Data Field Types in Datameer

When importing data into Datameer, it is important to know which type of data you are importing or need to import. When using your data within workbooks, certain Datameer functions only work with certain types of data. Congruently, certain functions return only a specific type of data.
Also there are data requirements when using infographic widgets to visualize your data.

Data Field Types

Field type	Product icon	Description	Internal representation
integer		64-Bit integer value	Java Long
big integer		Unlimited integer value	Java BigInteger
float		64-Bit float value	Java Double
big decimal		High-precision float value	Java BigDecimal
date		Date object	Java Date
string		String object	Java String
boolean		Boolean object	Java Boolean
list		a collection of multiple values of one data type	
number		float, big decimal, integer, or big integer	
any		float, big decimal, integer, big integer, date, string, list, or Boolean	

Integer

In mathematics integers (aka whole numbers) are made up of the set of natural numbers including zero (0,1,2,3, ...) along with the negatives of natural numbers (-1,-2,-3, ...). When talking about Integers in computer programming, it is necessary to define a minimum and maximum value. Datameer uses a 64-bit integer which allows the user to represent whole numbers between -9,223,372,036,854,775,808 to 9,223,372,036,854,775,807.

Big integer

Big integers are like integers, but they are not limited to 64 bits. They are represented using arbitrary-precision arithmetic. Big integers represent only whole numbers. Big integers in Datameer are treated differently than in Hive because Datameer allows a larger range of values, so they are written as strings into a Hive table if you export. By default, the precision for big integers is set at 32 upon import. This can updated if needed in the default properties by changing the value of *das.big-decimal.precision=.*

Float

In mathematics, there are real numbers that represent fractions (1/2, 12/68) or numbers with decimal places (12.75, -18.35). Datameer uses double precision floating-point representation (aka float) to manipulate and represent real numbers. The complete range of numbers that can be represented this way is approximately 2^{-1022} through $(1+(1-2^{-52}))x2^{1023}$. During import/upload, Datameer automatically recognizes a number with either a single period (.) or single comma (,) as a decimal separator and defines this data as a float data type. After ingestion, Datameer stores float and big decimal values using a period (.) character. The auto schema detection for the float data type works with CSV, JSON, XML, Key/value files.

Big decimal

Big decimals are similar to float values. The main advantage of this data field type is that they are exact to the number of decimal places for which they are configured, float values might be inaccurate in certain cases. If a number has more decimal places than big decimal was configured for, then the number is rounded. The number of decimal places can be configured in `conf/default.properties`:

```
# Maximum precision used for BIG_DECIMAL types. Precision is equal to
the maximum number of digits a BigDecimal
# can have.
system.property.das.big-decimal.precision=32
```

32 digits is the default precision used by Datameer for big decimal values upon import.

Date

In Datameer, data in the DATE primitive data type is always represented in a Gregorian, month-day-year (MDY) format (e.g., "Sep 16, 2010 02:56:39 PM"). Datameer detects if your data should be parsed into the DATE data type during ingest. This can also be done after ingest as other data types can be converted to the DATE primitive data type using workbook functions.

String

When using information other than numbers or dates in Datameer, it is represented as a string. This includes text, unparsed date patterns, URLs, JSON arrays, etc.

Boolean

Boolean data in computing has two values, either true or false. It is used in many logical expressions and is derived from Boolean algebra created by George Boole in the 19th century.

List

In Datameer multiple values can be combined into a list. Lists are a series of values of a single data type, which starts counting from zero (0).

Number

In Datameer integers, big integers, floats, and big decimals are considered to be numbers.

Any

Some visualizations and functions are able to use data represented by any data field type. These can be either a number, a string, a date, or a Boolean.

Data Mapping in Avro

Import mapping

When importing data to Datameer, data types are mapped as follows:

Avro Schema Type	Datameer Value Type
null	STRING
boolean	BOOLEAN
int	INTEGER
long	INTEGER
float	FLOAT
double	FLOAT
bytes	STRING
bytes with logical type decimal	BIG_DECIMAL
string	STRING
records	STRING
enums	STRING
arrays	STRING
maps	STRING
unions	STRING
fixed	STRING
fixed with logical type decimal	BIG_DECIMAL

Export mapping

When exporting data to Avro, data types are mapped as follows:

> **INFO**
>
> If a column marked as accept empty a union scheme type is created, e.g. union(null, string) for nullable string type.

Datameer Value Type	Avro Schema Type
STRING	string
BOOLEAN	boolean
INTEGER	long
FLOAT	double
DATE without pattern	long
DATE with pattern	string
BIG_INTEGER	string
BIG_DECIMAL	string
LIST<listtype>	arrays<converted list type>

Data Mapping in Parquet

Export data mapping

When exporting data to Parquet, data types are mapped as follows:

Datameer field type	Parquet field type
integer	INT64
date	BINARY
big integer	BINARY
float	DOUBLE
big decimal	BINARY
string	BINARY UTF8
Boolean	BOOLEAN
list (integer)	INT64
list (float)	DOUBLE
list (string)	BINARY
list (Boolean)	BOOLEAN

Import data mapping

When importing data from Parquet, data types are mapped as follows:

Parquet Field Type	Datameer Field Type
BOOLEAN	BOOLEAN
INT32	INTEGER
INT32 DECIMAL	BIG_DECIMAL
INT64	INTEGER
INT64 DECIMAL	BIG_DECIMAL
INT96	DATE
FLOAT	FLOAT
DOUBLE	FLOAT
BINARY	STRING
BINARY DECIMAL	BIG_DECIMAL
FIXED_LEN_BYTE_ARRAY	STRING
FIXED_LEN_BYTE_ARRAY DECIMAL	BIG_DECIMAL

Parquet files using the INT96 format are interpreted as time stamps. Datameer accepts these columns, but cuts off the nanoseconds. If the workbook has Ignore Errors enabled, then those error messages are stored in a separate column and the column with the error is NULL. The chart below provides additional Parquet storage mapping details.

Datameer Type	Parquet Type	Description
date	TIMESTAMP_MILLIS	Stored as INT64
integer	INT_64	
float	DOUBLE	
string	UTF8	BINARY format with UTF-8 encoding

big decimal	DECIMAL	BINARY format
big integer	DECIMAL	BINARY format with precision of 1 and scale of 0
list	Repeated elements of group of the list element type.	Optional group of a repeating group of optional element types. Nested lists are supported.