Informatica

Informatica® PowerExchange for Amazon S3 10.5.2.1

User Guide

Informatica PowerExchange for Amazon S3 User Guide
10.5.2.1
July 2022

Publication Date: 2022-08-11

Table of Contents

Preface

Use the *Informatica® PowerExchange® for Amazon S3 User Guide* to learn how to read from or write to Amazon S3 by using the Developer tool. Learn to create a connection, develop and run mappings and dynamic mappings in the native environment and in the Hadoop and Databricks environments.

Informatica Resources

Informatica provides you with a range of product resources through the Informatica Network and other online portals. Use the resources to get the most from your Informatica products and solutions and to learn from other Informatica users and subject matter experts.

Informatica Network

The Informatica Network is the gateway to many resources, including the Informatica Knowledge Base and Informatica Global Customer Support. To enter the Informatica Network, visit https://network.informatica.com.

As an Informatica Network member, you have the following options:

- Search the Knowledge Base for product resources.
- View product availability information.
- Create and review your support cases.
- Find your local Informatica User Group Network and collaborate with your peers.

Informatica Knowledge Base

Use the Informatica Knowledge Base to find product resources such as how-to articles, best practices, video tutorials, and answers to frequently asked questions.

To search the Knowledge Base, visit https://search.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

Use the Informatica Documentation Portal to explore an extensive library of documentation for current and recent product releases. To explore the Documentation Portal, visit https://docs.informatica.com.

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

Informatica Product Availability Matrices

Product Availability Matrices (PAMs) indicate the versions of the operating systems, databases, and types of data sources and targets that a product release supports. You can browse the Informatica PAMs at https://network.informatica.com/community/informatica-network/product-availability-matrices.

Informatica Velocity

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

You can find 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 extend and enhance your Informatica implementations. Leverage any of the hundreds of solutions from Informatica developers and partners on the Marketplace to improve your productivity and speed up time to implementation on your projects. You can find the Informatica Marketplace at https://marketplace.informatica.com.

Informatica Global Customer Support

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

To find your local Informatica Global Customer Support telephone number, visit the Informatica website at the following link:
https://www.informatica.com/services-and-training/customer-success-services/contact-us.html.

To find online support resources on the Informatica Network, visit https://network.informatica.com and select the eSupport option.

CHAPTER 1

Introduction to PowerExchange for Amazon S3

This chapter includes the following topics:

PowerExchange for Amazon S3 Overview

You can use PowerExchange for Amazon S3 to read and write delimited flat file data and binary files as pass-through data from and to Amazon S3 buckets.

Amazon S3 is a cloud-based store that stores many objects in one or more buckets.

Create an Amazon S3 connection to specify the location of Amazon S3 sources and targets you want to include in a data object. You can use the Amazon S3 connection in data object read and write operations. You can also connect to Amazon S3 buckets available in Virtual Private Cloud (VPC) through VPC endpoints.

You can run mappings in the native or non-native environment. Select the Blaze, Spark, or Databricks Spark engines when you run mappings in the non-native environment.

Example

You are a medical data analyst in a medical and pharmaceutical organization who maintains patient records. A patient record can contain patient details, doctor details, treatment history, and insurance from multiple data sources.

You use PowerExchange for Amazon S3 to collate and organize the patient details from multiple input sources in Amazon S3 buckets.

Introduction to Amazon S3

Amazon Simple Storage Service (Amazon S3) is storage service in which you can copy data from source and simultaneously move data to any target. You can use Amazon S3 to store and retrieve any amount of data at any time, from anywhere on the web. You can accomplish these tasks using the AWS Management Console web interface.

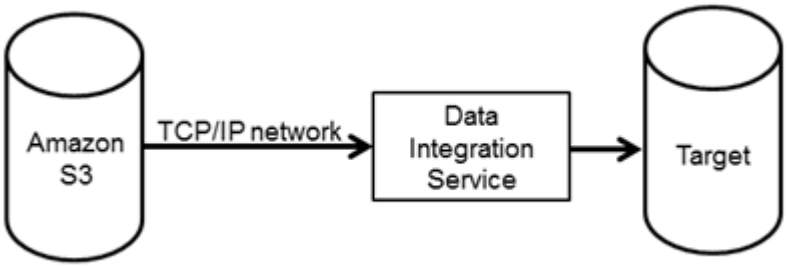
Amazon S3 stores data as objects within buckets. An object consists of a file and optionally any metadata that describes that file. To store an object in Amazon S3, you upload the file you want to store to a bucket. Buckets are the containers for objects. You can have one or more buckets. When using the AWS Management Console, you can create folders to group objects, and you can nest folders.

Data Integration Service and Amazon S3 Integration

The Data Integration Service uses the Amazon S3 connection to connect to Amazon S3.

Reading Amazon S3 Data

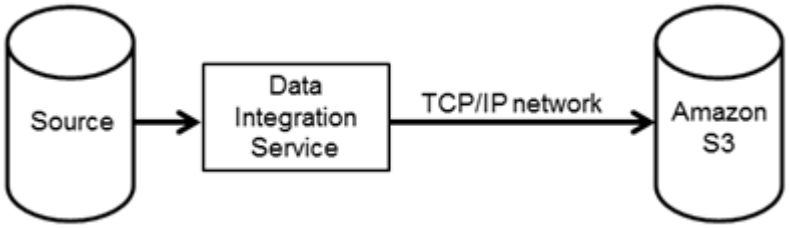
The following image shows how Informatica connects to Amazon S3 to read data:



When you run the Amazon S3 session, the Data Integration Service reads data from Amazon S3 based on the workflow and Amazon S3 connection configuration. The Data Integration Service connects and reads data from Amazon Simple Storage Service (Amazon S3) through a TCP/IP network. The Data Integration Service then stores data in a staging directory on the Data Integration Service host. Amazon S3 is a storage service in which you can copy data from source and simultaneously move data to any target. The Data Integration Service issues a copy command that copies data from Amazon S3 to the target.

Writing Amazon S3 Data

The following image shows how Informatica connects to Amazon S3 to write data:



When you run the Amazon S3 session, the Data Integration Service writes data to Amazon S3 based on the workflow and Amazon S3 connection configuration. The Data Integration Service stores data in a staging directory on the Data Integration Service host. The Data Integration Service then connects and writes data to Amazon Simple Storage Service (Amazon S3) through a TCP/IP network. Amazon S3 is a storage service in which you can copy data from source and simultaneously move data to Amazon S3 clusters. The Data Integration Service issues a copy command that copies data from Amazon S3 to the Amazon S3 target file.

CHAPTER 2

PowerExchange for Amazon S3 Configuration Overview

This chapter includes the following topics:

PowerExchange for Amazon S3 Configuration Overview

PowerExchange for Amazon S3 installs with the Informatica Services. You can enable PowerExchange for Amazon S3 with a license key.

Authentication Methods

You can configure the following authentication methods:

- **Basic authentication**: Provide the access key and secret access key.
- **IAM authentication**: Configure IAM authentication when the Secure Agent runs on an Amazon Elastic Compute Cloud (EC2) system.
- **Temporary security credentials via AssumeRole**: Use **AssumeRole** to access the AWS resources from the same or different AWS accounts.
- **Federated user single sign-on**:Configure federated user single sign-on to securely control access to the Amazon S3 resources.

Prerequisites

Before you can use PowerExchange for Amazon S3, perform the following tasks:

- Ensure that PowerExchange for Amazon S3 license is activated.
- Create an Access Key ID and Secret Access Key in AWS. You can provide these key values when you create an Amazon S3 connection
- Ensure that you have the **sts:AssumeRole** permission and a trust relationship established within the AWS accounts to use the temporary security credentials to access the AWS resources.
- Create the temporary security credentials policy to use the temporary security credentials to access the AWS resources.
- Verify that you have write permissions on all the directories within the `<INFA_HOME>` directory.
- To run mappings on Hortonworks and Amazon EMR distributions that use non-Kerberos authentication, configure user impersonation.
  For information about configuring user impersonation, see the *Data Engineering Integration Guide*.
- To run mappings on MapR secure clusters, configure the MapR secure clusters on all the nodes.
  For information about configuring MapR secure clusters, see the *Data Engineering Integration Guide*.
- To run mappings on the Spark engine and in EMR 5.20 or later distributions that use server-side encryption with KMS, enable the KMS policy for the Amazon S3 bucket.
- To successfully preview data from complex files, 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.
- To run mappings on Spark or Databricks Spark engine or to test connections through Administration tool using a secure domain, you must perform the following steps:
  - Download the `Baltimore CyberTrust Root` certificate file.
  - Provide the read, write, and execute permissions to the certificate file.
  - Run the following command to import the certificate file into the Informatica TrustStore location:

    ```
    <INFA_HOME>/java/jre/bin/keytool -keystore <infa_trust_store_location> -importcert -
    alias <Alias_Name> -file <BaltimoreCyberTrustRoot certificate file path>/
    <certificate_filename> -storepass <Truststore_Password>
    ```

  - Restart the Data Integration Service.

IAM Authentication

Optional. You can configure Amazon Identity and Access Management (IAM) authentication when the Data Integration Service runs on an Amazon Elastic Compute Cloud (EC2) system. Use IAM authentication for secure and controlled access to Amazon S3 resources when you run a session.

Note:

Use IAM authentication when you want to run a session on an EC2 system. Perform the following steps to configure IAM authentication:

1. Create a minimal Amazon IAM Policy. For more information, see "Create a Minimal Amazon IAM Policy" on page 12.
2. Create the Amazon EC2 role. Associate the minimal Amazon IAM policy while creating the EC2 role. The Amazon EC2 role is used when you create an EC2 system in the S3 bucket. For more information about creating the Amazon EC2 role, see the AWS documentation.
3. Create an EC2 instance. Assign the Amazon EC2 role that you created in step #2 to the EC2 instance.
4. Install the Data Integration Service on the EC2 system.

You can use Amazon IAM authentication when you run a mapping in the EMR cluster. To use Amazon IAM authentication in the EMR cluster, you must create the Amazon EMR Role. Create a new Amazon EMR Role or use the default Amazon EMR Role. You must assign the Amazon ERM Role to the EMR cluster for secure access to Amazon S3 resources.

Note: Before you configure IAM Role with EMR cluster, you must install the Informatica Services on an EC2 instance with the IAM Roles assigned.

Create a Minimal Amazon IAM Policy

You can configure an IAM policy through the AWS console. Use Amazon IAM authentication to securely control access to Amazon S3 resources.

You can configure an IAM policy through the AWS console. Use AWS IAM authentication to securely control access to Amazon S3 resources.

Use the following minimum required policies for users to successfully read data from an Amazon S3 bucket:

- GetObject
- ListBucket

Use the following minimum required policies for users to successfully write data to an Amazon S3 bucket:

- PutObject
- GetObject
- DeleteObject
- ListBucket
- ListBucketMultipartUploads. Applicable only for elastic mappings.

You can use the following sample minimal Amazon IAM policy:

```
{
    "Version": "2012-10-17",
    "Statement":
    [
        {
            "Effect": "Allow",
            "Action":[
                "s3:PutObject",
```

```
                "s3:GetObject",
                "s3:DeleteObject",
                "s3:ListBucket",
                    ],
            "Resource": [
                "arn:aws:s3:::<bucket_name>/*",
                "arn:aws:s3:::<bucket_name>"
                        ]
        }
    ]
}
```

AssumeRole

You can use the temporary security credentials using AssumeRole to access the AWS resources from the same or different AWS accounts.

Ensure that you have the **sts:AssumeRole** permission and a trust relationship established within the AWS accounts to use the temporary security credentials. The trust relationship is defined in the trust policy of the IAM role when you create the role. The IAM role adds the IAM user as a trusted entity allowing the IAM users to use the temporary security credentials and access the AWS accounts. For more information about how to establish the trust relationship, see the AWS documentation.

When the trusted IAM user requests for the temporary security credentials, the AWS Security Token Service (AWS STS) dynamically generates the temporary security credentials that are valid for a specified period and provides the credentials to the trusted IAM users. The temporary security credentials consist of access key ID, secret access key, and secret token.

To use the dynamically generated temporary security credentials, provide the value of the **IAM Role ARN** connection property when you create an Amazon S3 connection. The IAM Role ARN uniquely identifies the AWS resources. Then, specify the time duration in seconds during which you can use the temporarily security credentials in the **Temporary Credential Duration** advanced read and write operation properties.

AssumeRole Policy

To use the temporary security credentials to access the AWS resources, both the IAM user and IAM role require policies.

The following section lists the policies required for the IAM user and IAM role:

IAM user

An IAM user must have the `sts:AssumeRole` policy to use the temporary security credentials in the same or different AWS account.

The following sample policy allows an IAM user to use the temporary security credentials in an AWS account:

```
{
"Version":"2012-10-17", "Statement":{ "Effect":"Allow", "Action":"sts:AssumeRole",
"Resource":"arn:aws:iam::<ACCOUNT-HYPHENS>:role/<ROLE-NAME>" }
}
```

IAM role

An IAM role must have a `sts:AssumeRole` policy and a trust policy attached with the IAM role to allow the IAM user to access the AWS resource using the temporary security credentials. The policy specifies

the AWS resource that the IAM user can access and the actions that the IAM user can perform. The trust policy specifies the IAM user from the AWS account that can access the AWS resource.

The following policy is a sample trust policy:

```
{
"Version":"2012-10-17", "Statement":[{ "Effect":"Allow", "Principal":
{ "AWS":"arn:aws:iam::AWS-account-ID:root" },
"Action":"sts:AssumeRole" }
 ]
  }
   }
```

Here, in the `Principal` attribute, you can also provide the ARN of IAM user who can use the dynamically generated temporary security credentials and to restrict further access. For example,

```
"Principal" : { "AWS" : "arn:aws:iam:: AWS-account-ID :user/ user-name " }
```

To use the temporary security credentials with AWS Key Management Service (AWS KMS)-managed customer master key and enable the encryption with KMS, you must create a KMS policy.

You can perform the following operations to use the temporary security credentials and enable the encryption with KMS:

- GenerateDataKey
- DescribeKey
- Encrypt
- Decrypt
- ReEncrypt

Sample policy:

```
{
"Version":"2012-10-17", "Statement":[{ "Effect":"Allow", "Action":
[ "kms:GenerateDataKey", "kms:DescribeKey", "kms:Encrypt", "kms:Decrypt",
"kms:ReEncrypt*" ], "Resource":"*"
    }
  ]
   }
```

Rules and Guidelines of Using AssumeRole

Consider the following guidelines when you use the temporary security credentials:

- You can use assume role on Cloudera CDH, Hortonworks HDP, and Cloudera CDP distributions for the same AWS account and cross-account access. You cannot use assume role on MapR and Azure HDInsight distributions.
- To use assume role on EMR distributions for same AWS account and cross-account access, you must set up a security configuration. For more information on how to set up a security configuration, see the AWS documentation.
- The IAM user who requests for the temporary security credentials should not have access to any AWS resources.
- Only authenticated IAM users can request for the temporary security credentials from the AWS Security Token Service (AWS STS).
- Before you run a mapping, ensure that you have enough time to use the temporary security credentials for running the mapping. You cannot extend the time duration of the temporary security credentials for an ongoing mapping. For example, when you upload a file to an Amazon S3 bucket and if the temporary security credentials expire, you cannot extend the time duration of the temporary security credentials that causes the mapping to fail.

- After the temporary security credentials expire, AWS does not authorize the IAM users to access the resources using the credentials. You must request for new temporary security credentials before the previous temporary security credentials expire.
- Do not use the root user credentials of an AWS account to use the temporary security credentials. You must use the credentials of an IAM user to use the temporary security credentials.

Configure Databricks Cluster

Set the access key ID and secret access key values under Spark Config in your Databricks cluster configuration to access Amazon S3 storage. You must specify one key value pair per line and each key value pair must be separated by a single space.

```
spark.hadoop.fs.s3a.awsAccessKeyId xxyyzz
spark.hadoop.fs.s3a.awsSecretAccessKey xxxyyyzzz
```

Access using IAM role

Optional. Create an IAM role associated with the AWS account of the Databricks deployment. Amazon S3 bucket must belong to the same account associated with the Databricks deployment. If the bucket belongs to a different AWS account, then, the Cross-Account bucket policy must be enabled to access the bucket.

Server-side S3 encryption (AES-256)

Optional. Set the `server-side-encryption-algorithm` property under Spark Config in your Databricks cluster configuration:

```
spark.hadoop.fs.s3a.server-side-encryption-algorithm AES256
```

Server-side encryption using SSE-KMS

Optional. Set the following properties under Spark Config in your Databricks cluster configuration:

```
spark.hadoop.fs.s3a.server-side-encryption-kms-master-key-id arn:aws:kms:us-west-XX:key/
XXXYYYYYYY
spark.hadoop.fs.s3a.server-side-encryption-algorithm aws:kms
spark.hadoop.fs.s3a.impl com.databricks.s3a.S3AFileSystem
```

Configure Proxy Settings

You can configure proxy to connect to Amazon S3 in the native environment, on the Spark engine, or on the Databricks Spark engine. For more information on how to configure proxy, see the following Knowledge base articles:

- For the native environment, see KB 562908.
- For Spark engine, see KB 000186916.
- For Databricks Spark engine, see KB 000186919.

.

CHAPTER 3

Amazon S3 Connections

This chapter includes the following topics:

Amazon S3 Connections Overview

Amazon S3 connections enable you to read data from or write data to Amazon S3.

When you create an Amazon S3 connection, you define connection attributes. You can create an Amazon S3 connection in 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. The Developer tool uses the connection when you create data objects. The Data Integration Service uses the connection when you run mappings.

You can use AWS Identity and Access Management (IAM) authentication to securely control access to Amazon S3 resources. If you have valid AWS credentials and you want to use IAM authentication, you do not have to specify the access key and secret key when you create an Amazon S3 connection.

When you run a mapping that reads data from an Amazon S3 source and writes data to an Amazon S3 target on the Spark or Databricks Spark engine, the mapping fails if the AWS credentials such as Access Key or Secret Key are different for source and target.

Amazon S3 Connection Properties

When you set up an Amazon S3 connection, you must configure the connection properties.

The following table describes the Amazon S3 connection properties:

Property	Description
Name	The name of the connection. The name is not case sensitive and must be unique within the domain. You can change this property after you create the connection. The name cannot exceed 128 characters, contain spaces, or contain the following special characters:~ ` ! $ % ^ & * ( ) - + = { [}] | \ : ; " ' < , > . ? /
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.
Description	Optional. The description of the connection. The description cannot exceed 4,000 characters.
Location	The domain where you want to create the connection.
Type	The Amazon S3 connection type.
Access Key	Access key to access the Amazon S3 bucket. Provide the access key value based on the following authentication methods: - Basic authentication: provide the actual access key value. - IAM authentication: do not provide the access key value. - Temporary security credentials via assume role: provide access key of an IAM user with no permissions to access Amazon S3 bucket.
Secret Key	Secret access key to access the Amazon S3 bucket. The secret key is associated with the access key and uniquely identifies the account. Provide the access key value based on the following authentication methods: - Basic authentication: provide the actual access secret value. - IAM authentication: do not provide the access secret value. - Temporary security credentials via assume role: provide access secret of an IAM user with no permissions to access Amazon S3 bucket.
IAM Role ARN	The ARN of the IAM role assumed by the user to use the dynamically generated temporary security credentials. Enter the value of this property if you want to use the temporary security credentials to access the AWS resources. If you want to use the temporary security credentials with IAM authentication, do not provide the Access Key and Secret Key connection properties. If you want to use the temporary security credentials without IAM authentication, you must enter the value of the Access Key and Secret Key connection properties. For more information about how to obtain the ARN of the IAM role, see the AWS documentation.
Folder Path	The complete path to Amazon S3 objects. The path must include the bucket name and any folder name. Do not use a slash at the end of the folder path. For example, `<bucket name>/<my folder name>`.
Master Symmetric Key	Optional. Provide a 256-bit AES encryption key in the Base64 format when you enable client-side encryption. You can generate a master symmetric key using a third-party tool.

Property	Description
S3 Account Type	The type of the Amazon S3 account. Select **Amazon S3 Storage** or **S3 Compatible Storage**. Select the Amazon S3 storage option to use the Amazon S3 services. Select the S3 compatible storage option to specify the endpoint for a third-party storage provider such as Scality RING. By default, Amazon S3 storage is selected.
REST Endpoint	The S3 storage endpoint. Specify the S3 storage endpoint in HTTP/HTTPs format when you select the S3 compatible storage option. For example, `http://s3.isv.scality.com`.
Region Name	Select the AWS region in which the bucket you want to access resides. Select one of the following regions: - Asia Pacific (Mumbai) - Asia Pacific (Seoul) - Asia Pacific (Singapore) - Asia Pacific (Sydney) - Asia Pacific (Tokyo) - AWS GovCloud (US) - Canada (Central) - China (Beijing) - China (Hong Kong) - China (Ningxia) - EU (Ireland) - EU (Frankfurt) - EU (London) - EU (Paris) - South America (Sao Paulo) - US East (Ohio) - US East (N. Virginia) - US West (N. California) - US West (Oregon) Default is US East (N. Virginia). Not applicable for S3 compatible storage.
Customer Master Key ID	Optional. Specify the customer master key ID or alias name generated by AWS Key Management Service (AWS KMS) or the Amazon Resource Name (ARN) of your custom key for cross-account access. You must generate the customer master key for the same region where Amazon S3 bucket reside. You can specify any of the following values: **Customer generated customer master key** Enables client-side or server-side encryption. **Default customer master key** Enables client-side or server-side encryption. Only the administrator user of the account can use the default customer master key ID to enable client-side encryption.
Federated SSO IdP	SAML 2.0-enabled identity provider for the federated user single sign-on to use with the AWS account. PowerExchange for Amazon S3 supports only the `ADFS 3.0` identity provider. Select `None` if you do not want to use federated user single sign-on.

Federated user single sign-on connection properties

Configure the following properties when you select `ADFS 3.0` in **Federated SSO IdP**:

Property	Description
Federated User Name	User name of the federated user to access the AWS account through the identity provider.
Federated User Password	Password for the federated user to access the AWS account through the identity provider.
IdP SSO URL	Single sign-on URL of the identity provider for AWS.
SAML Identity Provider ARN	ARN of the SAML identity provider that the AWS administrator created to register the identity provider as a trusted provider.
Role ARN	ARN of the IAM role assumed by the federated user.

Rules and Guidelines for Scality RING-S3 Compatible Storage

Consider the following rules and guidelines when you use Scality RING as the S3 compatible storage:

- You use Scality RING as the S3 compatible storage when you run a mapping in the native environment and on the Spark engine.
- You cannot configure encryption when you use Scality RING as the S3 compatible storage.
- You can only configure basic authentication when you use Scality RING.
- You cannot add an object tag to the object stored on the Amazon S3 bucket.
- You cannot use Scality RING with Redshift sources and targets.

Creating an Amazon S3 Connection

Create an Amazon S3 connection before you create an Amazon S3 data object.

1. In the Developer tool, click **Window** > **Preferences**.
2. Select **Informatica** > **Connections**.
3. Expand the domain in the **Available Connections**.
4. Select the connection type **File System** > **Amazon S3**, and click **Add**.
5. Enter a connection name and an optional description.
6. Select Amazon S3 as the connection type.
7. Click **Next**.
8. Configure the connection properties.
9. Click **Test Connection** to verify the connection to Amazon S3.
10. Click **Finish**.

CHAPTER 4

PowerExchange for Amazon S3 Data Objects

This chapter includes the following topics:

Amazon S3 Data Object Overview

An Amazon S3 data object is a physical data object that uses Amazon S3 as a source or target. An Amazon S3 data object is the physical data object that represents data based on an Amazon S3 resource.

You can configure the data object read and write operation properties that determine how data can be read from Amazon S3 sources and loaded to Amazon S3 targets.

Create an Amazon S3 data object from the Developer tool. PowerExchange for Amazon S3 creates the data object read operation and data object write operation for the Amazon S3 data object automatically. You can edit the advanced properties of the data object read or write operation and run a mapping.

Note: To view the list of files available in a bucket, you must select the bucket name instead of expanding the bucket name list in the **Object Explorer** view.

When you use the **Data Preview** option for Amazon S3 data objects, set the Unicode characters `LANG` and `LC_ALL` to `en_US.UTF-8`. Otherwise, junk characters are displayed.

Amazon S3 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 Amazon S3 data objects:

Property	Description
Name	Name of the Amazon S3 data object.
Location	The project or folder in the Model Repository Service where you want to store the Amazon S3 data object.
Connection	Name of the Amazon S3 connection.
Resource Format	You can create an Amazon S3 file data object from the following data source in Amazon S3: - Avro. Applicable when you run a mapping in the native environment or on the Spark and Databricks Spark engine. - Binary. Applicable when you run a mapping in the native environment. - Flat - JSON. Applicable when you run a mapping on the Spark or Databricks Spark engine. - ORC. Applicable when you run a mapping in the native environment or on the Spark and Databricks Spark engine. - Parquet. Applicable when you run a mapping in the native environment or on the Spark and Databricks Spark engine. - Intelligent Structure Model. It reads any format that an intelligent structure parses. Applicable when you run a mapping on the Spark or Databricks Spark engine. You must choose the appropriate source format to read data from the source or write data to the target. Default is binary.

Amazon S3 Data Object Read Operation

Create a mapping with an Amazon S3 data object read operation to read data from Amazon S3.

You can download Amazon S3 files in multiple parts, specify the location of the staging directory, and compress the data when you read data from Amazon S3.

Directory Source in Amazon S3 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 an Amazon S3 data object read operation:

- File
- Directory

Note: Applicable when you run a mapping in the native environment or on the Spark and Databricks Spark engine.

You must select the source file during the data object creation to select the source type as **Directory** at the run time. PowerExchange for Amazon S3 provides the option to override the value of the **Folder Path** and **File Name** properties during run time. When you select the **Source Type** option as **Directory**, the value of the **File Name** is not honored.

For read operation, if you provide the **Folder Path** value during run time, the Data Integration Service considers the value of the **Folder Path** from the data object read operation properties. If you do not provide the **Folder Path** value during run time, the Data Integration Service considers the value of the **Folder Path** that you specify during the data object creation.

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.

When you run a mapping to read multiple files and if the Amazon S3 data object is defined using file with header option on the Spark or Databricks Spark engine, the mapping runs successfully. However, the Data Integration Service does not generate a validation error for the files with no header.

Working with Multiple Files

You can read multiple files, which are of flat format type, from Amazon S3 and write data to a target in the native environment.

To read multiple flat files in the native environment, all files must be available in the same Amazon S3 bucket. When you want to read from multiple folders in the Amazon S3 bucket, you must create a `.manifest` file that contains all the source files with the respective absolute path or directory path. You must specify the `.manifest` file name in the following format: `<file_name>.manifest`

For example, the `.manifest` file contains source files in the following format:

```
{
    "fileLocations":
    [
        {
            "URIs":
            [
                "dir1/dir2/dir3/file_1.csv"
                "dir1/dir2/dir3/file_2.csv"
                "dir1/file_3.csv"
            ]
        },
        {
            "URIPrefixes":
            [
                "dir1/dir2/dir3/"
                "dir1/dir2/dir4/"
            ]
        },
        {
            "WildcardURIs":
            [
                "dir1/dir2/dir3/dir5*/*/*.csv"
                "dir1/dir2/dir3/dir6/dir7/*"
            ]
        }
    ],
    "settings":
    {
        "stopOnFail": "true"
    }
}
```

You can specify URIs, URIPrefixes, WildcardURIs, or all sections within fileLocations in the `.manifest` file.

In **Data Preview**, the data of the first file is available in the URI specified in the `.manifest` file. If the URI section is empty, the first file in the folder specified in URIPrefixes is displayed.

You can specify an asterisk (*) wildcard in the file name, which are of flat format type, to fetch files from the Amazon S3 bucket. You can specify the asterisk (*) wildcard to fetch all the files or only the files that match the name pattern. The asterisk (*) wildcard is applicable at the folder and file level in a single bucket. Specify the wildcard character in the following format:

```
abc*.txt
abc.*
```

For example, if you specify `abc*.txt`, all the file names starting with the term `abc` and ending with the `.txt` file extension are read. If you specify `abc.*`, all the file names starting with the term `abc` are read regardless of the extension.

Use the wildcard character to specify files from a single folder.

You cannot use the wildcard characters to specify folder names. For example,

```
{ "WildcardURIs": [ "multiread_wildcard/dir1*/", "multiread_wildcard/*/" ] }
```

Note: PowerExchange for Amazon S3 supports only asterisk (*) wildcard character.

Wildcard Character Overview

When you run a mapping to read data from an Avro, flat, JSON, ORC, or Parquet file, you can use wildcard characters to specify the source file name.

To use wildcard characters for the source file name, select the **Allow Wildcard Characters** option in the advanced read properties of the Amazon S3 data object.

Wildcard Characters for Reading Data from Flat Files

When you run a mapping in the native environment to read data from a flat, you can use an asterisk (*) wildcard character to specify the source file name.

You can use the asterisk (*) wildcard to fetch all the files or only the files that match the name pattern. Specify the wildcard character in the following format:

```
abc*.txt
abc.*
```

For example, if you specify `abc*.txt`, all the file names starting with the term `abc` and ending with the `.txt` file extension are read. If you specify `abc.*`, all the file names starting with the term `abc` are read regardless of the extension.

You cannot use the wildcard characters to specify folder names. For example,

```
{ "WildcardURIs": [ "multiread_wildcard/dir1*/", "multiread_wildcard/*/" ] }
```

Wildcard Characters for Reading Data from Complex Files

When you run a mapping in the native environment or on the Spark and Databricks Spark engine to read data from an Avro, JSON, ORC, or Parquet file, you can use an asterisk (?) and (*) wildcard characters to specify the source file name.

You can use the following wildcard characters:

? (Question mark)

The question mark character (?) allows one occurrence of any character. For example, if you enter the source file name as `a?b.txt`, the Data Integration Service reads data from files with the following names:

- `a1b.txt`

- a2b.txt
- aab.txt
- acb.txt

*** (Asterisk)**

The asterisk mark character (*) allows zero or more than one occurrence of any character. If you enter the source file name as a*b.txt, the Data Integration Service reads data from files with the following names:

- aab.txt
- a1b.txt
- ab.txt
- abc11b.txt

Note: When you read data from the Avro, JSON, ORC, or Parquet file that contains a colon (:) character in the file name, the mapping fails.

Amazon S3 Data Object Read Operation Properties

Amazon S3 data object read operation properties include run-time properties that apply to the Amazon S3 data object.

The Developer tool displays advanced properties for the Amazon S3 data object operation in the **Advanced** view. The following table describes the advanced properties for an Amazon S3 data object read operation:

Property	Description
Source Type	Select the type of source from which you want to read data. You can select the following source types: - File - Directory Default is **File**. Applicable when you run a mapping in the native environment or on the Spark and Databricks Spark engine. For more information about source type, see "Directory Source in Amazon S3 Sources" on page 21.
Folder Path	Bucket name or folder path of the Amazon S3 source file that you want to overwrite. If applicable, include the folder name that contains the source file in the `<bucket_name>/<folder_name>` format. If you do not provide the bucket name and specify the folder path starting with a slash (/) in the `/<folder_name>` format, the folder path appends with the folder path that you specified in the connection properties. For example, if you specify the `<my_bucket1>/<dir1>` folder path in the connection property and `/<dir2>` folder path in this property, the folder path appends with the folder path that you specified in the connection properties in `<my_bucket1>/<dir1>/<dir2>` format. If you specify the `<my_bucket1>/<dir1>` folder path in the connection property and `<my_bucket2>/<dir2>` folder path in this property, the Data Integration Service reads the file from the `<my_bucket2>/<dir2>` folder path that you specify in this property.
File Name	Name of the Amazon S3 source file that you want to overwrite.
Allow Wildcard Characters	Indicates whether you want to use wildcard characters for the source file name. When you run a mapping in the native environment to read a flat file and select this option, you can use the `*` wildcard character for the source file name. When you run a mapping in the native environment or on the Spark and Databricks Spark engine to read an Avro, JSON, ORC, or Parquet file and select this option, you can use the `?` and `*` wildcard characters for the source file name. The question mark character (?) allows one occurrence of any character. The asterisk character (*) allows zero or more than one occurrence of any character.

Property	Description
Staging Directory	Amazon S3 staging directory. Ensure that the user has write permissions on the directory. In addition, ensure that there is sufficient space to enable staging of the entire file. Default staging directory is the temporary directory on the machine that hosts the Data Integration Service. **Note:** Applicable when you run a mapping in the native environment.
Hadoop Performance Tuning Options	Provide semicolon separated name-value attribute pairs to optimize performance when you copy large volumes of data between Amazon S3 and HDFS. **Note:** Applicable to the Amazon EMR cluster. For more information about Hadoop performance tuning options, see " Hadoop Performance Tuning Options for EMR Distribution" on page 44.
Compression Format	Decompresses data when you read data from Amazon S3. You can decompress the data in the following formats: - **None**. Select **None** to decompress files with the deflate, snappy, and zlib formats. - **Bzip2** - **Gzip** - **Lzo** Default is None. You can read files that use the deflate, snappy, zlib, Gzip, and Lzo compression formats in the native environment or on the Spark and Databricks Spark engine. You can read files that use the Bzip2 compression format on the Spark engine. For more information about compression formats, see "Data Compression in Amazon S3 Sources and Targets" on page 42.
Download Part Size	Downloads an Amazon S3 object in multiple parts. Default is 5 MB. When the file size of an Amazon S3 object is greater than 8 MB, you can choose to download the object in multiple parts in parallel. By default, the Data Integration Service downloads the file in multiple parts. **Note:** Applicable when you run a mapping in the native environment.

Property	Description
Multiple Download Threshold	Minimum threshold size to download an Amazon S3 object in multiple parts. Default is 10 MB. To download the object in multiple parts in parallel, you must ensure that the file size of an Amazon S3 object is greater than the value you specify in this property. **Note:** Applicable when you run a mapping in the native environment.
Temporary Credential Duration	The time duration during which an IAM user can use the dynamically generated temporarily credentials to access the AWS resource. Enter the time duration in seconds. Default is 900 seconds. If you require more than 900 seconds, you can set the time duration maximum up to 12 hours in the AWS console and then enter the same time duration in this property.

Schema Properties

The Developer tool displays schema properties for Amazon S3 file sources in the **Data Object Operations Details** window.

The following table describes the schema properties that you configure for Amazon S3 file sources:

Property	Description
Column Name	Displays the name of the column.
Column Type	Displays the format of the column.
Enable Column Projection	Displays the column details of the complex files sources.
Schema Format	Displays the schema format that you selected while creating the complex file data object. You can change the schema format. You can also parameterize the schema format. You can select one of the following Amazon S3 file formats when you run the mapping in the native environment or on the Spark and Databricks Spark engine: - Flat - Avro - ORC - Parquet - Intelligent Structure Model You can change the complex file format without losing the column metadata even after you configure the column projection properties for another complex file format. **Note:** You can switch from one schema format to another only once. If you change the schema format more than once, you might lose the original data types.

Property	Description
Schema	Displays the schema associated with the complex file. You can select a different schema. You can parameterize the schema or the schema path. For more information, see "Specifying the schema file format" on page 28. To parameterize the schema path, obtain the path from the server. When you use Refresh Schema for the source or target in a mapping and also, parameterize the schema, the parameterized schema takes precedence over the refresh schema. **Note:** If you disable the column projection, the schema associated with the complex file is removed. If you want to associate schema again with the complex file, enable the column projection and select schema.
Schema Properties	Applicable only to flat files.
Column Mapping	Displays the mapping between input and output ports. **Note:** If you disable the column projection, the mapping between input and output ports is removed. If you want to map the input and output ports, enable the column projection and select schema to associate a schema to the complex file.

Specifying the schema file format

You can override the default schema and specify the schema for Flat, Avro, ORC, and Parquet files. Specify the schema in the following formats:

- Flat

```
{"FileName":"<FilePath>\\<FileName>","Columns":
[{"Type":"number","Precision":2,"Scale":0,"Name":"N_NATIONKEY"},
{"Type":"string","Precision":14,"Scale":0,"Name":"N_NAME"},
{"Type":"number","Precision":1,"Scale":0,"Name":"N_REGIONKEY"},
{"Type":"string","Precision":112,"Scale":0,"Name":"N_COMMENT"}]}
```

- Avro

```
 {"type":"record","name":"InfaRecord","fields":[{ "name":"C_CUSTKEY" , "type":
[{"type":"bytes","logicalType":"decimal","precision":3,"scale":0},"null"]} ,
{ "name":"C_NAME" , "type": ["string", "null"] } , { "name":"C_ADDRESS" , "type":
["string", "null"] } , { "name":"C_PHONE" , "type": ["string", "null"] } ,
{ "name":"C_ACCTBAL" , "type": ["string", "null"] } , { "name":"C_MKTSEGMENT" ,
"type": ["string", "null"] } , { "name":"C_COMMENT" , "type": ["string", "null"] }]}
```

- ORC

```
message AmazonS3_Data_Object { optional   INT32 C_CUSTKEY (DECIMAL(3,0));optional
binary C_NAME (UTF8);optional binary C_ADDRESS (UTF8);optional   INT32 C_NATIONKEY
(DECIMAL(2,0));optional binary C_COMMENT (UTF8); }
```

- Parquet

```
message S3_Parquet_1_Level { optional   INT32 C_CUSTKEY (DECIMAL(3,0));optional
binary C_NAME (UTF8);optional binary C_ADDRESS (UTF8);optional binary C_PHONE
(UTF8);optional binary C_ACCTBAL (UTF8);optional binary C_MKTSEGMENT (UTF8);optional
binary C_COMMENT (UTF8); }
```

Flat File Schema Properties

You can configure format properties for a flat file that is delimited.

The following table describes the file format and column format properties that you configure for a flat file:

Property	Description
Maximum row to preview	Number of rows to show in data preview. Default is 0.
Delimiters	Character used to separate columns of data. Default is comma. If you enter a delimiter that is the same as the escape character or the text qualifier, you might receive unexpected results. You cannot specify a multibyte character as a delimiter.
Text Qualifier	Quote character that defines the boundaries of text strings. Default is double quotes. If you select a quote character, the Developer tool ignores delimiters within pairs of quotes.
Qualifier Mode	Qualifier behavior for the source object. You can select one of the following options: - **Minimal**. Default mode. Applies qualifier to data that have a delimiter value or a special character present in the data. Otherwise, the Data Integration Service does not apply the qualifier. - **All**. Applies qualifier to all data.
Row Delimiter	Character used to separate the rows of data. You must select the default value, `\012 LF (\n)`.
Header Line Number	Line number that you want to use as the header. You can also read a data from a file that does not have a header. To read data from a file with no header, specify the value of the Header Line Number field as 0.
First Data Row	Line number from where you want the Data Integration Service to read data.
Escape Character	Character immediately preceding a column delimiter character embedded in an unquoted string, or immediately preceding the quote character in a quoted string. When you specify an escape character, the Data Integration Service reads the delimiter character as a regular character. Default is backslash (\).
Retain Escape Character in Data	Not applicable.

Note: If you update the flat file format properties during the data object import and want to see the updated format properties in Data Preview, you must parse the flat file again by selecting the **Schema** property.

Parameterize Column Format Properties

You can parameterize the column format properties for flat files in a parameter file. The following table describes property strings that you can use in a parameter file:

Property	Value
maxRowsToPreview	A positive integer value. Default is 0.
delimiter	Specify the octal code for the character. Preface the octal code with a backslash (\). Specify the following values for the given delimiters: - `\011 TAB` for Tab - `;` for semicolon - `,` for comma - `\040 SP` for space Default is comma. You cannot specify a multibyte character as a delimiter.
textQualifier	Specify the following string values: - SINGLE_QUOTES for ' - DOUBLE_QUOTES for " - NO_QUOTES Default is double quotes.
importColumnFromFirstLine	True or false. Default is true.
rowDelimiter	Default value, `\012 LF (\n)`.
escapeCharacter	A string value. Default is `\`.

A Sample JSON Parameter File

```
{ "maxRowsToPreview": 10, "delimiter": ";", "textQualifier": "SINGLE_QUOTES",
"importColumnFromFirstLine":true, "escapeCharacter" : "-" }
```

Amazon S3 Data Object Write Operation

Create a mapping to write data to Amazon S3. Change the connection to an Amazon S3 connection, and define the write operation properties to write data to Amazon S3.

There is no control over the number of files created or file names written to the directory on the Spark or Databricks Spark engine. The Data Integration Service writes data to multiple files based on the source or source file size to the directory provided. You must provide the target file name and based on target file name, the Data Integration Service adds suffix characters such as MapReduce or Split information to the target file name.

If the file size is greater than 256 MB, the Data Integration Service creates multiple files inside the target folder. For example, `output.txt-m-00000`, `output.txt-m-00001`, and `output.txt-m-00002`.

Amazon S3 Data Encryption

To protect data, you can enable server-side encryption or client-side encryption to encrypt data inserted in Amazon S3 buckets.

You can encrypt data by using the master symmetric key or customer master key. Do not use the master symmetric key and customer master key together.

You can select type that you want to use to encrypt data in the **Encryption Type** advanced properties for the data object write operation. You can select the following encryption type you want to use to encrypt data:

None

The data is not encrypted.

Server Side Encryption

Select **Server Side Encryption** as the encryption type if you want Amazon S3 encrypts data using Amazon S3-managed encryption keys when you write the files to the target.

Server Side Encryption With KMS

If you select **Server Side Encryption With KMS** as the encryption type, the Unload command creates the staging files in the Amazon S3 bucket and Amazon S3 encrypts the file using AWS KMS-managed customer master key or Amazon Resource Name (ARN) for server-side encryption.

The AWS KMS-managed customer master key specified in the connection property must belong to the same region where Amazon S3 is hosted. For example, if Amazon S3 is hosted in the **US West (Oregon)** region, you must use the AWS KMS-managed customer master key enabled in the same region when you select the **Server Side Encryption With KMS** encryption type.

Note: You cannot use the **Server Side Encryption With KMS** encryption type on MapR 6.1 distribution.

Client Side Encryption

Select **Client Side Encryption** as the encryption type if you want the Data Integration Service to encrypt the data when you write the files to the target. Client-side encryption uses a master symmetric key or customer master key that you specify in the Amazon S3 connection properties to encrypt data.

Note: Applicable only when you run a mapping in the native environment.

If you specify both the master symmetric key and customer master key ID in the connection properties, and select a client-side encryption, the Data Integration Service uses the customer master key ID to encrypt data.

To enable client-side encryption, perform the following tasks:

1. Ensure that an organization administrator creates a master symmetric key or customer master key ID when you create an Amazon S3 connection.

   Note: The administrator user of the account can use the default customer master key ID to enable the client-side encryption.
2. Select **Client Side Encryption** as the encryption type in the advanced properties of the data object write operation.
3. Ensure that an organization administrator updates the security JAR files, required by the Amazon S3 client encryption policy, on the machine that hosts the Data Integration Service.

Note: When you select a client-side encryption and run a mapping to read or write an Avro, ORC, or Parquet file, the mapping runs successfully. However, the Data Integration Service ignores the client-side encryption.

The following table lists the encryption types for the support for various environments:

Encryption Type	Native Environment	Blaze Environment	Databricks Environment	Spark Environment
Server Side Encryption	Yes	Yes	Yes	Yes
Client Side Encryption	Yes	No	No	No
Server Side Encryption With KMS	Yes	No	Yes	Yes

For information about the Amazon S3 client encryption policy, see the *Amazon S3 documentation*.

Overwriting Existing Files

You can choose to overwrite the existing files.

Select the **Overwrite File(s) If Exists** option in the Amazon S3 data object write operation properties to overwrite the existing files. By default, the value of the **Overwrite File(s) If Exists** check box is true.

If you select the **Overwrite File(s) If Exists** option, the Data Integration Service deletes the existing files with same file name and creates a new files with the same file name in the target directory.

If you do not select the **Overwrite File(s) If Exists** option, the Data Integration Service does not delete the existing files in the target directory. The Data Integration Service adds the timestamp, process ID, and thread ID at the end of each target file name in the following format:

`YYYYMMDD_HHMMSS_processID-YYYYMMDD_HHMMSS_threadID.`

For example, the Data Integration Service renames the target file name in the following format:

`output.txt-20210616_175627_85354941112-20210616_175632_88554941121.`

If you select the **Overwrite File(s) If Exists** option on the Spark or Databricks Spark engine, the Data Integration Service splits the existing files into multiple files with same file name. Then the Data Integration Service deletes the split files and creates new files in the target directory.

When you select the **Overwrite File(s) If Exists** option to overwrite an Avro file on the Spark or Databricks Spark engine, the Data Integration Service overwrites the existing file and appends `_avro` to the folder name. For example, `targetfile_avro`

Object Tag

You can add a tag to the object stored on the Amazon S3 bucket when you run a mapping in the native environment or on the Spark and Databricks Spark engine. Each tag contains a key value pair. Tagging an object helps to categorize the storage.

You can add the object tags in the **Object Tags** field under the advanced properties of the data object write operation. Enter the object tag in the `Key=Value` format. You can also enter multiple object tags in the following format:

```
key1=Value1;key2=Value2
```

You can either enter the key value pairs or the specify the file path that contains the key value pairs. For example, you can specify the file path in the `C:\object\tags.txt` format. You can specify any file path on which the Data Integration Service is installed.

When you upload new objects in the Amazon S3 bucket, you can add tags to the new objects or add tags to the existing objects. If the Data Integration Service overrides a file that contains a tag in the Amazon S3 bucket, the tag is not retained. You must add a new tag for the overridden file. If you upload multiple files to the Amazon S3 bucket, each file that you upload must have the same set of tags associated with the multiple objects.

To add tags in the Amazon S3 target object, you must add the `s3:PutObjectTagging` permission in the Amazon S3 policy. Following is the sample policy:

```
{
"Version": "2012-10-17",
"Id": "Policy1500966932533",
"Statement": [
{
"Sid": "Stmt1500966903029",
"Effect": "Allow",
"Action": [
"s3:DeleteObject",
"s3:GetObject",
"s3:ListBucket",
"s3:PutObject",
"s3:PutObjectTagging"
],
"Resource": [
"arn:aws:s3:::<bucket_name>/*",
"arn:aws:s3:::<bucket_name>"
]
}
]
}
```

The following table lists the special characters that PowerExchange for Amazon S3 supports during entering the key value pair:

Special Characters	Support
+	Yes
-	Yes
=	No
.	Yes
_	Yes
:	Yes
/	Yes

Amazon S3 Data Object Write Operation Properties

Amazon S3 data object write operation properties include run-time properties that apply to the Amazon S3 data object.

The Developer tool displays advanced properties for the Amazon S3 data object operation in the **Advanced** view.

Note: By default, the Data Integration Service uploads the Amazon S3 file in multiple parts.

The following table describes the Advanced properties for an Amazon S3 data object write operation:

Property	Description
Overwrite File(s) If Exists	Overwrite the existing files. Select the check box if you want to overwrite the existing files. Default is true. For more information Overwrite File(s) If Exists, see "Overwriting Existing Files" on page 32.
Folder Path	Bucket name or folder path of the Amazon S3 source file that you want to overwrite. If applicable, include the folder name that contains the target file in the `<bucket_name>/<folder_name>` format. If you do not provide the bucket name and specify the folder path starting with a slash (/) in the `/<folder_name>` format, the folder path appends with the folder path that you specified in the connection properties. For example, if you specify the `<my_bucket1>/<dir1>` folder path in the connection property and `/<dir2>` folder path in this property, the folder path appends with the folder path that you specified in the connection properties in `<my_bucket1>/<dir1>/<dir2>` format. If you specify the `<my_bucket1>/<dir1>` folder path in the connection property and `<my_bucket2>/<dir2>` folder path in this property, the Data Integration Service writes the file in the `<my_bucket2>/<dir2>` folder path that you specify in this property.
File Name	Name of the Amazon S3 source file that you want to overwrite. **Note:** When you run a mapping on the Blaze engine to write data to a target, do not use a semi-colon in file name to run the mapping successfully.
Encryption Type	Method you want to use to encrypt data. Select one of the following values: - None - Client Side Encryption - Server Side Encryption - Server Side Encryption with KMS For more information, see "Amazon S3 Data Encryption" on page 31.
Staging Directory	Amazon S3 staging directory. Ensure that the user has write permissions on the directory. In addition, ensure that there is sufficient space to enable staging of the entire file. Default staging directory is the `/temp` directory on the machine that hosts the Data Integration Service. **Note:** Applicable when you run a mapping in the native environment.
File Merge	Merges the target files into a single file. **Note:** Applicable when you run a mapping on the Blaze engine.
Hadoop Performance Tuning Options	Provide semicolon separated name-value attribute pairs to optimize performance when you copy large volumes of data between Amazon S3 and HDFS. Applicable to the Amazon EMR cluster. **Note:** Applicable when you run a mapping in the native environment. For more information about Hadoop performance tuning options, see " Hadoop Performance Tuning Options for EMR Distribution" on page 44.

Property	Description
Compression Format	Compresses data when you write data to Amazon S3. You can compress the data in the following formats: - **None** - **Bzip2** - **Deflate** - **Gzip** - **Lzo** - **Snappy** - **Zlib** Default is None. You can write files that use the deflate, Gzip, snappy, Lzo, and zlib compression formats in the native environment or on the Spark and Databricks Spark engine.. You can write files that use the Bzip2 compression format on the Spark engine. For more information about compression formats, see "Data Compression in Amazon S3 Sources and Targets" on page 42.
Object Tags	Add single or multiple tags to the objects stored on the Amazon S3 bucket. You can either enter the key value pairs or specify the file path that contains the key value pairs. **Note:** Applicable when you run a mapping in the native environment or on the Spark and Databricks Spark engine to write a flat file to the target. For more information about the object tags, see "Object Tag" on page 32.
TransferManager Thread Pool Size	The number of threads to write data in parallel. Default is 10. PowerExchange for Amazon S3 uses the AWS TransferManager API to upload a large object in multiple parts to Amazon S3. When the file size is more than 5 MB, you can configure multipart upload to upload object in multiple parts in parallel. If you set the value of the **TransferManager Thread Pool Size** to greater than 50, the value reverts to 50. **Note:** Applicable when you run a mapping in the native environment to write a flat file to the target.
Part Size	The part size in bytes to upload an Amazon S3 object. Default is 5 MB. **Note:** Applicable when you run a mapping in the native environment to write a flat file to the target.
Temporary Credential Duration	The time duration during which an IAM user can use the dynamically generated temporarily credentials to access the AWS resource. Enter the time duration in seconds. Default is 900 seconds. If you require more than 900 seconds, you can set the time duration maximum up to 12 hours in the AWS console and then enter the same time duration in this property.
Stream Rollover Size in GB	Applicable to the streaming mappings.
Stream Rollover Time in hours	Applicable to the streaming mappings.
Interim Directory	Applicable to the streaming mappings.

Property	Description
Partition Option	Select one of the following partition options when you configure a dynamic mapping: - None. Partitioning is not configured. - Last N Columns Partitioned. The last N columns are selected for partitioning. - Partition Column Names. Comma-separated column names are selected for partitioning.
Partition Arguments	The number or names of partition columns. If you selected **None** as the partition option, do not specify a partition argument. If you selected **Last N Columns Partitioned** as the partition option, specify an integer value as the partition argument. If you selected **Partition Column Names** as the partition option, specify comma-separated column names as the partition argument.

Schema Properties

The Developer tool displays the column projection properties for Amazon S3 file targets in the **Data Object Operations Details** window.

The following table describes the schema properties that you configure for Amazon S3 file targets:

Property	Description
Column Name	Displays the name of the column.
Column Type	Displays the format of the column.
Enable Column Projection	Displays the column details of the complex files sources.
Schema Format	Displays the schema format that you selected while creating the complex file data object. You can change the schema format. You can also parameterize the schema format. You can select one of the following Amazon S3 file formats when you run the mapping in the native environment or on the Spark and Databricks Spark engine: - Flat - Avro - ORC - Parquet You can change the complex file format without losing the column metadata even after you configure the column projection properties for another complex file format. **Note:** You can switch from one schema format to another only once. If you change the schema format more than once, you might lose the original datatypes.
Schema	Displays the schema associated with the complex file. You can select a different schema. You can parameterize the schema or the schema path. For more information, see "Specifying the schema file format" on page 28. To parameterize the schema path, obtain the path from the server. When you use Refresh Schema for the source or target in a mapping and also, parameterize the schema, the parameterized schema takes precedence over the refresh schema. **Note:** If you disable the column projection, the schema associated with the complex file is removed. If you want to associate schema again with the complex file, enable the column projection and select schema.

Property	Description
Schema Properties	Applicable only to flat files.
Column Mapping	Displays the mapping between input and output ports. **Note:** If you disable the column projection, the mapping between input and output ports is removed. If you want to map the input and output ports, enable the column projection and click **Select Schema** to associate a schema to the complex file.

Specifying the schema file format

You can override the default schema and specify the schema for Flat, Avro, ORC, and Parquet files. Specify the schema in the following formats:

- Flat

```
{"FileName":"<FilePath>\\<FileName>","Columns":
[{"Type":"number","Precision":2,"Scale":0,"Name":"N_NATIONKEY"},
{"Type":"string","Precision":14,"Scale":0,"Name":"N_NAME"},
{"Type":"number","Precision":1,"Scale":0,"Name":"N_REGIONKEY"},
{"Type":"string","Precision":112,"Scale":0,"Name":"N_COMMENT"}]}
```

- Avro

```
 {"type":"record","name":"InfaRecord","fields":[{ "name":"C_CUSTKEY" , "type":
[{"type":"bytes","logicalType":"decimal","precision":3,"scale":0},"null"]} ,
{ "name":"C_NAME" , "type": ["string", "null"] } , { "name":"C_ADDRESS" , "type":
["string", "null"] } , { "name":"C_PHONE" , "type": ["string", "null"] } ,
{ "name":"C_ACCTBAL" , "type": ["string", "null"] } , { "name":"C_MKTSEGMENT" ,
"type": ["string", "null"] } , { "name":"C_COMMENT" , "type": ["string", "null"] }]}
```

- ORC

```
message AmazonS3_Data_Object { optional   INT32 C_CUSTKEY (DECIMAL(3,0));optional
binary C_NAME (UTF8);optional binary C_ADDRESS (UTF8);optional   INT32 C_NATIONKEY
(DECIMAL(2,0));optional binary C_COMMENT (UTF8); }
```

- Parquet

```
message S3_Parquet_1_Level { optional   INT32 C_CUSTKEY (DECIMAL(3,0));optional
binary C_NAME (UTF8);optional binary C_ADDRESS (UTF8);optional binary C_PHONE
(UTF8);optional binary C_ACCTBAL (UTF8);optional binary C_MKTSEGMENT (UTF8);optional
binary C_COMMENT (UTF8); }
```

Flat File Schema Properties

You can configure format properties for a flat file that is delimited.

The following table describes the file format and column format properties that you configure for a flat file:

Property	Description
Maximum row to preview	Number of rows to show in data preview. Default is 0.
Delimiters	Character used to separate columns of data. Default is comma. If you enter a delimiter that is the same as the escape character or the text qualifier, you might receive unexpected results. You cannot specify a multibyte character as a delimiter.
Text Qualifier	Quote character that defines the boundaries of text strings. Default is double quotes. If you select a quote character, the Developer tool ignores delimiters within pairs of quotes.

Property	Description
Qualifier Mode	Qualifier behavior for the target object. You can select one of the following options: - **Minimal**. Applies the qualifier to data that contains either a delimiter value or a special character. Otherwise, the Data Integration Service does not apply the qualifier. - **All**. Applies qualifier to all data. Default is minimal.
Row Delimiter	Character used to separate the rows of data. The default value is `\012 LF (\n)`.
Target Header	Indicates whether you want to write data with or without a header.
Escape Character	Character immediately preceding a column delimiter character embedded in an unquoted string, or immediately preceding the quote character in a quoted string. When you specify an escape character, the Data Integration Service reads the delimiter character as a regular character. Default is backslash (\).
Retain Escape Character in Data	Not applicable.

Note: If you update the flat file format properties during the data object import and want to see the updated format properties in Data Preview, you must parse the flat file again by selecting the **Schema** property.

Parameterize Column Format Properties

You can parameterize the column format properties for flat files in a parameter file. The following table describes property strings that you can use in a parameter file:

Property	Value
maxRowsToPreview	A positive integer value. Default is 0.
delimiter	Specify the octal code for the character. Preface the octal code with a backslash (\). Specify the following values for the given delimiters: - `\011 TAB` for Tab - `;` for semicolon - `,` for comma - `\040 SP` for space Default is comma. You cannot specify a multibyte character as a delimiter
textQualifier	Specify the following string values: - SINGLE_QUOTES for ' - DOUBLE_QUOTES for " - NO_QUOTES Default is double quotes.
rowDelimiter	Default value, `\012 LF (\n)`.
escapeCharacter	A string value. Default is `\`.

A Sample JSON Parameter File

```
{ "maxRowsToPreview": 10, "delimiter": ";", "textQualifier": "SINGLE_QUOTES",
"escapeCharacter" : "-" }
```

Rules and guidelines for writing to a flat file target

Consider the following rules when you run a mapping to write data to a flat file in the native environment or on the Spark engine:

- For a mapping that runs on the Spark engine, the first row of the header in the flat file contains an additional `#` symbol. For example, the header `ID_Char` appears as `#ID_Char`.
- For a mapping that runs in the native environment, the header in the flat file does not contain an additional `#` symbol. For example, the header `ID_Char` appears as `ID_Char`.
- For a mapping that runs on the Spark engine, the double quotes text qualifier is honored. For example, the column name `abcd` with the double quotes text qualifier appears as `"abcd"`.
- For a mapping that runs in the native environment, the double quotes text qualifier is not honored. For example, the column name `abcd` with the double quotes text qualifier appears as `abcd`.

FileName Port Overview

A FileName port is a string port with a default precision of 1024 characters that contains an endpoint name and source path of a file.

You cannot configure the FileName port. You can delete the FileName port if you do not want to read or write the data in the FileName. When you create a data object read or write operation for all the Amazon S3 files, the FileName port is displayed by default.

For a flat file, the FileName port appears when you run a mapping in the native environment. The FileName port for a flat file is not applicable to the non-native environments.

The Data Integration Service reads the FileName port for the Amazon S3 file formats in the following format: `<end point>/<bucket name>/<file name path>/<file>` format.

Here, `<end point>` indicates the region name of the Amazon S3 bucket.

For example, `s3-us-west-2.amazonaws.com /infa.qa.bucket/automation/file.parquet`.

Note: The Data Integration Service displays the value of the endpoint incorrectly when you run a mapping on the Spark engine. For example, the Data Integration Service displays the endpoint value as `s3a` instead of `s3-us-west-2.amazonaws.com`.

Working with FileName Port

You can use the data in the FileName port when you create a data object read or write operation.

When you run a mapping to read or write an Amazon S3 file using the FileName port, the result varies based on the type of the Amazon S3 file that you use and the engine where you run the mapping. For example, when you run a mapping in the native environment to read or write an Avro, Binary, JSON, ORC, or Parquet file, the Data Integration Service appends a period (.) to the directory name and appends an underscore (_) to the directory name when you run a mapping on the Spark engine.

When you run a mapping in the native environment to read or write a flat file using the FileName port, the Data Integration Service creates separate files for each entry in the FileName port in the following format:

<valueoftheNativeNamepropertyorFileNameDataObjectWriteOperation>=<valueComingToFileNamePort>

When you run a mapping in the native environment or on the Spark and Databricks Spark engine to read or write an Avro, Binary, JSON, ORC, or Parquet file using the FileName port, the Data Integration Service creates separate directories for each entry in the FileName port and adds the files within the directories in the following format:

- On the Spark and Databricks Spark engine, the Data Integration Service creates the directory in the <valueoftheNativeNamepropertyorFileNameDataObjectWriteOperation>_<fileextention> format and creates the file in the <valueoftheNativeNamepropertyorFileNameDataObjectWriteOperation>_<fileextention>=<valueComingToFileNamePort> format.
- In the native environment, the Data Integration Service creates the directory in the <valueoftheNativeNamepropertyorFileNameDataObjectWriteOperation>.<fileextention> format and creates the file in the <valueoftheNativeNamepropertyorFileNameDataObjectWriteOperation>.<fileextention>=<valueComingToFileNamePort> format.

Creating the Directories in the Same Format

When you run a mapping in the native environment or on the Spark and Databricks Spark engine to read or write an Avro, Binary, JSON, ORC, or Parquet file, you can enable the Data Integration Service to create the directories in the same format.

To create the directories in the same format, you must add the DSparkS3TargetNameWithUnderscore custom property and set the value to true.
Perform the following steps to add and set the DSparkS3TargetNameWithUnderscore custom property to true:

1. In the **Administrator Console**, navigate to the Data Integration Service.

   The **Data Integration Service** page appears.
2. Click the **Processes** tab.

   The **Processes** page appears.
3. Click the pencil icon to edit the custom property in the **Custom Properties** section.

   The **Edit Custom Properties** dialog box appears.
4. Click **New** to add a new custom property.

   The **New Custom Property** dialog box appears.
5. In the **Name** field, enter DSparkS3TargetNameWithUnderscore as the name of the custom property.
6. In the **Value** field, enter true as the value of the custom property.
7. Click **OK**.
8. Restart the Data Integration Service.

You can also add the custom property in the **Custom Properties** section under the **Properties** tab.

Rules and Guidelines for Using FileName Port

Use the following rules and guidelines when you run a mapping to read or write data using the FileName port:

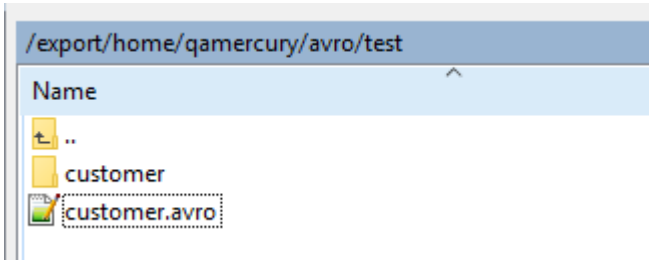
- When you run a mapping to read an Amazon S3 file and if one of the values in the FileName port does not contain any value, the Data Integration Service creates the file in the following format: <valueoftheNativeNamepropertyorFileNameDataObjectWriteOperation>_<fileextention>=<>

However, if you run a mapping to read the newly created file, the mapping fails with the following error message:

```
java.lang.AssertionError: assertion failed: Empty partition column value in '< >'
at scala.Predef$.assert(Predef.scala:170)
```

You must ensure that all entries in the FileName port contains a value to read the newly created file successfully.

- Do not use a colon (:) and forward slash (/) character in the file name data of the FileName port of the source or target object to run a mapping.
- If you connect the FileName port to the target empty zero KB files are created in the target folder.
- When you use wildcard character `*` to read data from a complex file source, the Data Integration Service reads data only from folders or files matching the selection criteria.
  For example, if the file path is `/export/home/qamercury/avro/test/cust*` and **Allow Wildcard Characters** option is selected:



  The Data Integration Service ignores all the other folders and only reads `customer.avro` and the files present inside the customer folder.
- Do not connect FileName port to a FileName port because the FileName port in the source might contain colon (:) and forward slash (/) characters.
- If you create a complex file target in the root directory, map the FileName port to an ID field, and run the mapping in the native environment, the Data Integration Service creates a NULL folder in the root directory and places the target file under the NULL folder.
- When you run a mapping in the native environment to read or write an Amazon S3 file and if there are multiple entries with the same name in the source port, you must use a Sorter transformation.
  Use the Sorter transformation to sort the source port that you want to map to the FileName port of the Target transformation. After you sort the source port, map the port of the Sorter transformation to the FileName port of the Target transformation. The Data Integration Service creates only one file for each entry with the same name. If you do not use the Sorter transformation, the Data Integration Service creates multiple files for each entry with the same name.

  For example, you want to map the following `Employee` source port to the FileName port of the Target transformation and write the data to an Avro target file `target1`:

Name	ID	SSN
Anna	1	1
John	4	4
Smith	4	4

Name	ID	SSN
John	5	5
Anna	2	2

Add a Sorter transformation to sort the source port and map the source port to the port of the Sorter transformation. Then, map the port of the Sorter transformation to the FileName port of the Target transformation. The Data Integration Service creates the following directories along with a single file within each directory:

Directory	Description
`target1.avro=Anna`	In `target1.avro=Anna` directory, the Data Integration Service creates one file with the following value: `1,1,1,2,2,2`.
`target1.avro=John`	In `target1.avro=John` directory, the Data Integration Service creates one file with the following values: `4,4,4,5,5,5`.
`target1.avro=Smith`	In `target1.avro=Smith` directory, the Data Integration Service creates one file with the following values: `4,4,4`.

If you do not add a Sorter transformation, the Data Integration Service creates the following directories along with multiple files within each directories:

Directory	Description
`target1.avro=Anna`	In `target1.avro=Anna` directory, the Data Integration Service creates two files with the following value: `1,1,1` and `2,2,2`
`target1.avro=John`	In `target1.avro=John` directory, the Data Integration Service creates two files with the following value: `4,4,4` and `5,5,5`.
`target1.avro=Smith`	In `target1.avro=Smith` directory, the Data Integration Service creates one file with the following value: `4,4,4`.

Data Compression in Amazon S3 Sources and Targets

You can decompress the data when you read data from Amazon S3 or compress data when you write data to Amazon S3.

Data Compression is applicable when you run a mapping in the native environment or on the Spark and Databricks Spark engine.

Configure the compression format in the **Compression Format** option under the advanced properties for an Amazon S3 data object read and write operation. The source or target file in Amazon S3 contains the same extension that you select in the **Compression Format** option.

When you perform a read operation, the Data Integration Service decompresses the data and then sends the data to Amazon S3 bucket. When you perform a write operation, the Data Integration Service compresses the data.

The following table lists the compression formats for the support for various operations and file formats in the native environment or on the Spark and Databricks Spark engine:

Compression format	Read	Write	Avro File	JSON File	ORC File	Parquet File
None	Yes	Yes	Yes	No	Yes	Yes
Bzip2	No	No	No	Yes	No	No
Deflate	Yes	Yes	Yes	Yes	No	No
Gzip	Yes	Yes	No	Yes	No	Yes
Lzo	Yes	Yes	No	No	No	Yes
Snappy	Yes	Yes	Yes	Yes	Yes	Yes
Zlib	Yes	Yes	No	No	Yes	No

Note: Reading from files that use deflate, snappy, and zlib compression formats is implicit. You must select `None` to read files that use deflate, snappy, and zlib compression formats. For example, to read a parquet file that uses snappy compression, select `None`.

You can compress and decompress a binary file that uses gzip compression.

You can compress or decompress a flat file that uses the none, deflate, gzip, snappy, and zlib compression formats when you run a mapping in the native environment. You can compress or decompress a flat file that use the none, gzip, bzip2, and lzo compression formats when you run a mapping on the Spark engine.

When you run a mapping on the Spark or Databricks Spark engine to write multiple Avro files of different compression formats, the Data Integration Service does not write the data to the target properly. You must ensure that you use the same compression format for all the Avro files.

Note: In the native environment, when you create a mapping to read or write an ORC file and select Lzo as the compression format, the mapping fails.

To read a compressed file from Amazon S3 on the Spark engine, the compressed file must have specific extensions. If the extensions used to read the compressed file are not specific or not valid, the Data Integration Service does not process the file.

The following table describes the extensions that are appended based on the compression format that you use:

Compression Format	File Name Extension
Gzip	.GZ
Deflate	.deflate
Bzip2	.BZ2
Lzo	.LZO

Compression Format	File Name Extension
Snappy	.snappy
Zlib	.zlib

Configuring LZO Compression Format

To write .jar files in the LZO compression format, you must copy the files for LZO compression to the machine where the Data Integration Service runs.

Perform the following steps to configure the Data Integration Service for LZO compression:

1. Copy the lzo.jar file from the cluster to the following directory on the machine on which the Data Integration Service runs: `<Informatica installation directory>/<distribution>/infaLib`
2. Copy the LZO native binaries from the cluster to one of the following directories on the machine on which the Data Integration Service runs:
   - `<Informatica installation directory>/<distribution>/lib/native`
   - `<Informatica installation directory>/<distribution>/lib/native/Linux-amd64-64` for MapR clusters
3. On the Data Integration Service **Processes** tab, add or update the LD_LIBRARY_PATH environment variable to include the path the to LZO native binaries on the Data Integration Service machine.
4. Restart the Data Integration Service.

Hadoop Performance Tuning Options for EMR Distribution

You can use Hadoop Performance Tuning Options to optimize the performance in the Amazon EMR distribution when you copy large volumes of data between Amazon S3 buckets and HDFS.

You must provide semicolon separated name-value attribute pairs for Hadoop Performance Tuning Options.

Use the following parameters for Hadoop Performance Tuning Options:

- `mapreduce.map.java.opts`
- `fs.s3a.fast.upload`
- `fs.s3a.multipartthreshold`
- `fs.s3a.multipartsize`
- `mapreduce.map.memory.mb`

The following sample shows the recommended values for the parameter:

```
mapreduce.map.java.opts=-
Xmx4096m;fs.s3a.fast.upload=true;fs.s3a.multipart.threshold=33554432;fs.s3a.multipart.size=33
554432;mapreduce.map.memory.mb=4096
```

Creating an Amazon S3 Data Object

Create an Amazon S3 data object to add to a mapping.

Note: PowerExchange for Amazon S3 supports only UTF-8 encoding to read or write data.

1. Select a project or folder in the **Object Explorer** view.
2. Click **File** > **New** > **Data Object**.
3. Select **Amazon S3 Data Object** and click **Next**.

   The **Amazon S3 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: to read any format that an intelligent structure parses.
   - Binary: to read any resource format.
   - Flat: to read a flat resource.
   - Avro: to read an Avro resource.
   - ORC: to read an ORC resource.
   - JSON: to read a JSON resource.
   - Parquet: to read a Parquet resource.
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 Amazon S3 connection from which you want to import the Amazon S3 object.
8. To add a resource, click **Add** next to the **Selected Resources** option.

   The **Add Resource** dialog box appears.
9. Select the check box next to the Amazon S3 object you want to add and click **OK**.

   Note: To use an intelligent structure model, select the appropriate `.amodel` file.
10. Click **Next**.
11. Choose **Sample Metadata File**.

    You can click **Browse** and navigate to the directory that contains the file.

    Note: The **Delimited** and **Fixed-width** format properties are not applicable for PowerExchange for Amazon S3.
12. Click **Next**.
13. Configure the format properties.

Property	Description
Delimiters	Character used to separate columns of data. If you enter a delimiter that is the same as the escape character or the text qualifier, you might receive unexpected results. Amazon S3 reader and writer support Delimiters. You cannot specify a multibyte character as a delimiter.
Text Qualifier	Quote character that defines the boundaries of text strings. If you select a quote character, the Developer tool ignores delimiters within pairs of quotes. Amazon S3 reader supports Text Qualifier.

Property	Description
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. The Developer tool prefixes"FIELD_" to field names that are not valid. Amazon S3 reader and writer support Import Column Names From First Line.
Row Delimiter	Specify a line break character. Select from the list or enter a character. Preface an octal code with a backslash (\). To use a single character, enter the character. The Data Integration Service uses only the first character when the entry is not preceded by a backslash. The character must be a single-byte character, and no other character in the code page can contain that byte. Default is line-feed, \012 LF (\n).
Escape Character	Character immediately preceding a column delimiter character embedded in an unquoted string, or immediately preceding the quote character in a quoted string. When you specify an escape character, the Data Integration Service reads the delimiter character as a regular character.

Note: The **Start import at line**, **Treat consecutive delimters as one**, and **Retain escape character in data** properties in the **Column Projection** dialog box are not applicable for PowerExchange for Amazon S3.

14. Click **Next** to preview the flat file data object.
15. Click **Finish**.

    The data object appears under the Physical Data Objects category in the project or folder in the **Object Explorer** view. When you create an Amazon S3 data object, the value of the folder path is displayed incorrectly in the **Resources** tab. Read and write operations are created for the data object. Depending on whether you want to use the Amazon S3 data object as a source or target, you can edit the read or write operation properties.

    Note: Select a read transformation for a data object with an intelligent structure model. You cannot use a write transformation for a data object with an intelligent structure model in a mapping.
16. For a read operation with an intelligent structure model, specify the path to the input file or folder. In the **Data Object Operations** panel, select the **Advanced** tab. In the **File path** field, specify the path to the input file or folder.

Projecting Columns Manually

After sampling the metadata, you can manually edit the projected columns.

Perform the following steps to project columns manually:

1. Go to **Column Projection** tab.
2. Click **Edit Column Projection**.
3. Click **New** icon and add fields manually.

Filtering Metadata

You can filter the metadata to optimize the search performance.

1. Select a project or folder in the **Object Explorer** view.
2. Select an Amazon S3 data object and click **Add**.

3. Click **Next**.
4. Click **Add** next to the **Selected Resources** option.

   The **Add Resource** dialog box appears.
5. Select the bucket or the folder from where you want to search the data.
6. Type the name of the file or any regular expressions in the **Name** field to search for the metadata available in the selected bucket or the folder in the following format: `abc*` or `[0-9]*`.
7. Click **Go**.

   The list of all the file names starting with the alphabet or the number that you entered in the **Name** field is displayed.

Creating an Amazon S3 Data Object Read or Write Operation

You can add an Amazon S3 data object read or write operation to a mapping or mapplet as a source.

Before you create an Amazon S3 data object read or write operation, you must create at least one Amazon S3 data object. You can create the data object read or write operation for one or more Amazon S3 data objects.

Perform the following steps to create an Amazon S3 data object read or write operation:

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 read or write operation.
4. Select **Read** or **Write** as the type of data object operation.
5. Click **Add**.

   The **Select Resources** dialog box appears.
6. Select the Amazon S3 object for which you want to create the data object read or write operation and click **OK**.
7. Click **Finish**.

The Developer tool creates the data object read or write operation for the selected data object.

Rules and Guidelines for Creating an Amazon S3 Data Object Operation

Use the following rules and guidelines when you create an Amazon S3 data object operation:

- In the **Data Object Operations** tab, you can select the **View Operation** link next to the data object read or write operation name to open the operation directly after you create the data object read or write operation.
- When you create a data object read or write operation, you can add new columns or modify the columns in the **Ports** tab directly.
- To modify the columns of a flat file, you must reconfigure the column projection properties.

- To modify the columns of an Avro, ORC, or Parquet file, change the Amazon S3 file format in the **Schema** field of the column projection properties.
- When you create a mapping to read or write an Avro, JSON, ORC, or Parquet file, you can copy the columns of the Source transformations, Target transformations, or any other transformations from the **Ports** tab. Then, you can paste the columns in the data object read or write operation directly.
- When you copy the columns from any transformation to the data object read or write operation, you can change the data type of the columns. The Data Integration Service resets the precision value of the data type to the default value.
  However, the Data Integration Service does not change the precision value of the String data type to the default value.

Creating an Amazon S3 Target

You can create an Amazon S3 target using the **Create Target** option.

1. Select a project or folder in the **Object Explorer** view.
2. Select a source or a transformation in the mapping.
3. Delete the FileName port from the **Ports** properties of the data object read operation and save.

   Note: If you do not want to delete the FileName port, you must add a transformation and map the fields to the Source transformation. Then, right-click on the transformation and select **Create Target** option to create an Amazon S3 target.
4. Right-click on the Source transformation or the transformation and select **Create Target**.
   The **Create Target** dialog box appears.
5. Select **Others** and then select **AmazonS3 Data Object** from the list in the **Data Object Type** section.
6. Click **OK**.
   The **New AmazonS3 Data Object** dialog box appears.
7. Enter a name for the data object.
8. In the **Resource Format** list, select any of the following formats to create the target type:
   - Avro
   - Flat
   - JSON
   - ORC
   - Parquet
9. Click **Finish**.

The new target appears under the **Physical Data Objects** category in the project or folder in the **Object Explorer** view.

Rules and Guidelines for Creating a new Amazon S3 Target

Use the following rules and guidelines when you create a new Amazon S3 target:

- If you right-click on a Source transformation directly to create an Amazon S3 target, the Data Integration Service fails to create an Amazon S3 target with the following error message:

```
Cannot create a AmazonS3 because the transformation contains a port with the name
FileName. FileName is a reserved word in the data object.
```

- You must specify a connection for the newly created Amazon S3 target in the **Connection** field to run a mapping.
- When you write an Avro or Parquet file using the **Create Target** option, you cannot provide a Null data type.
- When you select a flat resource format that contains different data types and select the **Create Target** option to create an Amazon S3 target, the Data Integration Service creates string ports for all the data types with a precision of 256 characters.
- When you select a flat resource format to create an Amazon S3 target, the Data Integration Service maps all the data types in the source file to the String data type in the target file. You must manually map the data types in the source and target files.
- For a newly created Amazon S3 target, the Data Integration Service considers the value of the folder path that you specify in the **Folder Path** connection property and file name from the **Native Name** property in the Amazon S3 data object details.
  Provide a folder path and file name in the Amazon S3 data object read and write advanced properties to overwrite the values.
- When you use a flat resource format to create a target, the Data Integration Service considers the following values for the formatting options:

Formatting Options	Values
Delimiters	Comma (,)
Text Qualifier	No quotes
Import Column Names From First Line	Generates header
Row Delimiter	Backslash with a character n (\n)
Escape Character	Empty

If you want to configure the formatting options, you must manually edit the projected columns.

For more information about editing the projecting columns manually, see "Projecting Columns Manually" on page 46.

Filtering Metadata

You can filter the metadata to optimize the search performance.

1. Select a project or folder in the **Object Explorer** view.
2. Select an Amazon S3 data object and click **Add**.

3. Click **Next**.
4. Click **Add** next to the **Selected Resources** option.

   The **Add Resource** dialog box appears.
5. Select the bucket or the folder from where you want to search the data.
6. Type the name of the file or any regular expressions in the **Name** field to search for the metadata available in the selected bucket or the folder in the following format: `abc*` or `[0-9]*`.
7. Click **Go**.

   The list of all the file names starting with the alphabet or the number that you entered in the **Name** field is displayed.

CHAPTER 5

PowerExchange for Amazon S3 Mappings

This chapter includes the following topics:

PowerExchange for Amazon S3 Mappings Overview

After you create an Amazon S3 data object read or write operation, you can create a mapping.

You can create an Informatica mapping containing an Amazon S3 data object read operation as the input, and a relational or flat file data object operation as the target. You can create an Informatica mapping containing objects such as a relational or flat file data object operation as the input, transformations, and an Amazon S3 data object write operation as the output to load data to Amazon S3 buckets.

Validate and run the mapping. You can deploy the mapping and run it or add the mapping to a Mapping task in a workflow. An Amazon S3 mapping does not read valid rows if there are rows that contain errors in the Amazon S3 source object.

Note: To successfully run a mapping on the Spark and Databricks Spark engine when you select multiple objects from different Amazon S3 buckets, ensure that all the Amazon S3 buckets belong to the same region and use the same credentials to access the Amazon S3 buckets.

Mapping Validation and Run-time Environments

You can validate and run mappings in the native environment or in a non-native environment, such as Hadoop or Databricks.

When you validate a mapping, you can validate it against one or all of the engines. The Developer tool returns validation messages for each engine.

When you run a mapping, you can choose to run the mapping in the native environment or in a non-native environment, such as Hadoop or Databricks. Configure the run-time environment in the Developer tool to optimize mapping performance and process data that is greater than 10 terabytes. When you run mappings in the native environment, the Data Integration Service processes and runs the mapping. When you run mappings in a non-native environment, the Data Integration Service pushes the processing to a compute cluster, such as Hadoop or Databricks.

You can run standalone mappings, mappings that are a part of a workflow in a non-native environment. When you select the Hadoop environment, the Data Integration Service pushes the mapping logic to the Blaze or Spark engine.

Note: When the tracing level is none and you run a mapping on the Spark engine, the Data Integration Service does not log the PowerExchange for Amazon S3 details in Spark logs.

When you select the Databricks environment, the Integration Service pushes the mapping logic to the Databricks Spark engine, the Apache Spark engine packaged for Databricks.

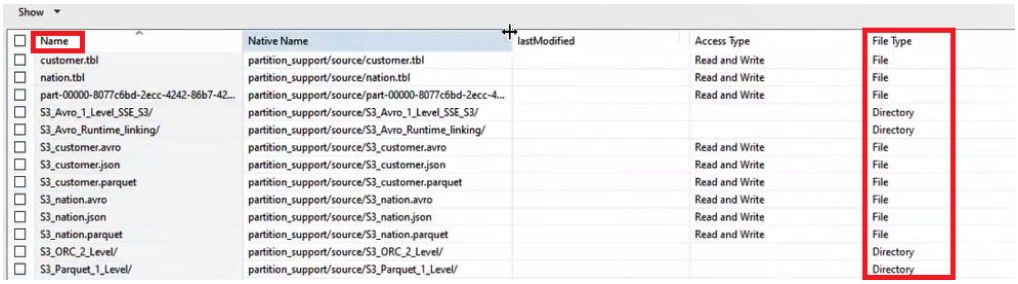
Directory-Level Partitioning

When you run mappings on the Spark and Databricks Spark engines, you can read data from and write data to Avro, ORC, and Parquet files that are partitioned based on directories.

Importing a data object with partition files

Perform the following steps to import a data object to read or write from partition files:

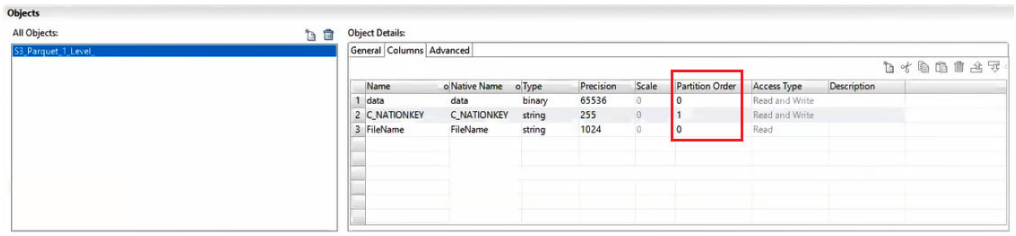
1. Select a project or folder in the **Object Explorer** view.
2. Click **File** > **New** > **Data Object**.
3. Select **AmazonS3 Data Object** and click **Next**.
   The **AmazonS3 Data Object** dialog box appears.
4. Click **Browse** next to the **Location** option and select the target project or folder.
5. In the **Resource Format** list, select Avro, Parquet, or ORC from the drop-down.
6. Click **Add** next to the **Selected Resource** option to add a resource to the data object. The **Add Resource** dialog box appears. You can use the **File Type** column to distinguish between a directory and a file. The following image shows the Add resource dialogue box where you can select the file name and directory:



7. Select the check box for a directory. Click **OK**.

8. Click **Finish**.
   The partitioned columns are displayed with the order of partitioning in the data object **Overview** tab.

   The following image shows the data object overview tab:

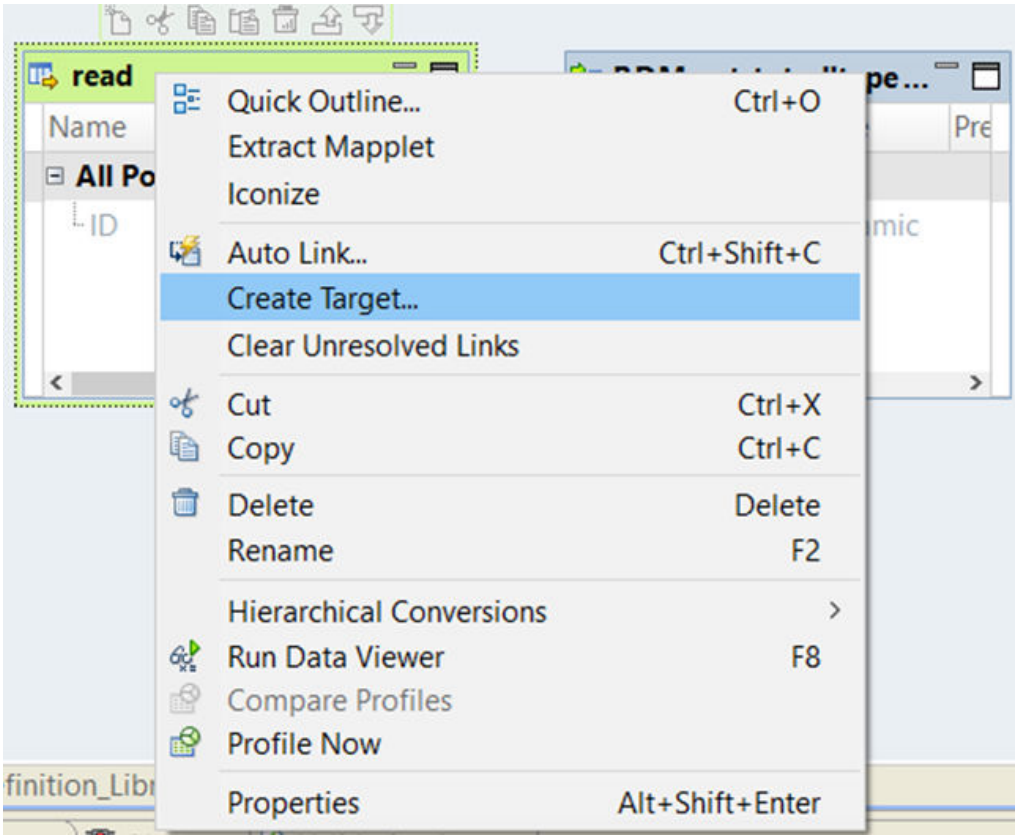


Create target with partition files

Perform the following steps to create target with partition files:

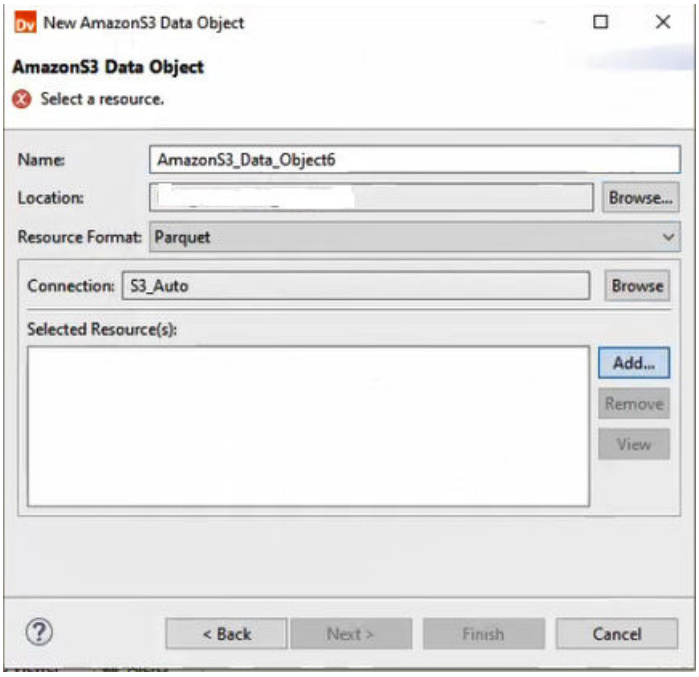
1. Select a project or folder in the **Object Explorer** view.
2. Select a source or a transformation in the mapping.
3. Right-click the Source transformation and select **Create Target**.
   The **Create Target** dialog box appears.

   The following image shows the **Create Target** option:

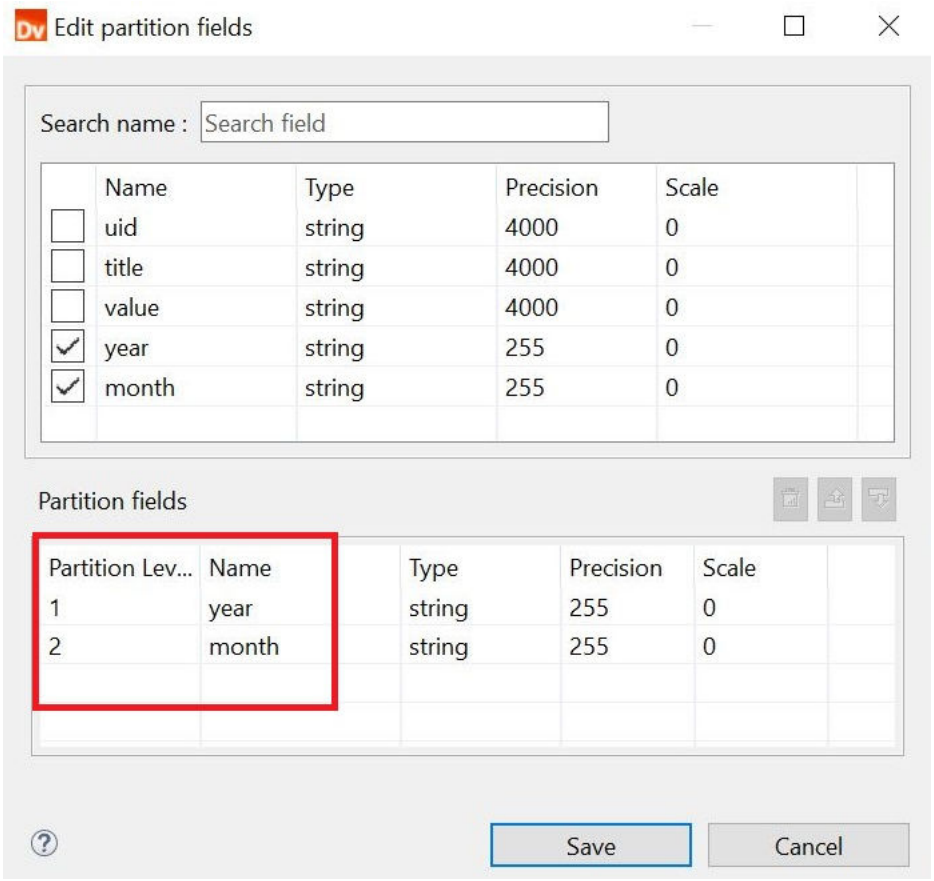


4. Select **Others** and then select **AmazonS3** data object from the list in the **Data Object Type** section.
5. Click **OK**.
   The **New AmazonS3 Data Object** dialog box appears.

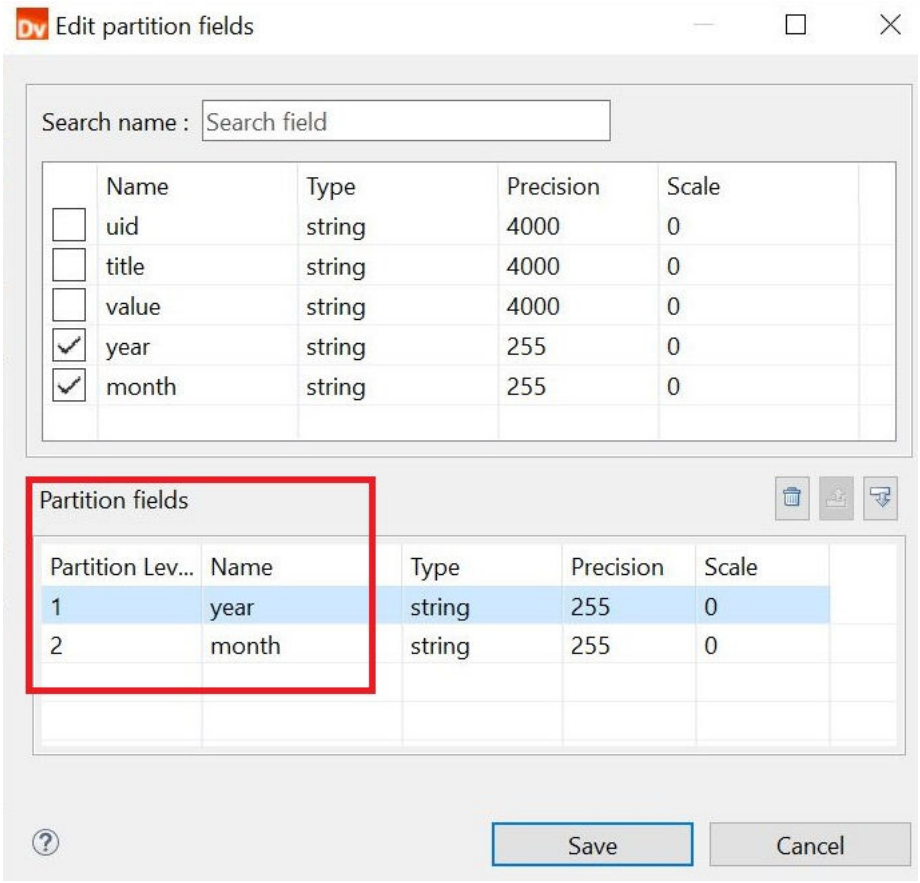
The following image shows the **New AmazonS3 Data Object** dialog box:



6. Enter a name for the data object.
7. Enter the partition fields.
   The following image shows the **Edit partition fields** dialog box:



8. You can change the partition order using the up and down arrows.
   The following image shows the partitioned fields after changing the order:



9. Click **Finish**.
   The partitioned columns are displayed with the order of partitioning in the data object **Overview** tab.

Rules and Guidelines for Directory-Level Partitioning

Consider the following rules and guidelines when you read from and write to a partition folder:

- You must select the file type as **Directory** to read from a partition folder.
- When you import a directory that has a partition folder, the data type for the partition column is imported as a String.
- You can read data from or write data to partition folders with Avro, Parquet, and ORC files.
- You cannot use the **Data Preview** option with partition columns.
- For **Create Target**, you can add partition fields and arrange the partition fields in an order.
- When you import a data object, the data and FileName port always show 0 as the partition order.
- You cannot use the FileName port when you use directory-level partitioning.
- The partitioned directory that you select cannot have a partitioned column named FileName. The name is case insensitive.

Audits

To validate the consistency and accuracy of data processed in a mapping for a read operation, you can create an audit for the mapping.

An audit is composed of rules and conditions. Use a rule to compute an aggregated value for a single column of data. Use a condition to make comparisons between multiple rules or between a rule and constant values.

You can run audits with mappings that run on the Data Integration Service or the Spark engine.

For more information, see the *Data Engineering Integration 10.5 User Guide*.

Amazon S3 Dynamic Mapping Overview

You can use Amazon S3 data objects as dynamic sources and targets in a mapping.

Use the Amazon S3 dynamic mapping to accommodate changes to source, target, and transformation logics at run time. You can use an Amazon S3 dynamic mapping to manage frequent schema or metadata changes or to reuse the mapping logic for data sources with different schemas. Configure rules, parameters, and general transformation properties to create the dynamic mapping.

If the data source for a source and target changes, you can configure a mapping to dynamically get metadata changes at runtime. If a source changes, you can configure the Read transformation to accommodate changes. If a target changes, you can configure the Write transformation accommodate target changes.

You do not need to manually synchronize the data object and update each transformation before you run the mapping again. The Data Integration Service dynamically determine transformation ports, transformation logic in the ports, and the port links within the mapping.

There are the two options available to enable a mapping to run dynamically. You can select one of the following options to enable the dynamic mapping:

- In the **Data Object** tab of the data object read or write operation, select the **At runtime, get data object columns from data source** option when you create a mapping.
  When you enable the dynamic mapping using this option, you can refresh the source and target schemas at the runtime.
- In the **Ports** tab of the data object write operation, select the value of the **Columns defined by** property as **Mapping Flow** when you configure the data object write operation properties.
  When you enable the dynamic mapping using this option, you can add all the Source transformation or transformation ports to the target dynamically and the Data Integration Service creates a target file with the ports at runtime.

Note: Dynamic mapping is applicable when you run the mapping in the native environment or on the Spark and Databricks Spark engine.

For information about dynamic mappings, see the *Informatica Developer Mapping Guide*.

Refresh Schema

You can refresh the source or target schema at the runtime when you enable a mapping to run dynamically. You can refresh the imported metadata before you run the dynamic mapping.

You can enable a mapping to run dynamically using the **At runtime, get data object columns from data source** option in the **Data Object** tab of the Read and Write transformations when you create a mapping.

When you add or override the metadata dynamically, you can include all the existing source and target objects in a single mapping and run the mapping. You do not have to change the source schema to update the data objects and mappings manually to incorporate all the new changes in the mapping.

You can use the mapping template rules to tune the behavior of the execution of such pipeline mapping.

When the Source or Target transformation contains updated ports such as changes in the port names, data types, precision, or scale, the Data Integration Service fetches the updated ports and runs the mapping dynamically. You must ensure that at least one of the column name in the source or target file is the same as before refreshing the schema to run the dynamic mapping successfully.

Even though the original order of the source or target ports in the table changes, the Data Integration Service displays the original order of the ports in the table when you refresh the schemas at runtime.

If there are more columns in the source file as compared to the target file, the Data Integration Service does not map the extra column to the target file and loads null data for all the unmapped columns in the target file.

If the Source transformation contains updated columns that do not match the Target transformation, the Data Integration Service does not link the new ports by default when you refresh the source or target schema. You must create a run-time link between the transformations to link ports at run time based on a parameter or link policy in the **Run-time Linking** tab. For information about run-time linking, see the *Informatica Developer Mapping Guide*.

Even though you delete the FileName port from the Source or Target transformation, the Data Integration Service adds the FileName port when you refresh the source or target schema.

Note: When you refresh a schema of a flat file, the Data Integration Service writes all data types as String data types.

Mapping Flow

You can add all the Source transformation or transformation ports to the target dynamically when enable a mapping to run dynamically using the **Mapping Flow** option. You can then use the dynamic ports in the Write transformation.

When you select the **Mapping Flow** option, the Data Integration Service allows the Target transformation to override ports of the Write transformation with all the updated incoming ports from the pipeline mapping and loads the target file with the ports at runtime.

To enable a dynamic mapping using the **Mapping Flow** option, select the value of the **Columns defined by** property as **Mapping Flow** in the **Ports** tab in the Write transformation.

When you use the **Mapping Flow** option to read data from a flat file that contains a port of Integer or Double data type, the mapping runs successfully. However, the Data Integration Service does not write the data of the port with Integer or Double data type and the consecutive ports regardless of the data type.

Note: When you run a dynamic mapping on the Spark or Databricks Spark engine using the **Mapping Flow** option to fetch the metadata changes from any source that contains a FileName port, the mapping fails. You must add a transformation and configure the **Input Rules** in the **Ports** tab of the transformation to exclude the FileName port from the Write transformation. Then, map the rest of the ports.

Amazon S3 Dynamic Mapping Example

Your organization has a large amount of data that keeps changing. Your organization needs to incorporate all the updated data in a short span of time. Create a dynamic mapping, where you can refresh the source

schema dynamically to fetch the updated data. Add all the dynamic ports to the target to override the metadata of the existing ports.

1. Import the Amazon S3 read and write data objects.
2. Select a project or folder in the **Object Explorer** view.
3. Click **File** > **New** > **Mapping.**

   The **Mapping** dialog box appears.
4. Enter the name of the mapping in the **Name** field.
5. Click **Finish**.
6. Drag the data object into a mapping.

   The **AmazonS3 Data Object Access** dialog box appears.
7. Select the **Read** option and click **OK**.
8. In the **Data Object** tab, select the **At runtime, get data object columns from data source** check box.
9. Drag the data object into a mapping.

   The **AmazonS3 Data Object Acess** dialog box appears.
10. Select the **Write** option and click **OK**.
11. In the **Ports** tab, select the value of the **Columns defined by** as **Mapping Flow**.
12. Select all the source ports and add the ports to the target.
13. Save and run the mapping.

CHAPTER 6

PowerExchange for Amazon S3 Lookups

This chapter includes the following topics:

PowerExchange for Amazon S3 Lookup Overview

You can use an Amazon S3 data object read operation to look up data in an Amazon S3 table.

You can add an Amazon S3 data object read operation as a lookup in a mapping. You can then configure a lookup condition to look up data from the Amazon S3 table.

When you preview a Lookup transformation based on an Amazon S3 logical data object, the performance might be slow.

You can configure a cached lookup operation to cache the lookup data in a mapping that runs on the Spark engine.

When you enable lookup caching, the Data Integration Service caches the lookup values. The Data Integration Service queries the lookup source once, caches the values, and looks up values in the cache. Caching the lookup values can increase performance on large lookup tables.

When you disable caching, the Data Integration Service does not cache the lookup values. The Data Integration Service queries the lookup source instead of building and querying the lookup cache. Each time a row passes, the Data Integration Service issues a SELECT statement to the lookup source for lookup values.

You can set cached lookup in the run-time properties of the lookup operation in a mapping.

For more information about the cached lookup, see "Lookup Transformation" in the *Developer Transformation Guide*.

General Properties

The general properties display the name and description of the Amazon S3 V2 lookup.

The following table describes the general properties that you can view and edit for an Amazon S3 V2 lookup:

Property	Description
Name	Name of the Amazon S3 V2 lookup.
Description	Description of the Amazon S3 V2 lookup.
Physical Data Object	Name of the Amazon S3 V2 data object read operation.
On multiple matches	Determines which row the Amazon S3 V2 lookup returns when it finds multiple rows that match the lookup condition. You can select one of the following options: - Return first row - Return last row - Return any row - Return all rows - Report error

Ports Properties

The ports properties display the input ports from the source in the mapping to the Amazon S3 V2 lookup. You can specify the ports to be available as output ports from the Amazon S3 V2 lookup. The ports properties display the data types, precision, and scale of the source port.

The following table describes the ports properties:

Property	Description
Name	Name of the source port.
Type	Data type of the source port.
Precision	Maximum number of significant digits for numeric data types, or maximum number of characters for string data types. For numeric data types, precision includes scale.
Scale	Maximum number of digits after the decimal point of numeric values.
Output	Specify the ports that must be available as output ports from the Amazon S3 V2 lookup.
Description	Description of the port.
Input Rules	A set of rules that filter the ports to include or exclude in the transformation based on port names or data type. Configure input rules when you define dynamic ports.

Run-time Properties

Set the run-time properties to configure a cached lookup in a mapping.

The following table describes the run-time properties for an Amazon S3 V2 data object lookup operation in the Run-time view:

Property	Description
Lookup caching enabled	Indicates whether the Data Integration Service caches lookup values. By default, the **Lookup caching enabled** check box is selected. When you disable caching, each time a row passes into the transformation, the Integration Service issues a select statement to the lookup source for lookup values.

Lookup Properties

Specify the lookup properties to look up an Amazon S3 V2 table. You can configure a lookup condition to look up data from the Amazon S3 V2 table.

There are two types of option that you must select in the **Specify by** property to configure a lookup condition:

- **Value**: Select this option if you want to configure a lookup condition using the column name.
- **Parameter**: Select this option if you want to parameterize the lookup condition.

The following table describes the lookup properties that you can specify for an Amazon S3 V2 lookup if you select the **Value** option:

Property	Description
Lookup Column	The name of the columns that you want to look up.
Operator	Operators that you can use to filter records. You can select one of the following operators: `=, !=, <=, >=, and`
Input Port	The input source port.

The following table describes the lookup properties that you can specify for an Amazon S3 V2 lookup if you select the **Parameter** option:

Property	Description
Parameter	The name of the parameter that you want to use to look up. You can also create a new parameter. Click **New** to create a new parameter. Enter the parameter name and specify an expression in the **New Parameter** dialog box. Click **Validate** to check if the expression that you specified is valid or not.

Adding an Amazon S3 V2 Data Object Operation as a Lookup in a Mapping

Use an Amazon S3 V2 lookup to look up data in an Amazon S3 V2 data object.

1. Open a mapping from the **Object Explorer** view.
2. From the **Object Explorer** view, drag an Amazon S3 V2 data object read operation to the editor.

   The **Add to Mapping** dialog box appears.
3. Select **Lookup** to add the data object read operation as a lookup in the mapping.
4. Click inside the Amazon S3 V2 object operation and connect the lookup input ports and the lookup output ports.
5. In the **Properties** view, configure the following parameters:
   a. On the **General** tab, select the option that you want the Data Integration Service to return when it finds multiple rows that match the lookup condition.
   b. On the **Ports** tab, configure the output ports and input rules.
   c. On the **Run-time** tab, select **Lookup caching enabled**.
   d. On the **Lookup** tab, enter the lookup condition properties.
6. When the mapping is valid, click **File** > **Save** to save the mapping to the Model repository.

APPENDIX A

Amazon S3 Data Type Reference

This appendix includes the following topics:

Data Type Reference Overview

The Developer tool uses the following data types in PowerExchange for Amazon S3 mappings.

Amazon S3 native data types

Native data types are specific to the sources and targets used as a physical data object. Native 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 source data, 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.

Amazon S3 and Transformation Data Types

You can use the data types that PowerExchange for Amazon S3 supports in an Amazon S3 mappings.

Use the supported data types in the following Amazon S3 files in a mapping:

- Flat
- Avro
- JavaScript Object Notation (JSON)
- ORC
- Parquet

Note: You can use Amazon S3 Connector to read or write Avro, JSON, ORC, and Parquet files only on Linux 64-bit operating system.

Flat File and Transformation Data Types

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

The following table compares flat file data types to transformation data types:

Flat File Data type	Transformation Data type	Range
Bigint	Bigint	Precision of 19 digits, scale of 0
Number	Decimal	For transformations that support precision up to 38 digits, the precision is 1 to 38 digits, and the scale is 0 to 38. For transformations that support precision up to 28 digits, the precision is 1 to 28 digits, and the scale is 0 to 28. If you specify the precision greater than the maximum number of digits, the Data Integration Service converts decimal values to double in high precision mode. If the precision is greater than 15, the Data Integration Service converts decimal values to double in low-precision mode.
String	String	1 to 104,857,600 characters
Nstring	String	1 to 104,857,600 characters

Avro Data Types and Transformation Data Types

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

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

Avro Data Type	Transformation Data Type	Range
Array	Array	Unlimited number of characters.
Boolean	Integer	TRUE (1) or FALSE (0).
Bytes	Binary	Precision 4000.
Date	Date/Time	January 1, 0001 to December 31, 9999.
Decimal	Decimal	Decimal value with declared precision and scale. Scale must be less than or equal to precision. For transformations that support precision up to 38 digits, the precision is 1 to 38 digits, and the scale is 0 to 38. For transformations that support precision up to 28 digits, the precision is 1 to 28 digits, and the scale is 0 to 28. If you specify the precision greater than the maximum number of digits, the Data Integration Service converts decimal values to double in high precision mode.

Avro Data Type	Transformation Data Type	Range
Double	Double	Precision 15.
Fixed	Binary	1 to 104,857,600 bytes.
Float	Double	Precision 15.
Int	Integer	-2,147,483,648 to 2,147,483,647 Precision 10 and scale 0.
Long	Bigint	-9,223,372,036,854,775,808 to 9,223,372,036,854,775,807. Precision 19 and scale 0.
Map	Map	Unlimited number of characters.
Record	Struct	Unlimited number of characters.
String	String	1 to 104,857,600 characters.
Time	Date/Time	Time of the day. Precision to microsecond.
Timestamp	Date/Time	January 1, 0001 00:00:00 to December 31, 9999 23:59:59.997. Precision to microsecond.
Union	Corresponding data type in a union of ["primitive_type|complex_type", "null"] or ["null", "primitive_type|complex_type"].	Dependent on primitive or complex data type.

Avro Union Data Type

A union indicates that a field might have more than one data type. For example, a union might indicate that a field can be a string or a null. A union is represented as a JSON array containing the data types.
The Developer tool only interprets a union of ["primitive_type|complex_type", "null"] or ["null", "primitive_type|complex_type"]. The Avro data type converts to the corresponding transformation data type.

Avro Timestamp Data Type Support

The following table lists the Timestamp data type support for Avro file formats:

Timestamp Data type	Native	Spark
Timestamp_micros	Yes	Yes
Timestamp_millis	Yes	No
Time_millis	Yes	No
Time_micros	Yes	No

Unsupported Avro Data Types

The Developer tool does not support the following Avro data types:

- Enum
- Null
- Timestamp_tz

JSON Data Types and Transformation Data Types

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

The following table compares the JSON data types that the Data Integration Service supports and the corresponding transformation data types:

JSON	Transformation	Range
Array	Array	Unlimited number of characters.
Double	Double	Precision of 15 digits.
Integer	Integer	-2,147,483,648 to 2,147,483,647. Precision of 10, scale of 0.
Object	Struct	Unlimited number of characters.
String	String	1 to 104,857,600 characters.

Unsupported JSON Data Types

The Developer tool does not support the following JSON data types:

- Date
- Decimal
- Timestamp
- Enum
- Union

ORC Data Types and Transformation Data Types

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

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

ORC File Data Type	Transformation Data Type	Range and Description
BigInt	BigInt	-9223372036854775808 to 9,223,372,036,854,775,807.
Boolean	Integer	TRUE (1) or FALSE (0).
Char	String	1 to 104,857,600 characters.
Date	Date/Time	January 1, 0001 to December 31, 9999.
Double	Double	Precision of 15 digits.
Float	Double	Precision of 15 digits.
Integer	Integer	-2,147,483,648 to 2,147,483,647.
SmallInt	Integer	-32,768 to 32,767.
String	String	1 to 104,857,600 characters.
Timestamp	Date/Time	January 1, 0001 00:00:00 to December 31, 9999 23:59:59.997. Precision to microsecond.
TinyInt	Integer	-128 to 127.
Varchar	String	1 to 104,857,600 characters.

When you run a mapping on the Spark or Databricks Spark engine to write an ORC file to a target, the Data Integration Service writes the data of the Char and Varchar data types as String.

Note: You can use ORC data types to read and write complex file objects in mappings that run on the Spark engine only.

Unsupported ORC Data Types

The Developer tool does not support the following ORC data types:

- Map
- List
- Struct
- Union

Parquet Data Types and Transformation Data Types

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

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

Parquet	Transformation	Range
Binary	Binary	1 to 104,857,600 bytes
Binary (UTF8)	String	1 to 104,857,600 characters
Boolean	Integer	-2,147,483,648 to 2,147,483,647 Precision of 10, scale of 0
Date	Date/Time	January 1, 0001 to December 31, 9999.
Decimal	Decimal	Decimal value with declared precision and scale. Scale must be less than or equal to precision. For transformations that support precision up to 38 digits, the precision is 1 to 38 digits, and the scale is 0 to 38. For transformations that support precision up to 28 digits, the precision is 1 to 28 digits, and the scale is 0 to 28. If you specify the precision greater than the maximum number of digits, the Data Integration Service converts decimal values to double in high precision mode.
Double	Double	Precision of 15 digits.
Float	Double	Precision of 15 digits.
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
Map	Map	Unlimited number of characters.
Struct	Struct	Unlimited number of characters.
Time	Date/Time	Time of the day. Precision to microsecond.
Timestamp	Date/Time	January 1, 0001 00:00:00 to December 31, 9999 23:59:59.997. Precision to microsecond.
group (LIST)	Array	Unlimited number of characters.

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.

Parquet Timestamp Data Type Support

The following table lists the Timestamp data type support for Parquet file formats:

Timestamp Data type	Native	Spark
Timestamp_micros	Yes	No
Timestamp_millis	Yes	No
Time_millis	Yes	No
Time_micros	Yes	No
int96	Yes	Yes

Unsupported Parquet Data Types

The Developer tool does not support the following Parquet data types:

- Timestamp_nanos
- Time_nanos
- Timestamp_tz

Rules and Guidelines for Data Types

Consider the following rules and guidelines for data types:

- Avro data types support:
  - Date, Decimal, and Timestamp data types are applicable when you run a mapping in the native environment or on the Spark engine in Cloudera CDH 6.3 distribution.
  - Time data type is applicable when you run a mapping in the native environment in Cloudera CDH 6.3 distribution.
- JSON data types support:
  - For PowerExchange for Microsoft Azure Data Lake Storage Gen2, you can read and write complex file objects in JSON format in mappings that run in the native environment, Spark engine, and Databricks Spark engine.
    For other file-based adapters, you can read and write complex file objects in JSON format in mappings that run on the Spark engine only.
- Parquet data types support:
  - Before you create and run a new mapping on the Databricks engine to read a parquet file with hierarchical data types, you must set the `-DINFA_HADOOP_DIST_DIR=hadoop\Databricks_7.2` option in the `developerCore.ini` file.
  - When you set the `-DINFA_HADOOP_DIST_DIR=hadoop\<Distro>` option in the `developerCore.ini` file and import a Parquet file, the format of the imported metadata differs based on the distribution. For Cloudera CDP 7.1, the metadata is imported as string and for other supported distributions, the metadata is imported as UTF8.

- Date, Time, and Timestamp data types till microseconds are applicable when you run a mapping in the native environment , Blaze, and Spark engine in the Hortonworks HDP 3.1, Azure HDInsight HDI 4.0, and Cloudera CDP 7.1 distributions.
- Date, Time_Millis, and Timestamp_Millis data types are applicable when you run a mapping in the native environment or Spark engine in MapR 6.1.
- Decimal data types are applicable when you run a mapping in the native environment and Spark engine in Cloudera CDH 6.3, Hortonworks HDP 3.1, Amazon EMR 5.20, MapR 6.1, and Azure HDInsight HDI 4.0 distributions.
- Date, Time, Timestamp, and Decimal data types are applicable when you run a mapping on the Databricks Spark engine.
- When you run a mapping and use Date data type that does not have a time value, the Data Integration Service adds the time value, based on the time zone, to the date in the target.
  For example, Date data type used in the source:

  `1980-01-09`

  Value generated in the target:

  `1980-01-09 00:00:00`
- When you run a mapping in the native environment and use Time data type in the source, the Data Integration Service writes incorrect date value to the target.
  For example, Time data type used in the source:

  `1980-01-09 06:56:01.365235000`

  Incorrect Date value is generated in the target:

  `1899-12-31 06:56:01.365235000`
- When you run a mapping in the native environment and use Date data type in the source, the Data Integration Service writes incorrect time value to the target.
  For example, Date data type used in the source:

  `1980-01-09 00:00:00`

  Incorrect Time value generated in the target:

  `1980-01-09 05:30:00`

- To run a mapping that reads and writes Date, Time, Timestamp, and Decimal data types, update the `-DINFA_HADOOP_DIST_DIR` option to the `developerCore.ini` file. The `developerCore.ini` file is located in the following directory:
  `<Client installation directory>\clients\DeveloperClient\`

  Add the following path to the `developerCore.ini` file:

  `-DINFA_HADOOP_DIST_DIR=hadoop\<Hadoop distribution>_<version>`

  For example: `-DINFA_HADOOP_DIST_DIR=hadoop\CDH_6.3`

- To use precision up to 38 digits for Decimal data type in the native environment, set the `EnableSDKDecimal38` custom property to `true` for the Data Integration Service. The `EnableSDKDecimal38` custom property is applicable to all file-based PowerExchange adapters except PowerExchange for HDFS.

APPENDIX B

Troubleshooting

This appendix includes the following topics:

Troubleshooting Overview

Use the following sections to troubleshoot errors in PowerExchange for Amazon S3.

Troubleshooting for PowerExchange for Amazon S3

How to enable Metadata Access Service for PowerExchange for Amazon S3?

You can optionally enable Metadata Access Service to import metadata from Amazon S3. For information on how to enable Metadata Access Service, see https://knowledge.informatica.com/s/article/HOW-TO-Enable-Metadata-Access-Service-to-import-metadata-from-Amazon-S3-and-Amazon-Redshift?language=en_US

How to solve the following error that occurs while running an Amazon S3 mapping on the Spark engine to write a Parquet file and then run another Amazon S3 mapping or preview data in the native environment to read that Parquet file: "The requested schema is not compatible with the file schema."

For information about the issue, see https://kb.informatica.com/solution/23/Pages/58/497835.aspx?myk=497835

What are the performance tuning guidelines to read data from or write data to Amazon S3?

For information about performance tuning guidelines, see https://docs.informatica.com/data-integration/powerexchange-adapters-for-informatica/h2l/0990-performance-tuning-guidelines-to-read-data-from-or-write-da/abstract.html

How to solve the out of disk space error that occurs when you use PoweExchange for Amazon S3 to read and preview data?

For information about the issue, see https://kb.informatica.com/solution/23/Pages/62/516321.aspx?myk=516321

How to solve the following error that occurs when you enable server-side encryption with KMS and run an Amazon S3 mapping on the Spark engine with EMR 5.16 distribution: " [java.lang.RuntimeException: java.lang.ClassNotFoundException: Class com.amazon.ws.emr.hadoop.fs.EmrFileSystem not found]"

For information about the issue, see https://kb.informatica.com/solution/23/Pages/69/570023.aspx?myk=570023

How to solve the following error that occurs when you run an Amazon S3 mapping on the Spark engine and then run another mapping in the native environment with Server-side Encryption with KMS enabled: "The encryption method specified is not supported"

For information about the issue, see https://kb.informatica.com/solution/23/Pages/69/571479.aspx?myk=571479

Mapping on the Spark engine fails with an error when you use the Amazon S3 bucket without enabling the KMS policy and use server-side encryption with KMS enabled on EMR 5.20 or later distributions.

If you run a mapping on the Spark engine with such configurations, the mapping fails with the following error message:

```
com.amazon.ws.emr.hadoop.fs.shaded.com.amazonaws.services.s3.model.AmazonS3Exception:
Invalid arn us-west-1 (Service: Amazon S3; Status Code: 400; ErrorCode:
KMS.NotFoundException;
```

If you use server-side encryption with KMS enabled on EMR 5.20 or later distributions, you must enable the KMS policy for the Amazon S3 bucket to run the mapping successfully.

Mapping on the Spark engine fails with an error when you use the Amazon S3 bucket with a dot (.) in the bucket name on CDP 7.1 distribution.

If you run a mapping on the Spark engine with such configurations, the mapping fails with the following error message:

```
Unable to execute HTTP request: Certificate for xxxx doesn't match any of the
subject alternative names
```

Perform the following steps to run the mapping successfully:

1. In the CDP cluster, go to **HDFS**.
2. Click **Configuration**.
3. In **Cluster-wide Advanced Configuration Snippet (Safety Valve) for core-site.xml**, add the property **fs.s3a.path.style.access** and set the value to **true**.
4. Restart the cluster.
5. Refresh the cluster configuration object.
6. Restart the Data Integration Service.

Decimal point is shifted when you write decimal data to an Avro file.

If the precision and scale in the source file are different from the precision and scale in the schema, the decimal point is shifted when you write the source data to an Avro target.

The issue occurs on distributions that use Avro 1.8.x and 1.9.x versions.

Use Avro 1.10.0 version to fix the issue. For more information, see https://issues.apache.org/jira/browse/AVRO-2837.

INDEX

P

R

S

T

W