

Databricks Tables Data Type Conversions

Contents:

- *Access/Read*
- *Write/Publish*
 - *Create new table*
 - *Append to existing table*
 - *Notes on Datetime columns*

NOTE: The Trifacta® data types listed in this page reflect the raw data type of the converted column. Depending on the contents of the column, the Transformer Page may re-infer a different data type, when a dataset using this type of source is loaded.

Access/Read

When a Databricks Tables data type is imported, its JDBC data type is remapped according to the following table.

Tip: Data precision may be lost during conversion. You may want to generate min and max values and compute significant digits for values in your Hive tables and then compute the same in the Trifacta application.

Source Data Type	Supported?	Trifacta Data Type	Notes
array	Y	Array	
bigint	Y	Integer	NOTE: The Trifacta platform may infer bigint columns containing very large or very small values as String data type.
binary	Y	String	
boolean	Y	Bool	
char	Y	String	
date	Y	Datetime	
decimal	Y	Decimal	
double	Y	Decimal	
float	Y	Decimal	NOTE: On import, some float columns may be interpreted as Integer data type in the Trifacta platform. To fix, you can explicitly set the column's data type to Decimal in the Transformer page.
int	Y	Integer	
map	Y	Object	
smallint	Y	Integer	
string	Y	String	
struct	Y	Object	

timestamp	Y	Datetime	
tinyint	Y	Integer	
uniontype	N		
varchar	Y	String	

Write/Publish

Create new table

NOTE: By default, the maximum length of values published to VARCHAR columns is 256 characters. As needed, this limit can be changed for multiple publication targets. For more information, see *Configure Application Limits*.

Trifacta Data Type	Databricks Tables Data Type	Notes
String	string	
Integer	bigint	<p>NOTE: The Trifacta platform may infer Integer columns containing very large or very small values as String data type. Before you publish, you should verify that your columns containing extreme values are interpreted as Integer type. You can import a target schema to assist in lining up your columns with the expected target. For more information, see <i>Overview of RapidTarget</i>.</p>
Decimal	double	
Bool	boolean	
Datetime	Timestamp /string (see Notes on Datetime columns below)	Target data type is based on the underlying data. Time zone information is retained.
Object	string	
Array	string	

Append to existing table

If you are publishing to a pre-existing table, the following data type conversions apply:

- **Columns:** Trifacta data types
- **Rows:** Target table data types

In any table cell, a Y indicates that the append operation for that data type mapping is supported.

NOTE: You cannot append to Databricks Tables map and array column types from Trifacta columns of Map and Array type, even if you imported data from this source.

	String	Integer	Datetime	Bool	Decimal	Map	Array	Out of Range error
CHAR	Y	Y	Y	Y	Y	Y	Y	
VARCHAR	Y	Y	Y	Y	Y	Y	Y	
STRING	Y	Y	Y	Y	Y	Y	Y	

INT		Y					NULL
BIGINT		Y					n/a
TINYINT							NULL
SMALLINT							NULL
DECIMAL		Y			Y		NULL
DOUBLE		Y			Y		n/a
FLOAT					Y		NULL
TIMESTAMP			Y				
BOOLEAN				Y			

Notes on Datetime columns

Run Job

Columns in new tables created for output of `Datetime` columns are written with the Databricks Tables `timestamp` data type. These columns can be appended.

A single job cannot write `Datetime` values to one table as `String` type and to another table as `Timestamp` type. This type of job should be split into multiple types. The table schemas may require modification.

- The above issue may appear as the following error when executing the job:

```
Unable to publish due to datetime data type conflict in column XXXX
```