content/support/en/downloads.html>에 접속하여 Hadoop 1.0 버전을 다운로드 한다.
2. <http://www.cloudera.com>에서 제공하는 Hadoop 설치 방법에 따라 Hadoop을 설치하고, HADOOP_HOME 등의 환경 변수를 설정한다.

스쿱 설치

Altibase 하둡 커넥터는 아파치 라이선스 2.0 하에 배포되는 SW인 Apache Sqoop 기반의 커넥터이다.

아래의 순서대로 스쿱을 설치하고 스쿱이 Altibase 서버에 접속할 수 있도록 Altibase JDBC 드라이버를 스쿱 환경에 설치한다.

1. <http://Mirror.apache-kr.org/sqoop/1.4.4>에 접속하여 Hadoop 1.0을 지원하는 Sqoop 패키지(sqoop-1.4.4.bin_hadoop-1.0.0.tar.gz)를 다운로드 한다. 현재 Altibase 하둡 커넥터는 Sqoop 1.4.4 버전을 공식 지원한다(Sqoop 1.9 버전은 추후 지원 예정).
2. Sqoop 홈페이지의 설치 방법에 따라 Sqoop을 설치한다.

JDBC 드라이버 설치

하둡과 스쿱을 모두 설치했다면, 아래의 지침을 따라 JDBC 드라이버를 설치하도록 한다.

Altibase 사용시

1. Altibase용 JDBC 드라이버 파일(\$ALTIBASE_HOME/lib/Altibase.jar)을 \$SQOOP_HOME/lib 폴더에 복사한다.

```
% cp $ALTIBASE_HOME/lib/Altibase.jar $SQOOP_HOME/lib
```

Altibase 하둡 커넥터 설치

아래의 순서대로 Altibase 하둡 커넥터를 설치한다.

1. Altibase 하둡 커넥터 패키지를 <http://support.altibase.com>에서 다운로드한다.

Altibase용 패키지: altibase_sqoop14_connector.jar

2. Altibase 하둡 커넥터 패키지를 \$SQOOP_HOME/lib 폴더에 복사한다.

실행 및 테스트

아래는 Altibase 서버에 접속하여 테이블 목록을 가져오는 명령어로 Altibase 하둡 커넥터가 정상적으로 설치되었는지를 확인할 수 있다.

```
% sqoop list-tables
--connect jdbc:Altibase://127.0.0.1:20300/mydb
--driver Altibase.jdbc.driver.AltibaseDriver
--username SYS
--password MANAGER
--connection-manager com.altibase.sqoop.manager.AltibaseManager
```

실행 결과 아래의 로그와 함께 테이블 목록이 출력된다면 Altibase 하둡 커넥터가 정상적으로 로딩된 것이다.

```
13/10/02 13:48:15 INFO manager.AltibaseManager: init default option autocommit false
13/10/02 13:48:15 INFO manager.SqlManager: Using default fetchSize of 1000
13/10/02 13:48:15 INFO manager.AltibaseManager: Altibase manager 1.0 connector create
```

3.기능

이 장은 Altibase 하둡 커넥터의 기능을 예제와 함께 설명한다.

커맨드 라인 옵션

Altibase 하둡 커넥터는 스쿱 기반으로 동작하기 때문에 스쿱이 제공하는 기능을 모두 지원한다. 이장은 일부 기능에 대해서만 설명하므로, 스쿱의 전체 기능은 <http://sqoop.apache.org/docs/1.4.4/SqoopUserGuide.html>을 참고하도록 한다.

구문

Altibase 하둡 커넥터를 실행하는 구문은 아래와 같다.

```
sqoop <command> --connect <url>
--driver <driver>
--username <user>
--password <password>
--connection-manager com.altibase.sqoop.manager.AltibaseManager
```

옵션

| 옵션 | 설명 |
|---|--|
| <code><command></code> | 이 문서에서는 아래의 명령어만 기술하므로, 다른 커맨드에 대해서는 위에서 언급한 스쿱 문서를 참고하기 바란다. export import list-databases list-tables |
| <code>--connect <url></code> | Altibase url을 명시한다. jdbc:Altibase://ip:port/mydb |
| <code>--driver <driver></code> | Altibase의 JDBC 드라이버 클래스를 명시한다. Altibase.jdbc.driver.AltibaseDriver |
| <code>--username <user></code> | 연결하려는 데이터베이스 사용자를 명시한다. |
| <code>--password <password></code> | 사용자의 패스워드를 명시한다. |
| <code>--connection-manager <connector></code> | Altibase 하둡 커넥터 클래스를 명시한다. com.altibase.sqoop.manager.AltibaseManager |

여기에 기술한 커맨드와 옵션 외에 다른 커맨드 또는 import 제어 옵션, export 제어 옵션 등에 대해서는 스쿱 문서 (<http://sqoop.apache.org/docs/1.4.4/SqoopUserGuide.html>)를 참고하기 바란다.

Import

Import는 Altibase의 데이터를 HDFS 또는 Hive로 가져오는 기능이다. 이 절은 이러한 기능을 예제와 함께 설명한다.

텍스트 파일로 HDFS에 가져오기

Altibase에 존재하는 특정 테이블의 데이터를 HDFS의 지정된 디렉토리에 텍스트 파일 형태로 저장하려면 sqoop 명령어와 함께 아래의 옵션들을 사용하면 된다.

```
% sqoop import
--connect <url>
--driver <jdbc_driver>
--username <user>
--password <password>
--connection-manager com.altibase.sqoop.manager.AltibaseManager
-table <table_name>
-split-by <split_name>
```

저장되는 텍스트 파일은 기본적으로 CSV 형태이다. 아래 표의 옵션들을 사용해서 필드 종료 문자, 라인 종료 문자 같은 구분자들을 지정할 수 있다. 만약 사용자가 이 중에서 하나라도 지정하지 않으면 기본값이 사용된다.

| 옵션 | 설명 | 기본값 |
|--------------------------|---|-----|
| --enclosed-by | 필드를 감싸는 문자로 필수 옵션이다. | \" |
| --escaped-by | 이스케이프 문자 | |
| --fields-terminated-by | 필드 구분 문자 | , |
| --lines-terminated-by | 라인 종료 문자 | \n |
| --mysql-delimiters | MySQL의 기본 구분자로 아래와 같다. fields-terminated-by: , lines-terminated-by: \n escaped-by: \n optionally-enclosed-by: ' | |
| --optionally-enclosed-by | 필드를 감싸는 문자로 선택 옵션이다. | |

주의: 필드 내에 라인 구분자, 필드 구분자가 포함되어 있는 경우, 큰 따옴표(") 같은 묶음 문자로 필드를 감싸 주어야 한다. 필드 내에 묶음 문자로 쓰이는 큰 따옴표(")가 포함된 경우에는 이스케이프 문자를 큰 따옴표 앞에 두어 묶음 문자가 아님을 표시해야 한다. 현재 sqoop에는 묶음 문자와 이스케이프 문자를 동일하게 지정하면 비정상적으로 동작하는 버그가 있기 때문에, Altibase 하둡 커넥터는 묶음 문자의 기본값만 큰 따옴표(")로 제공하고 이스케이프 문자는 사용자가 직접 지정하도록 지원한다.

Sequence 파일로 HDFS에 가져오기

Altibase에 존재하는 특정 테이블의 데이터를 HDFS의 지정된 디렉토리에 Sequence 파일 형태로 저장하려면 sqoop 명령어와 함께 아래의 옵션들을 사용하면 된다.

```
% sqoop import
--connect <url>
--driver <jdbc_driver>
--username <user>
--password <password>
--connection-manager com.altibase.sqoop.manager.AltibaseManager
--table <table_name>
--split-by <split_name>
--as-sequencefile
```

Avro 파일로 HDFS에 가져오기

Altibase에 존재하는 특정 테이블의 데이터를 HDFS의 지정된 디렉토리에 Avro 파일 형태로 저장하려면 sqoop 명령어와 함께 아래의 옵션들을 사용하면 된다.

```
% sqoop import
--connect <url>
--driver <jdbc_driver>
--username <user>
--password <password>
--connection-manager com.altibase.sqoop.manager.AltibaseManager
--table <table_name>
--split-by <split_name>
--as-avrodatafile
```

쿼리를 사용해서 HDFS에 가져오기

사용자가 지정한 질의문으로 Altibase에 존재하는 특정 테이블의 데이터를 조회하여 HDFS에 저장하려면 sqoop 명령어와 함께 아래의 옵션들을 사용하면 된다.

```
% sqoop import
--connect <url>
--driver <jdbc_driver>
--username <user>
--password <password>
--connection-manager com.altibase.sqoop.manager.AltibaseManager
--table <table_name>
--split-by <split_name>
--boundary-query <query>
```

Hive에 가져오기

Altibase에 존재하는 특정 테이블의 데이터를 Hive에 저장하려면 sqoop 명령어와 함께 아래의 옵션들을 사용하면 된다.

```
% sqoop import
--connect <url>
--driver <jdbc_driver>
--username <user>
--password <password>
--connection-manager com.altibase.sqoop.manager.AltibaseManager
--table <table_name>
--split-by <split_name>
--hive-import
```

Export

Export는 HDFS의 데이터를 Altibase로 내보내는 기능이다. 옵션에 따라 데이터베이스에 데이터를 삽입하거나 기존 데이터를 갱신하는 동작이 수행된다.

데이터 삽입

HDFS의 데이터를 Altibase의 특정 테이블에 저장하려면 sqoop 명령어와 함께 아래의 옵션들을 사용하면 된다.

```
% sqoop export
-D sqoop.export.records.per.statement=<size>
--connect <url>
--driver <jdbc_driver>
--username <user>
--password <password>
--connection-manager com.altibase.sqoop.manager.AltibaseManager
--table <table_name>
--export-dir <dir>
```

sqoop.export.records.per.statement의 값을 1보다 크게 지정하면 Altibase 하둡 커넥터는 배치 모드(batch mode)로 동작한다. 배치 모드에서는 한 번의 Execute 수행으로 여러 건의 레코드가 삽입된다. 이 옵션은 커맨드 옵션(export) 뒤에 다른 옵션들보다 먼저 나와야 한다. sqoop.export.records.per.statement의 값을 1로 지정하면 Altibase 하둡 커넥터가 배치 모드로 동작하지 않는다.

CSV 파일에서 데이터 삽입

CSV 파일을 사용해서 export하는 경우, 구분자들을 지정하는 옵션을 사용할 수 있다. 사용 가능한 옵션에 대한 상세한 설명은 "[텍스트 파일로 HDFS에 가져오기](#)" 절을 참고하기 바란다.

배치 모드로 데이터 삽입

Altibase JDBC 드라이버의 배치 모드 기능을 사용하여 HDFS의 데이터를 Altibase의 특정 테이블에 저장하려면 sqoop 명령어와 함께 아래의 옵션들을 사용하면 된다.

```
% sqoop export
--connect <url>
--driver <jdbc_driver>
--username <user>
--password <password>
--connection-manager com.altibase.sqoop.manager.AltibaseManager
--table <table_name>
--export-dir <dir>
--batch
```

--batch 옵션 지정 여부에 관계없이 Altibase 하둡 커넥터는 기본으로 배치 모드로 동작하며, 한 번의 Execute 수행으로 기본 100건의 레코드가 삽입된다. Altibase 하둡 커넥터가 배치 모드로 동작하지 않게 하려면, -D sqoop.export.records.per.statement 옵션을 1로 지정해야 한다.

데이터 갱신

Altibase의 특정 테이블의 데이터를 HDFS의 데이터로 갱신하려면 sqoop 명령어와 함께 아래의 옵션들을 사용하면 된다.

```
% sqoop export
--connect <url>
--driver <jdbc_driver>
--username <user>
--password <password>
--connection-manager com.altibase.sqoop.manager.AltibaseManager
--table <table_name>
--export-dir <dir>
--update-key <column>
```

데이터 갱신 또는 삽입

HDFS의 데이터가 Altibase에 존재하면 갱신하고, 존재하지 않으면 삽입하는 기능이다. sqoop 명령어와 함께 아래의 옵션들을 사용하면 된다.

```
% sqoop export
--connect <url>
--driver <jdbc_driver>
--username <user>
--password <password>
--connection-manager com.altibase.sqoop.manager.AltibaseManager
--table <table_name>
--export-dir <dir>
--update-key <column>
--update-mode allowinsert
```

주의: 이 기능은 Altibase의 merge 구문을 이용하기 때문에, Altibase 하둡 커넥터가 버전 6.3.1 이상의 Altibase와 함께 동작할 때만 지원된다.

Staging-table을 이용한 데이터 삽입

Sqoop은 export 처리를 여러 개의 트랜잭션으로 수행하기 때문에 일부 데이터는 커밋이 실패할 수도 있다. --staging-table 옵션은 이것을 방지하기 위한 것으로, HDFS의 데이터를 이 옵션에 지정한 테이블에 먼저 삽입한 후에, 이 테이블의 데이터를 Altibase의 대상 테이블로 이동하는 기능이다. sqoop 명령어와 함께 아래의 옵션들을 사용하면 된다.

```
% sqoop export
--connect <url>
--driver <jdbc_driver>
--username <user>
--password <password>
--connection-manager com.altibase.sqoop.manager.AltibaseManager
--table <table_name>
--export-dir <dir>
--update-key <column>
--staging-table <table_name>
```

list-databases

Altibase의 데이터베이스 목록을 조회하려면, sqoop 명령어와 함께 아래의 옵션들을 사용하면 된다.

```
% sqoop list-databases
--connect <url>
--driver <jdbc_driver>
--username <user>
--password <password>
--connection-manager com.altibase.sqoop.manager.AltibaseManager
```

list-tables

Altibase에 존재하는 테이블들을 조회하려면, sqoop 명령어와 함께 아래의 옵션들을 사용하면 된다.

```
% sqoop list-tables
--connect <url>
--driver <jdbc_driver>
--username <user>
--password <password>
--connection-manager com.altibase.sqoop.manager.AltibaseManager
```

A.부록: 데이터 타입

이 부록은 Altibase 하둡 커넥터가 지원하는 Altibase의 데이터 타입에 대해 기술한다.

지원되는 데이터 타입

아래는 Altibase 하둡 커넥터를 사용하여 import 또는 export 할 때의 Altibase와 Sqoop(Altibase 하둡 커넥터) 간에 각 데이터 타입의 변환을 보여주는 표이다. 또한 각 데이터 타입별 import 및 export 지원 여부도 보여준다.

| Altibase 데이터 타입 | Sqoop 데이터 타입 | Import 지원 여부 | Export 지원 여부 |
|-----------------|--------------------------------|--------------|--------------|
| CHAR | String | O | O |
| VARCHAR | String | O | O |
| NCHAR | String | O | O |
| NVARCHAR | String | O | O |
| INTEGER | Integer | O | O |
| BIGINT | Long | O | O |
| SMALLINT | Integer | O | O |
| NUMBER | Double | O | O |
| NUMERIC | java.math.BigDecimal | O | O |
| DECIMAL | java.math.BigDecimal | O | O |
| FLOAT | Double | O | O |
| DOUBLE | Double | O | O |
| REAL | Float | O | O |
| DATE | java.sql.Timestamp | O | O |
| BLOB | com.cloudera.sqoop.lib.BlobRef | O | X |
| CLOB | com.cloudera.sqoop.lib.ClobRef | O | X |

참고: BLOB와 CLOB 데이터 타입은 Sqoop에서 현재 import 기능만 지원하므로, Altibase 하둡 커넥터도 import만 지원한다. BLOB와 CLOB 데이터 타입의 export 기능은 추후 지원할 예정이다.

