

Creating Time Stamp Parameterized Target Files using Amazon S3 Connector

Abstract

You can use Amazon S3 Connector to read delimited file data from and write delimited file data to Amazon S3. Amazon S3 Connector enables you to specify the name and path of the target file at run time. This document explains the steps to create a target file with the time stamp at run time.

Supported Versions

- Informatica Cloud Amazon S3 Connector Spring 2017

Table of Contents

Overview.	2
Creating a Salesforce Connection.	3
Creating an Amazon S3 Connection.	4
Creating a Mapping.	4
Creating a Mapping Configuration Task.	5
Supported STRFTIME Formats.	6

Overview

You are a data analyst in a cloud services-based organization who maintains their customers` transactions records in Salesforce. The amount of transactions happening per day and data being stored in Salesforce is humongous. For an internal analysis, you need to port data from Salesforce to Amazon S3. As the amount of data stored in Salesforce is huge, you would like to read the data on a daily basis and write it to an Amazon S3 target file. You want the target file to contain a time stamp and be created automatically at a specified folder location when the task runs.

For Amazon S3 files, the target transformation provides you an option to parameterize the target file name and the target file name can include Apache STRFTIME function common formats.

Creating a Salesforce Connection

Create a Salesforce connection to read data from the Salesforce source. The following image shows the sample Salesforce connection properties:

New Connection

Connection Details

Connection Name:*

Description:

Type:* ▼

Salesforce Connection Properties

Runtime Environment:* ▼

User Name:*

Password:*

Security Token:

Service URL:*

Bypass proxy server settings defined for the Secure Agent

Creating an Amazon S3 Connection

Create an Amazon S3 connection to write data to an Amazon S3 target. The following image shows the sample Amazon S3 connection properties:

New Connection

OK Cancel Test

Connection Details

Connection Name:* AmazonS3connection

Description: To write data to Amazon S3

Type:* ? Amazon S3 (Informatica Cloud) ▼

Amazon S3 Connection Properties

Runtime Environment:* ? INWAmazonS3 ▼

Access Key:

Secret Key:

Folder Path:* sample.bucket/customerTransaction

Master Symmetric Key:

Customer Master Key ID:

Code Page: UTF-8 ▼

Region Name: US East(N. Virginia) ▼

Ensure that you do not use a forward slash at the end of the folder path in the connection properties. The Amazon S3 bucket must be available in the region you selected.

Creating a Mapping

Design a mapping to read the Salesforce source and create the Amazon S3 target files at run time.

Perform the following steps:

1. Click **Design > Mappings** and then click **New Mapping**.
2. In the **New Mapping** dialog box, enter the mapping name and description, and click **OK**.
3. On the Transformation palette, click **Source** to configure the Salesforce source.
4. Select the Salesforce connection, source type, and object to read the data from.
5. In **Query Options**, add a filter to read records updated on a daily basis. For example, you can set the `LastModifiedDate` or the `SystemModstamp` greater than the `$(LastRunDate)`.
6. Click **OK**.
7. On the Transformation palette, click **Target** to configure the Amazon S3 target.
8. On the Incoming Fields tab, configure field rules to specify the fields to include in the target.
9. To specify the target, click the **Target** tab.
10. Select the Amazon S3 connection.
11. For the target type, choose **Parameter**.

12. Select an existing parameter or create a new parameter.
13. On the **Field Mapping** tab, select Automatic in **Field map options**.
14. Save the mapping.

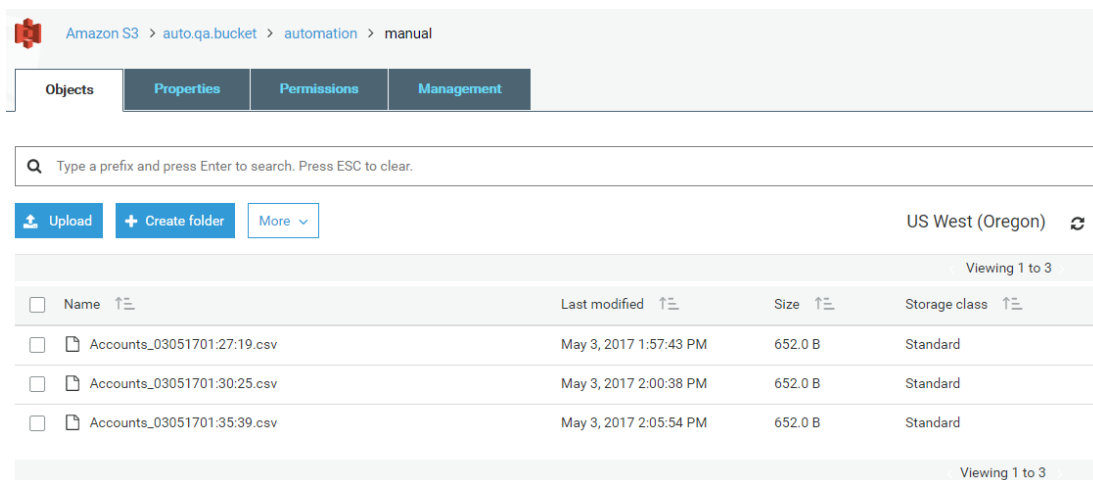
Creating a Mapping Configuration Task

Create a new Mapping Configuration task by selecting the mapping you created in the previous section. Perform the following steps to create a Mapping Configuration task:

1. Click **Tasks** > **Mapping Configuration** and then click **New** in the **Mapping Configurations Tasks** page.
2. Specify the Name, Description, and Runtime Environment.
3. Select Mapping in the **Task Based On** field.
4. Select the mapping you created in the **Mapping** field and click **Next**.
5. For **NewParameter Object**, click **Create Target**.
6. To include a time stamp in the target file name, click **Handle Special Characters** and add special characters to the file name in the **Object** field. For example, add the special characters shown here to include all the time stamp information: `Accounts_%d%m%y%T.csv`. You must specify a `.csv` target file name.
7. Click **OK**.
8. Click **Formatting Options** if you want to configure the formatting options for the file, and click **OK**.
9. Click **Next** and specify **Advanced properties** for the target, if needed.
10. Save and run the Mapping Configuration task.

Each time the task is run, a new Amazon S3 target file is created by appending the current date and time format you specified in the mapping. The target file is created at the folder path you specified in the Amazon S3 connection properties.

The following image shows the sample output files:



Supported STRFTIME Formats

When you specify the file name for the target file, you include special characters based on Apache STRFTIME function formats. The Mapping Configuration task uses these formats to include time stamp information in the file name.

The following table describes some common STRFTIME function formats that you might use:

Special Character	Description
%d	Day as a two-decimal number, with a range of 01-31.
%m	Month as a two-decimal number, with a range of 01-12.
%y	Year as a two-decimal number without the century, with range of 00-99.
%Y	Year including the century, for example 2015.
%T	Time in 24-hour notation, equivalent to %H:%M:%S.
%H	Hour in 24-hour clock notation, with a range of 00-24.
%I	Hour in 12-hour clock notation, with a range of 01-12.
%M	Minute as a decimal, with a range of 00-59.
%S	Second as a decimal, with a range of 00-60.
%p	Either AM or PM.

Authors

Madhuri Veluri
Lead QA Engineer

Chanchal Das
Lead Technical Writer