

Connect to Amazon S3 using Teradata Parallel Transporter Access Module

As many Teradata customers have interest in integrating Teradata Vantage with AWS First Party Services, this Getting Started Guide document will help you to integrate Amazon S3 with Teradata Vantage using the Teradata Parallel Transporter (TPT) access module.

The approach this guide explains is one of many potential approaches to integrate with the service and is offered on an as-is basis. Although the approach has been implemented and tested internally, there is no formal support from either Teradata or AWS on the approach.

That said, your feedback is very desired and appreciated – what worked, what didn’t, how can this be improved, or etc. Please send your feedback to [wenjie.tehan@teradata.com](mailto:wenjie.tehan@teradata.com).

# Overview

The [Teradata Access Module for Amazon S3](https://docs.teradata.com/reader/p~0sSD4zl4K8YPbEGnM3Rg/d~1yZcRSrPphEh64NatBSA) is a shared library that acts as an interface between the Teradata Parallel Transporter (TPT) load/unload operators and Amazon S3 (Simple Storage Service). The Teradata Access Module for Amazon S3 supports checkpoint and restart/recovery operations in scenario, where data is exported from Amazon S3 and loaded into the Teradata Vantage.

The Teradata Access Module for Amazon S3 runs on following operation systems:

|  |  |
| --- | --- |
| Operating System | Teradata Access Module for Amazon S3 |
| Linux | libs3axsmod.so |
| Windows | libs3axsmod.dll |

This Getting Started Guide will show how to connect and move data between Teradata Vantage and Amazon S3. We will use Teradata Access Module for Amazon S3 on Windows (libs3axsmod.dll) to perform the following:

* Export data from Teradata Vantage to Amazon S3
* Import data from Amazon S3 to Teradata Vantage

**Prerequisites:**

* [A S3 bucket](https://docs.aws.amazon.com/AmazonS3/latest/gsg/CreatingABucket.html) to store exported data
* Download [Teradata Tools and Utilities – Windows](https://downloads.teradata.com/download/tools/teradata-tools-and-utilities-windows-installation-package)
  + Install Teradata Parallel Transporter (Base and Stream)
  + Install Teradata Access Module for Amazon S3

### About Teradata Vantage

Vantage is the modern cloud platform that unifies data warehouses, data lakes, and analytics into a single connected ecosystem.

Vantage combines descriptive, predictive, prescriptive analytics, autonomous decision-making, ML functions, and visualization tools into a unified, integrated platform that uncovers real-time business intelligence at scale, no matter where the data resides.

Vantage enables companies to start small and elastically scale compute or storage, paying only for what they use, harnessing low-cost object stores and integrating their analytic workloads.

Vantage supports R, Python, Teradata Studio, and any other SQL-based tools. You can deploy Vantage across public clouds, on-premises, on optimized or commodity infrastructure, or as-a-service.

# Getting Started

Following configurations need to be done before start copying data.

## “HOME” Environment Variable

“HOME” environment is required for configuration files. On windows platform, “HOME” environment variable has to be manually created by going into “Control Panel” -> “System” -> “Advanced system settings” -> “Environment Variables”. Add a new User variable “HOME” and give it a path to where you want the configuration files to be stored (i.e. C:\Program Files\Teradata\Config).

Create “aws” folder under where “HOME” environment variable points (i.e. C:\Program Files\Teradata\Config\aws).

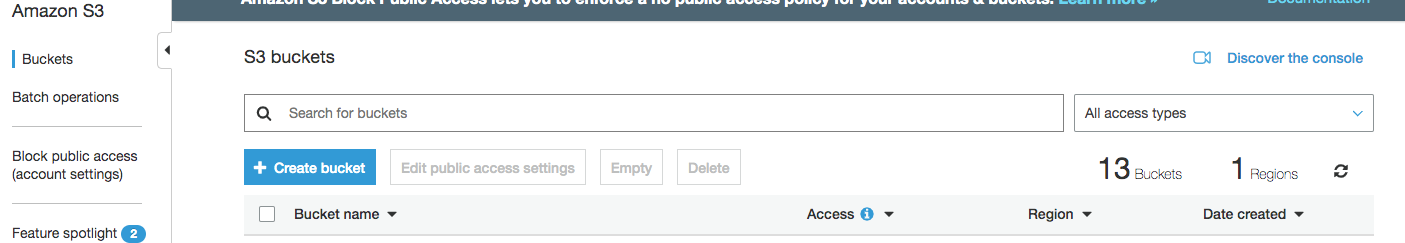
*For Linux operating system, each user has a predefined “HOME” directory. Create “.aws” directory (i.e. ~/.aws) under your “HOME” directory.*

## The Configuration Files

There are two configuration files – config and credentials.

### Config File

The “config” file contains the region name where the Amazon S3 bucket is located. “Region” for a particular bucket is listed in the S3 console along with the bucket name.



Create config file under <HOME>/aws directory for Windows (i.e. C:\Program Files\Teradata\Config\aws), or under ~/.aws/config directory for Linux.

The following is an example of the contents of the config file:

[default]

region = us-west-2

[export\_job\_7]

region = us-east

[load\_job\_5]

region = us-west-2

Default region is expected to be present.

### Credentials File

The credentials file contains the user credentials such as [“Access Key Id” and “Secret Access Key”](https://docs.aws.amazon.com/general/latest/gr/aws-sec-cred-types.html). The following is an example of the contents of the credentials file with a default location of <HOME>/aws/credentials (or ~/.aws/credentials for Linux).

[default]

aws\_access\_key\_id = AKIA\*\*\*\*\*\*\*\*\*\*\*\*T6XA

aws\_secret\_access\_key = Mpm1\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*CNT4

[export\_job\_7]

aws\_access\_key\_id = AKIA\*\*\*\*\*\*\*\*\*\*\*\*WLCX

aws\_secret\_access\_key = Mpm1\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*QQL7

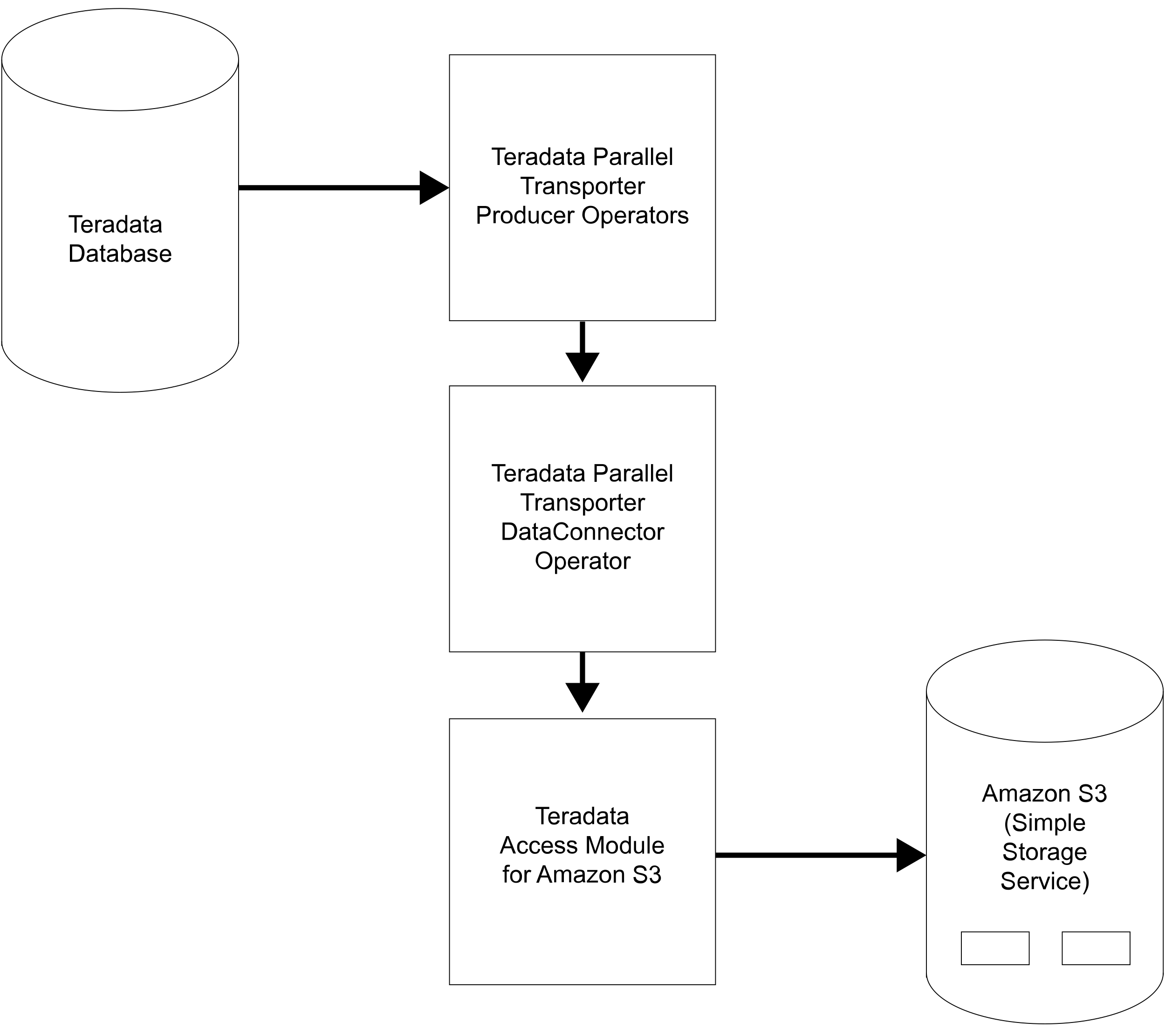
[load\_job\_5]

aws\_access\_key\_id = AKIA\*\*\*\*\*\*\*\*\*\*\*\*WWNB

aws\_secret\_access\_key = Mpm1\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*CNT4

Note: You can also store the config file and the credentials file in any directory of your choice. However, you need to specify the directory path where the config file and the credentials file are stored, through the initialization string parameter named S3ConfigDir if they are not located under your home directory.

# Export Data from Teradata to Amazon S3



Following scripts are needed to export data from Teradata to S3.

* Job Variable File (optional)
* Setup Job (optional)
* Export Job

Create all job files and place them in the same directory (i.e. C:\Program Files\Teradata\TPT Scripts).

## Job Variable File

This file defines the values for job variables. This file is optional. If not created, specifying variables in the job script using the “SET” directive before the “DEFINE JOB” statement. It’s recommended to create a separate job variable file to enhance the efficiency and security of job scripting.

Following is a sample of job variable file. You can add additional job variable names and values when required.

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\* TPT attributes - Common for all Samples \*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

TargetTdpId = '<Teradata Database ID>'

,TargetUserName = '<TargetUserName>'

,TargetUserPassword = '<TargetUserPassword>'

,TargetErrorList = [ '3706','3803','3807' ]

,DDLPrivateLogName = 'DDL\_OPERATOR\_LOG'

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\* TPT EXPORT Operator attributes \*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

,ExportPrivateLogName = 'EXPORT\_OPERATOR\_LOG'

,SourceTdpId = '<Teradata Database ID>'

,SourceUserName = '<SourceUserName>'

,SourceUserPassword = '<SourceUserPassword>'

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\* TPT DataConnector Consumer Operator \*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

,FileWriterFormat = 'Formatted'

,FileWriterPrivateLogName = 'FILE\_WRITER\_LOG'

,FileWriterFileName = 'S3W001DT'

,FileWriterDirectoryPath = '.'

,FileWriterOpenMode = 'Write'

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\* APPLY STATEMENT parameters \*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

,ExportInstances = 1

,FileWriterInstances = 1

Replace “<>” brackets with appropriate values and remove the brackets. Save the file in the job script directory once you are done (i.e. C:\Program Files\Teradata\TPT Scripts\jobVarExport).

## Setup File (Optional)

This file is used to set up and populate source table to export to Amazon S3. If you already have a table in place, this file is not needed. Sample setup job file can be found [here](https://docs.teradata.com/reader/p~0sSD4zl4K8YPbEGnM3Rg/vwhRVr1Rdp9JO4~yHJkGjw).

## Export Job

Assuming the config and credentials files are created and placed in the <HOME>/aws directory (~/.aws for Linux), following is a sample export job script.

DEFINE JOB EXPORT\_FROM\_TERADATA

DESCRIPTION 'Export data from Teradata to Amazon S3'

(

STEP EXPORT\_THE\_DATA

(

APPLY TO OPERATOR ( $FILE\_WRITER()

ATTR

(

AccessModuleName = 'libs3axsmod.dll',

AccessModuleInitStr = 'S3Bucket=<S3BucketName> S3Prefix="<PrefixName>/" S3Object=<output file name> S3SinglePartFile=True'

)

)

SELECT \* FROM OPERATOR ( $EXPORT

ATTR

(

SelectStmt = 'SELECT \* FROM <source table name>;'

)

);

);

);

The parameters inside “<>” brackets need to be replaced appropriately and “<>” brackets need to be removed. Save the file once you are done into the job script directory (i.e., C:\Program Files\Teradata\TPT Scripts\export).

*Note: if you are using Linux platform, use “libs3axsmod.so” as “AccessModuleName” instead.*

## Run Job

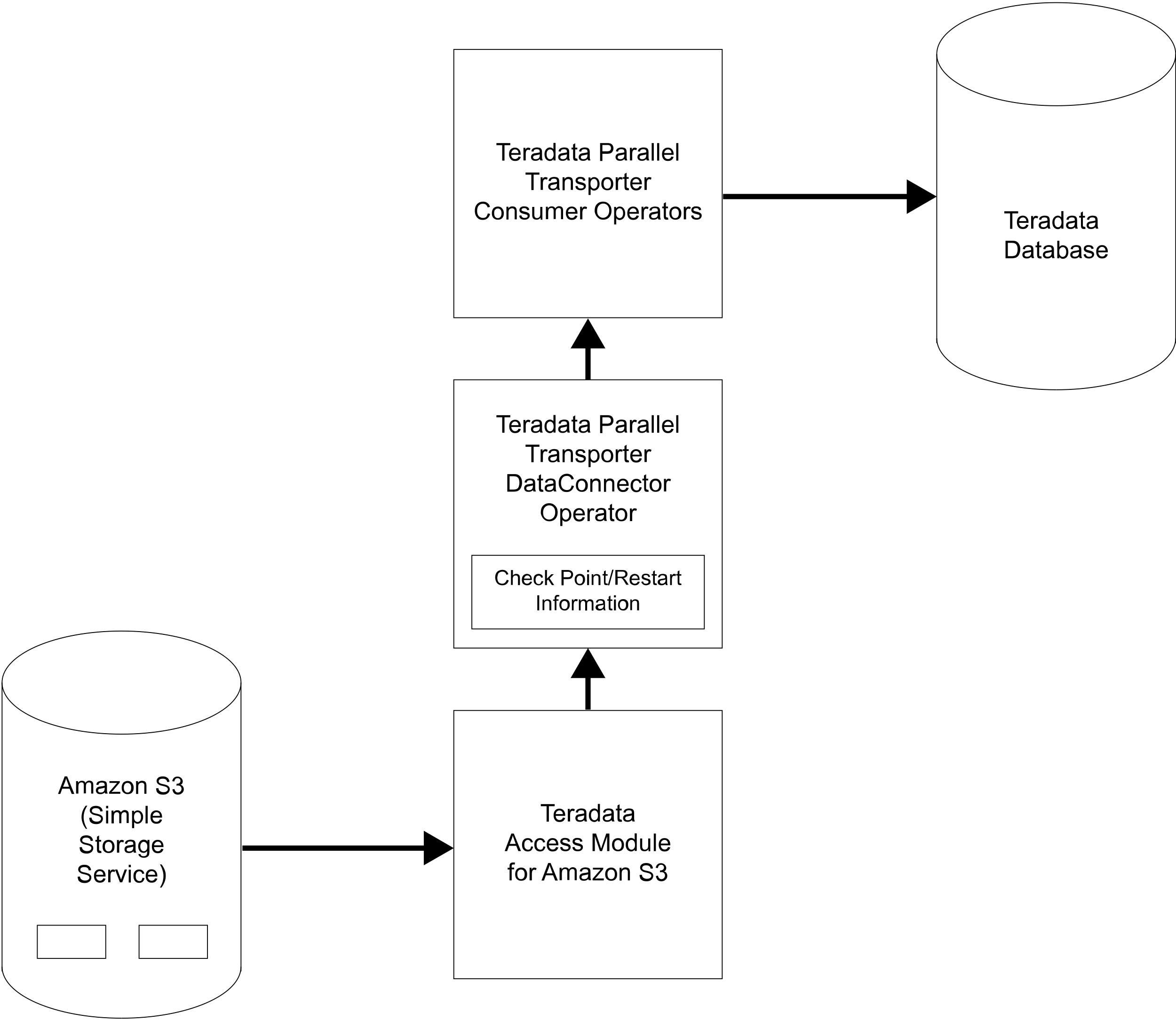
To run job script, open Windows Command Prompt (CMD) and change directory to scripts location and run:

* tbuild -f <job script> -v <job variable file> -j <job name>

i.e. > tbuild -f export -v jobVarExport -j td2s3

The output file will be placed in the Amazon S3 bucket you defined in the export job - <S3BucketName>/<PrefixName>/<output file name>.

# Import Data to Teradata from Amazon S3



These scripts are needed to import data from Teradata to Amazon S3.

* Job Variable File (optional)
* Setup Job (optional)
* Import Job

Create the script files and place them in the same directory (i.e. C:\Program Files\Teradata\TPT Scripts).

## Job Variable File

This file defines the values for job variables. This file is optional. If not created, specifying variables in the job script using the “SET” directive before the “DEFINE JOB” statement. It’s recommended to create a separate job variable file to enhance the efficiency and security of job scripting.

Following is a sample of job variable file. You can add additional job variable names and values when required.

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\* TPT attributes - Common for all Samples \*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

TargetTdpId = '<Teradata Database ID>'

,TargetUserName = '<TargetUserName>'

,TargetUserPassword = '<TargetUserPassword>'

,TargetErrorList = [ '3706','3803','3807' ]

,DDLPrivateLogName = 'DDL\_OPERATOR\_LOG'

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\* TPT LOAD Operator attributes \*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

,LoadPrivateLogName = 'LOAD\_OPERATOR\_LOG'

,LoadTargetTable = '<target table name>'

,LoadLogTable = '<target table name>\_log'

,LoadErrorTable1 = '<target table name>\_e1'

,LoadErrorTable2 = '<target table name>\_e2'

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\* TPT DataConnector Producer Operator \*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

,FileReaderFormat = 'Formatted'

,FileReaderPrivateLogName = 'S3W001P2\_1'

,FileReaderFileName = 'S3W001DT'

,FileReaderDirectoryPath = '.'

,FileReaderOpenMode = 'Read'

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\* APPLY STATEMENT parameters \*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

,LoadInstances = 1

,FileReaderInstances = 1

Replace “<>” brackets with appropriate values and remove the brackets. Save the file in the job script directory once you are done (i.e. C:\Program Files\Teradata\TPT Scripts\jobVarImport).

## Setup Job (Optional)

This file is used to clean up and create target table to import to Teradata. If you already have a target table in place, and no errors from previous runs, this file is not needed. Sample setup job file can be found [here](https://docs.teradata.com/reader/p~0sSD4zl4K8YPbEGnM3Rg/vwhRVr1Rdp9JO4~yHJkGjw).

## Import Job

Assuming the config and credentials files are created and placed in the <HOME>/aws directory (~/.aws directory for Linux), following is a sample import job script.

DEFINE JOB IMPORT\_TO\_TERADATA

DESCRIPTION 'Import data to Teradata from Amazon S3'

(

SET TargetTable = '<target table name>';

STEP IMPORT\_THE\_DATA

(

APPLY $INSERT TO OPERATOR ($LOAD)

SELECT \* FROM OPERATOR ($FILE\_READER ()

ATTR

(

AccessModuleName = 'libs3axsmod.dll',

AccessModuleInitStr = 'S3Bucket=<S3BucketName> S3Prefix="<PrefixName>/" S3Object=<input file name> S3SinglePartFile=True'

)

);

The parameters inside “<>” brackets need to be replaced appropriately and “<>” brackets need to be removed. Save the file once you are done into the job script directory (i.e. C:\Program Files\Teradata\TPT Scripts\import).

*Note: if you are using Linux platform, use “libs3axsmod.so” as “AccessModuleName” instead.*

## Run Job

To run job script, open Windows Command Prompt (CMD) and change directory to scripts location and run:

* tbuild -f <job script> -v <job variable file> -j <job name>

i.e. > tbuild -f import -v jobVarImport -j s32td

The data from <S3BucketName>/<PrefixName>/<input file name> should be loaded into the <target table name> you defined in the import job.

17095 Via Del Campo, San Diego, CA 92127   [Teradata.com](http://www.teradata.com)

Teradata and the Teradata logo are registered trademarks of Teradata Corporation and/or its affiliates in the U.S. and worldwide. Teradata continually improves products as new technologies and components become available. Teradata, therefore, reserves the right to change specifications without prior notice. All features, functions, and operations described herein may not be marketed in all parts of the world. Consult your Teradata representative or Teradata.com for more information.

Copyright © 2021 by Teradata Corporation    All Rights Reserved.    Produced in U.S.A.

7.21 EBxxxx