



Informatica® PowerExchange for Microsoft
Azure Blob Storage
10.2 HotFix 1

User Guide

Informatica PowerExchange for Microsoft Azure Blob Storage User Guide
10.2 HotFix 1
July 2018

© Copyright Informatica LLC 2016, 2019

This software and documentation are provided only under a separate license agreement containing restrictions on use and disclosure. No part of this document may be reproduced or transmitted in any form, by any means (electronic, photocopying, recording or otherwise) without prior consent of Informatica LLC.

U.S. GOVERNMENT RIGHTS Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation is subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License.

Informatica, the Informatica logo, PowerExchange, and Big Data Management are trademarks or registered trademarks of Informatica LLC in the United States and many jurisdictions throughout the world. A current list of Informatica trademarks is available on the web at <https://www.informatica.com/trademarks.html>. Other company and product names may be trade names or trademarks of their respective owners.

Portions of this software and/or documentation are subject to copyright held by third parties. Required third party notices are included with the product.

The information in this documentation is subject to change without notice. If you find any problems in this documentation, report them to us at infa_documentation@informatica.com.

Informatica products are warranted according to the terms and conditions of the agreements under which they are provided. INFORMATICA PROVIDES THE INFORMATION IN THIS DOCUMENT "AS IS" WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING WITHOUT ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND ANY WARRANTY OR CONDITION OF NON-INFRINGEMENT.

Publication Date: 2019-04-12

Table of Contents

Preface	5
Informatica Resources.	5
Informatica Network.	5
Informatica Knowledge Base.	5
Informatica Documentation.	6
Informatica Product Availability Matrixes.	6
Informatica Velocity.	6
Informatica Marketplace.	6
Informatica Global Customer Support.	6
Chapter 1: Introduction to PowerExchange for Microsoft Azure Blob Storage.....	7
PowerExchange for Microsoft Azure Blob Storage Overview.	7
PowerExchange for Microsoft Azure Blob Storage Example.	7
Introduction to Microsoft Azure Blob Storage.	8
Chapter 2: PowerExchange for Microsoft Azure Blob Storage Configuration... ..	9
PowerExchange for Microsoft Azure Blob Storage Configuration Overview.	9
Prerequisites.	9
Java Heap Memory Configuration (Optional).	10
Configure Temporary Directory Location (Optional).	10
Chapter 3: Microsoft Azure Blob Storage Connections.....	11
Microsoft Azure Blob Storage Connection Overview.	11
Microsoft Azure Blob Storage Connection Properties.	11
Creating a Microsoft Azure Blob Storage Connection.	12
Chapter 4: Microsoft Azure Blob Storage Data Objects.....	13
Microsoft Azure Blob Storage Data Objects Overview.	13
Microsoft Azure Blob Storage Data Object Properties.	13
Microsoft Azure Blob Storage Data Object Read Operation Properties.	14
Microsoft Azure Blob Storage Data Object Write Operation Properties.	15
Creating a Microsoft Azure Blob Storage Data Object.	16
Creating a Data Object Operation.	17
Chapter 5: Microsoft Azure Blob Storage Mappings.....	18
Microsoft Azure Blob Storage Mappings Overview.	18
Mapping Validation and Run-time Environments.	18

Chapter 6: Data Type Reference.....	19
Data Type Reference Overview.	19
Microsoft Azure Blob Storage and Transformation Data Types.	20
Avro File Data Types and Transformation Data Types.	20
Parquet File Data Types and Transformation Data Types.	21
Index.	22

Preface

The Informatica PowerExchange® for Microsoft Azure Blob Storage User Guide provides information about reading data from and writing data to Microsoft Azure Blob Storage. The guide is written for database administrators and developers who are responsible for developing mappings that read data from and write data to Microsoft Azure Blob Storage.

The guide assumes you have knowledge of Microsoft Azure Blob Storage and Informatica.

Informatica Resources

Informatica Network

Informatica Network hosts Informatica Global Customer Support, the Informatica Knowledge Base, and other product resources. To access Informatica Network, visit <https://network.informatica.com>.

As a member, you can:

- Access all of your Informatica resources in one place.
- Search the Knowledge Base for product resources, including documentation, FAQs, and best practices.
- View product availability information.
- Review your support cases.
- Find your local Informatica User Group Network and collaborate with your peers.

As a member, you can:

- Access all of your Informatica resources in one place.
- Search the Knowledge Base for product resources, including documentation, FAQs, and best practices.
- View product availability information.
- Find your local Informatica User Group Network and collaborate with your peers.

Informatica Knowledge Base

Use the Informatica Knowledge Base to search Informatica Network for product resources such as documentation, how-to articles, best practices, and PAMs.

To access the Knowledge Base, visit <https://kb.informatica.com>. If you have questions, comments, or ideas about the Knowledge Base, contact the Informatica Knowledge Base team at KB_Feedback@informatica.com.

Informatica Documentation

To get the latest documentation for your product, browse the Informatica Knowledge Base at https://kb.informatica.com/_layouts/ProductDocumentation/Page/ProductDocumentSearch.aspx.

If you have questions, comments, or ideas about this documentation, contact the Informatica Documentation team through email at infa_documentation@informatica.com.

Informatica Product Availability Matrixes

Product Availability Matrixes (PAMs) indicate the versions of operating systems, databases, and other types of data sources and targets that a product release supports. If you are an Informatica Network member, you can access PAMs at

<https://network.informatica.com/community/informatica-network/product-availability-matrixes>.

Informatica Velocity

Informatica Velocity is a collection of tips and best practices developed by Informatica Professional Services. Developed from the real-world experience of hundreds of data management projects, Informatica Velocity represents the collective knowledge of our consultants who have worked with organizations from around the world to plan, develop, deploy, and maintain successful data management solutions.

If you are an Informatica Network member, you can access Informatica Velocity resources at <http://velocity.informatica.com>.

If you have questions, comments, or ideas about Informatica Velocity, contact Informatica Professional Services at ips@informatica.com.

Informatica Marketplace

The Informatica Marketplace is a forum where you can find solutions that augment, extend, or enhance your Informatica implementations. By leveraging any of the hundreds of solutions from Informatica developers and partners, you can improve your productivity and speed up time to implementation on your projects. You can access Informatica Marketplace at <https://marketplace.informatica.com>.

Informatica Global Customer Support

You can contact a Global Support Center by telephone or through Online Support on Informatica Network.

To find your local Informatica Global Customer Support telephone number, visit the Informatica website at the following link:

<http://www.informatica.com/us/services-and-training/support-services/global-support-centers>.

If you are an Informatica Network member, you can use Online Support at <http://network.informatica.com>.

CHAPTER 1

Introduction to PowerExchange for Microsoft Azure Blob Storage

This chapter includes the following topics:

- [PowerExchange for Microsoft Azure Blob Storage Overview, 7](#)
- [PowerExchange for Microsoft Azure Blob Storage Example, 7](#)
- [Introduction to Microsoft Azure Blob Storage, 8](#)

PowerExchange for Microsoft Azure Blob Storage Overview

You can use PowerExchange for Microsoft Azure Blob Storage to connect to Microsoft Azure Blob Storage from Informatica.

Use PowerExchange for Microsoft Azure Blob Storage to read data from or write data to Microsoft Azure Blob Storage. Use PowerExchange for Microsoft Azure Blob Storage to read delimited files and the industry-standard file formats, Avro and Parquet. You can read and write Avro and Parquet files in the native environment.

You can read and write the gzip compressed `.csv` files only in the native environment. Create a Microsoft Azure Blob Storage connection to read or write Microsoft Azure Blob Storage data into a Microsoft Azure Blob Storage data object. When you use Microsoft Azure Blob Storage objects in mappings, you must configure properties specific to Microsoft Azure Blob Storage. You can validate and run mappings in the native environment or in the Hadoop environment.

PowerExchange for Microsoft Azure Blob Storage Example

You work in sales operations and want to score leads to drive higher sales for your organization. You need to bring in leads from Salesforce to Microsoft Azure Blob Storage. You can score leads for sales readiness in Microsoft Azure Machine Learning, and then load the lead scores back into Salesforce. You can keep data up to date with the latest leads and lead scores by scheduling a workflow to run on a regular basis.

You have leads in Salesforce with data such as the contact information, industry, company size, and marketing information.

You configure a mapping to insert leads from Salesforce to Microsoft Azure Blob Storage. Use Microsoft Azure Machine Learning to score the leads, and then create another mapping to load the lead scores into Salesforce.

You create a workflow so that the tasks run serially promising leads and increase efficiency.

Introduction to Microsoft Azure Blob Storage

Microsoft Azure Blob Storage is a cloud-storage solution that stores unstructured data in the cloud as objects or blobs. Microsoft Azure Blob Storage can store text or binary data of any type, such as a document, media files, or application installer. Microsoft Azure Blob Storage is referred to as object storage.

Blobs are files of any type and size, and are organized into containers in Microsoft Azure Storage. You can access delimited files that are append blobs or block blobs with Microsoft Azure Blob Storage connections.

CHAPTER 2

PowerExchange for Microsoft Azure Blob Storage Configuration

This chapter includes the following topics:

- [PowerExchange for Microsoft Azure Blob Storage Configuration Overview, 9](#)
- [Prerequisites, 9](#)
- [Java Heap Memory Configuration \(Optional\), 10](#)
- [Configure Temporary Directory Location \(Optional\), 10](#)

PowerExchange for Microsoft Azure Blob Storage Configuration Overview

PowerExchange for Microsoft Azure Blob Storage installs with the Informatica services and clients.

Prerequisites

To successfully preview data from a local complex file or run a mapping in the native environment with a local complex file, you must configure the INFA_PARSER_HOME property for the Data Integration Service in Informatica Administrator. Perform the following steps to configure the INFA_PARSER_HOME property:

- Log in to Informatica Administrator.
- Click the Data Integration Service and then click the **Processes** tab on the right pane.
- Click **Edit** in the **Environment Variables** section.
- Click **New** to add an environment variable.
- Enter the name of the environment variable as **INFA_PARSER_HOME**.
- Set the value of the environment variable to the absolute path of the Hadoop distribution directory on the machine that runs the Data Integration Service. Verify that the version of the Hadoop distribution directory that you define in the INFA_PARSER_HOME property is the same as the version you defined in the cluster configuration.

Java Heap Memory Configuration (Optional)

Configure the memory for the Java heap size in the node that runs the Data Integration Service.

1. In the Administrator tool, navigate to the Data Integration Service for which you want to change the Java heap size.
2. Edit the Custom Properties section in the Data Integration Service Properties.
3. To increase the heap memory size for a large dataset, define the following properties:

```
ExecutionContextOptions.JVMMaxMemory = <size> MB  
ExecutionContextOptions.JVMMinMemory = <size> MB
```

Where <size> is a valid heap size, such as 2048 MB.

4. Click Ok.
5. Restart the Data Integration Service.

Configure Temporary Directory Location (Optional)

Follow below steps to configure the temporary directory location in the node that runs the Data Integration Service.

1. In the Administrator tool, navigate to the Data Integration Service for which you want to change the temporary directory location.
2. Click the **Processes** tab.
3. Click **Custom Properties**. The **Edit Custom Properties** dialog box appears.
4. Click **New** to add a new custom property.
5. Add the JVMOption custom property for the Data Integration Service and specify the value in the following format:

```
-Djava.io.tmpdir=<required tmp directory location>
```

For example,

```
Property Name: JVMOption1  
Value: -Djava.io.tmpdir=/opt/Informatica/tmp/ZUDAP/
```

6. Click **Ok**.
7. Restart the Data Integration Service.

CHAPTER 3

Microsoft Azure Blob Storage Connections

This chapter includes the following topics:

- [Microsoft Azure Blob Storage Connection Overview, 11](#)
- [Microsoft Azure Blob Storage Connection Properties, 11](#)
- [Creating a Microsoft Azure Blob Storage Connection, 12](#)

Microsoft Azure Blob Storage Connection Overview

Microsoft Azure Blob Storage connection enables you to read data from or write data to Microsoft Azure Blob Storage.

You can use Microsoft Azure Blob Storage connections to create data objects and run mappings. The Developer tool uses the connection when you create a data object. The Data Integration Service uses the connection when you run mappings.

You can create an Microsoft Azure Blob Storage connection from the Developer tool or the Administrator tool. The Developer tool stores connections in the domain configuration repository. Create and manage connections in the connection preferences.

Microsoft Azure Blob Storage Connection Properties

When you set up a Microsoft Azure Blob Storage connection, you must configure the connection properties.

The following table describes the Microsoft Azure Blob Storage connection properties:

Property	Description
Name	Name of the Microsoft Azure Blob Storage connection.
ID	String that the Data Integration Service uses to identify the connection. The ID is not case sensitive. It must be 255 characters or less and must be unique in the domain. You cannot change this property after you create the connection. Default value is the connection name.

Property	Description
Description	Description of the connection.
Location	The domain where you want to create the connection.
Type	Type of connection. Select AzureBlob.

The **Connection Details** tab contains the connection attributes of the Microsoft Azure Blob Storage connection. The following table describes the connection attributes:

Property	Description
Account Name	Name of the Microsoft Azure Storage account.
Account Key	Microsoft Azure Storage access key.
Container Name	The root container or sub-folders with the absolute path.

Creating a Microsoft Azure Blob Storage Connection

Before you create a Microsoft Azure Blob Storage data object, create a connection in the Developer tool.

1. Click **Window > Preferences**.
2. Select **Informatica > Connections**.
3. Expand the domain in the **Available Connections**.
4. Select the connection type **File System > AzureBlob**, and click **Add**.
5. Enter a connection name and an optional description.
6. Enter an ID for the connection.
7. Select **AzureBlob** as the connection type.
8. Click **Next**.
9. Configure the connection properties.
10. Click **Test Connection** to verify the connection to Microsoft Azure Blob Storage.
11. Click **Finish**.

CHAPTER 4

Microsoft Azure Blob Storage Data Objects

This chapter includes the following topics:

- [Microsoft Azure Blob Storage Data Objects Overview, 13](#)
- [Microsoft Azure Blob Storage Data Object Properties, 13](#)
- [Creating a Microsoft Azure Blob Storage Data Object, 16](#)
- [Creating a Data Object Operation, 17](#)

Microsoft Azure Blob Storage Data Objects Overview

A Microsoft Azure Blob Storage data object is a physical data object that uses Microsoft Azure Blob Storage as a source or target. A Microsoft Azure Blob Storage data object represents the data in a Microsoft Azure Blob Storage file.

You can configure the data object read and write operation properties that determine how data can be read from Microsoft Azure Blob Storage sources and loaded to Microsoft Azure Blob Storage targets. You first create a connection to create a Microsoft Azure Blob Storage data object. When you create a data object, the read and write operations are created by default. You can modify the default read and write operations or create additional operations.

Microsoft Azure Blob Storage Data Object Properties

Specify the data object properties when you create the data object.

The following table describes the properties that you configure for the Microsoft Azure Blob Storage data objects:

Property	Description
Name	Name of the Microsoft Azure Blob Storage data object.
Location	The project or folder in the Model Repository where you want to store the Microsoft Azure Blob Storage data object.
Connection	Name of the Microsoft Azure Blob Storage connection.

Microsoft Azure Blob Storage Data Object Read Operation Properties

Microsoft Azure Blob Storage data object read operation properties include advanced properties that apply to the Microsoft Azure Blob Storage data object.

The Developer tool displays advanced properties for the Microsoft Azure Blob Storage data object operation in the **Advanced** view.

The following table describes the advanced properties that you can configure for a Microsoft Azure Blob Storage data object read operation:

Property	Description
Number of concurrent connections to Blob Store	The number of concurrent connections to Blob Store to download files. Default is 4.
Source Type	Select the type of source from which you want to read data. You can select the following source types: <ul style="list-style-type: none">- File- Directory Default is File . Applicable to the native environment.
Blob Name Override	Overrides the file name.
Blob Container Override	Overrides the default container name. You can specify the container name or sub-folders in the root container with the absolute path.

Directory Source in Microsoft Azure Blob Storage Sources

You can select the type of source from which you want to read data.

You can select the following type of sources from the **Source Type** option under the advanced properties for a Microsoft Azure Blob Storage data object read operation:

- File
- Directory

Note: The content type of the blob must be `application/x-gzip` to read compressed files for both File and Directory source types.

Use the following rules and guidelines to select **Directory** as the source type:

- All the source files in the directory must contain the same metadata.

- All the files must have data in the same format. For example, delimiters, header fields, and escape characters must be same.
- All the files under a specified directory are parsed. The files under subdirectories are not parsed.
- The connector does not perform any validation if there are multiple blob formats in the directory you select and might result into errors.

File Port in Microsoft Azure Blob Storage Sources

The file port stores the file name from which the Data Integration Service reads the data at run-time.

When you import a Microsoft Azure Blob Storage Source object, the Data Integration Service creates a file port. The file port stores the name of the file from which the Data Integration Service reads the data at run-time.

For example, a directory contains a number of files and each file contains multiple records that you want to read. You select the directory as source type in the Microsoft Azure Blob Storage Data Object Read Operation Properties. When you run the mapping, the Data Integration Service reads each record and stores the respective file name in the file port. The file port is applicable to the `.csv`, binary, Avro, and Parquet file formats in the native environment.

Microsoft Azure Blob Storage Data Object Write Operation Properties

Microsoft Azure Blob Storage data object write operation properties include advanced properties that apply to the Microsoft Azure Blob Storage data object.

The Developer tool displays advanced properties for the Microsoft Azure Blob Storage data object operation in the **Advanced** view.

The following table describes the advanced properties that you can configure for a Microsoft Azure Blob Storage data object write operation:

Property	Description
Number of concurrent connections to Blob Store	The number of concurrent connections to Blob Store to upload files. Default is 4.
Blob Name Override	Overrides the file name. Use <code>.gz</code> extension to write compressed blob files to Microsoft Azure Blob Storage.
Blob Container Override	Overrides the default container name. You can specify the container name or sub-folders in the root container with the absolute path.
Compress newly created Blob	Compresses the newly created blob when set to True. Applicable only to delimited files in the native environment and the hive engine.
Write Strategy	Override: overrides an existing blob. Append: appends to an existing blob. Append operation is applicable only to the append blob type.
Blob Type	Writes data to a block blob or an append blob. You can write data to an append blob only for the flat format files in the native environment.

Creating a Microsoft Azure Blob Storage Data Object

Create a Microsoft Azure Blob Storage data object to add to a mapping.

1. Select a project or folder in the **Object Explorer** view.
2. Click **File > New > Data Object**.
3. Select **AzureBlob Data Object** and click **Next**.
The AzureBlob Data Object dialog box appears.
4. Enter a name for the data object.
5. In the **Resource Format** list, select any of the following formats:
 - Intelligent Structure Model: not applicable.
 - Binary: to read any resource format.
 - Flat: to read and write delimited resources.
 - Avro: to read and write Avro resources.
 - Json: not applicable.
 - Parquet: to read and write Parquet resources.
6. Click **Browse** next to the **Location** option and select the target project or folder.
7. Click **Browse** next to the **Connection** option and select the AzureBlob connection from which you want to import the Microsoft Azure Blob Storage object.
8. To add a resource, click **Add** next to the **Selected Resources** option.
The Add Resource dialog box appears.
9. Select the checkbox next to the Microsoft Azure Blob Storage object you want to add and click **OK**.
You must select the object according to the selected resource format. To use an intelligent structure model, select the appropriate `.amodel` file.
10. Click **Finish** if you selected Avro, Json, or Parquet file and skip the remaining steps. Click **Next** if you selected a flat file.
11. Applicable only to the delimited files. Choose **Sample Metadata File**.
You can click Browse and navigate to the directory that contains the file.
12. Click **Next**.
13. Configure the format properties.

Property	Description
Delimiters	Character used to separate columns of data. Make sure that the delimiter you select is not a part of the data. If you enter a delimiter that is the same as the escape character or the text qualifier, you might receive unexpected results. Microsoft Azure Blob Storage reader and writer support Delimiters.
Text Qualifier	Quote character that defines the boundaries of text strings. Make sure that the text qualifier you select is not a part of the data unless it is escaped. If you select a quote character, the Developer tool ignores delimiters within pairs of quotes. Microsoft Azure Blob Storage reader supports Text Qualifier.
Import Column Names From First Line	If selected, the Developer tool uses data in the first row for column names. Select this option if column names appear in the first row. Do not select this option to read header-less files. Applicable only to the native environment.

14. Click **Next** to configure the column properties and edit the column attributes.
15. Click **Finish**.

The data object appears under Data Objects in the project or folder in the Object Explorer view. The data object read and write operations are created by default when a data object is created.

Creating a Data Object Operation

You can create the data object read or write operation for a Microsoft Azure Blob Storage data objects. You can then add the Microsoft Azure Blob Storage data object operation to a mapping.

1. Select the data object in the **Object Explorer** view.
2. Right-click and select **New > Data Object Operation**.
The Data Object Operation dialog box appears.
3. Enter a name for the data object operation.
4. Select the type of data object operation.
You can choose read or write operation.
5. Click **Add**.
The Select Resources dialog box appears.
6. Select the Microsoft Azure Blob Storage data object for which you want to create the data object operation and click **OK**.
Note: Select the input or output file.
7. Click **Finish**.
The Developer tool creates the data object operation for the selected data object.

CHAPTER 5

Microsoft Azure Blob Storage Mappings

This chapter includes the following topics:

- [Microsoft Azure Blob Storage Mappings Overview, 18](#)
- [Mapping Validation and Run-time Environments, 18](#)

Microsoft Azure Blob Storage Mappings Overview

After you create a Microsoft Azure Blob Storage data object operation, you can develop a mapping.

You can define the following objects in the mapping:

- Microsoft Azure Blob Storage data object read operation as the input to read data from Microsoft Azure Blob Storage.
- Microsoft Azure Blob Storage data object write operation as the output to write data to Microsoft Azure Blob Storage.

Note: You do not require any additional steps to run PowerExchange for Microsoft Azure Blob Storage in Hadoop Environment.

Mapping Validation and Run-time Environments

You can validate and run mappings in the native environment or Hadoop Environment.

You can validate a mapping in the native environment, Hadoop Environment, or both. The Data Integration Service validates whether the mapping can run in the selected environment. You must validate the mapping for an environment before you run the mapping in that environment.

When you run a mapping in the native environment, the Data Integration Service runs the mapping.

When you run a mapping on a Hadoop cluster, you can select the Hive or Spark engine. The Data Integration Service pushes the mappings to the selected engine for processing.

Note: A mapping that uses different Azure Blob Storage accounts for the Azure Blob Storage source and target objects fails on the Spark engine.

CHAPTER 6

Data Type Reference

This chapter includes the following topics:

- [Data Type Reference Overview, 19](#)
- [Microsoft Azure Blob Storage and Transformation Data Types, 20](#)
- [Avro File Data Types and Transformation Data Types, 20](#)
- [Parquet File Data Types and Transformation Data Types, 21](#)

Data Type Reference Overview

Informatica developer uses the following data types in Microsoft Azure Blob Storage mappings:

- Microsoft Azure Blob Storage native data types. Microsoft Azure Blob Storage data types appear in the physical data object column properties.
- Transformation data types. Set of data types that appear in the transformations. They are internal data types based on ANSI SQL-92 generic data types, which the Data Integration Service uses to move data across platforms. Transformation data types appear in all transformations in a mapping.

When the Data Integration Service reads a source, it converts the native data types to the comparable transformation data types before transforming the data. When the Data Integration Service writes to a target, it converts the transformation data types to the comparable native data types.

Microsoft Azure Blob Storage and Transformation Data Types

The following table lists the Microsoft Azure Blob Storage data types that the Data Integration Service supports and the corresponding transformation data types:

Microsoft Azure Blob Storage Native Data Type	Transformation Data Type	Range and Description
Bigint	Bigint	-9,223,372,036,854,775,808 to 9,223,372,036,854,775,807 Precision 19, scale 0
Decimal	Decimal	Precision 15
Integer	Integer	-2,147,483,648 to 2,147,483,647 Precision 10, scale 0
String	String	1 to 104,857,600 characters

Avro File Data Types and Transformation Data Types

Avro file data types map to transformation data types that the Data Integration Service uses to move data across platforms.

The following table lists the Avro file data types that the Data Integration Service supports and the corresponding transformation data types:

Avro File Data Type	Transformation Data Type	Range and Description
Boolean	Integer	TRUE (1) or FALSE (0)
Bytes	Binary	Precision 4000
Double	Double	Precision 15
Float	Double	Precision 15
Int	Integer	-2,147,483,648 to 2,147,483,647 Precision 10, scale 0
Long	Bigint	-9,223,372,036,854,775,808 to 9,223,372,036,854,775,807 Precision 19, scale 0

Avro File Data Type	Transformation Data Type	Range and Description
Null	Integer	-2,147,483,648 to 2,147,483,647 Precision 10, scale 0
String	String	1 to 104,857,600 characters

Note: The Null data type is supported only in the native environment.

Parquet File Data Types and Transformation Data Types

Parquet file data types map to transformation data types that the Data Integration Service uses to move data across platforms.

The following table lists the Parquet file data types that the Data Integration Service supports and the corresponding transformation data types:

Parquet File Data Type	Transformation Data Type	Range and Description
Binary (UTF-8)	String	1 to 104,857,600 characters
Boolean	Integer	TRUE (1) or FALSE (0)
Byte_Array	Binary	Arbitrarily long byte array
Double	Double	Precision 15
Float	Double	Precision 15
Int32	Integer	-2,147,483,648 to +2,147,483,647 Precision of 10, scale of 0
Int64	Bigint	-9,223,372,036,854,775,808 to +9,223,372,036,854,775,807 Precision of 19, scale of 0

The Parquet schema that you specify to read or write a Parquet file must be in smaller case. Parquet does not support case-sensitive schema.

INDEX

C

- Configure Temporary Directory Location [10](#)
- Connection
 - details [11](#)
 - properties [11](#)
- Create
 - Data objects [11](#)
 - storage connection [12](#)

D

- Data
 - objects [13](#)
- Data object
 - operation [17](#)
 - properties [13](#)
 - storage [16](#)
- Data types
 - Avro file [20](#)
 - description [20](#)
 - range [20](#)
 - transformation [20](#)
- directory source
 - Microsoft Azure Blob Storage sources [14](#)

E

- Example
 - Microsoft Azure Blob Storage [7](#)

F

- file port
 - Microsoft Azure Blob Storage sources [15](#)

J

- java heap size [10](#)

M

- mapping
 - storage [18](#)
 - validation [18](#)
- mappings
 - run [11](#)
- Microsoft Azure
 - Blob Storage [7](#)

O

- operation
 - read [18](#)
 - write [18](#)

P

- Parquet file data types
 - transformation data types [21](#)
- PowerExchange [7](#)
- PowerExchange for Microsoft Azure Blob Storage configuration
 - overview [9](#)
- Prerequisites [9](#)

S

- Storage connection
 - overview [11](#)
- storage solution
 - cloud [8](#)

U

- unstructured data
 - blobs [8](#)
 - objects [8](#)