teradata.

Teradata® Tools and Utilities for Microsoft Windows Installation Guide

Release 17.10

October 2021

B035-2407-061K
DOCS.TERADATA.COM

Copyright and Trademarks

Copyright © 1996 - 2021 by Teradata. All Rights Reserved.

All copyrights and trademarks used in Teradata documentation are the property of their respective owners. For more information, see Trademark Information.

Product Safety

Safety type	Description
NOTICE	Indicates a situation which, if not avoided, could result in damage to property, such as to equipment or data, but not related to personal injury.
CAUTION	Indicates a hazardous situation which, if not avoided, could result in minor or moderate personal injury.
WARNING	Indicates a hazardous situation which, if not avoided, could result in death or serious personal injury.

Third-Party Materials

Non-Teradata (i.e., third-party) sites, documents or communications (“Third-party Materials”) may be accessed or accessible (e.g., linked or posted) in or in connection with a Teradata site, document or communication. Such Third-party Materials are provided for your convenience only and do not imply any endorsement of any third party by Teradata or any endorsement of Teradata by such third party. Teradata is not responsible for the accuracy of any content contained within such Third-party Materials, which are provided on an “AS IS” basis by Teradata. Such third party is solely and directly responsible for its sites, documents and communications and any harm they may cause you or others.

Warranty Disclaimer

Except as may be provided in a separate written agreement with Teradata or required by applicable law, the information available from the Teradata Documentation website or contained in Teradata information products is provided on an "as-is" basis, without warranty of any kind, either express or implied, including the implied warranties of merchantability, fitness for a particular purpose, or noninfringement.

The information available from the Teradata Documentation website or contained in Teradata information products may contain references or cross-references to features, functions, products, or services that are not announced or available in your country. Such references do not imply that Teradata Corporation intends to announce such features, functions, products, or services in your country. Please consult your local Teradata Corporation representative for those features, functions, products, or services available in your country.

The information available from the Teradata Documentation website or contained in Teradata information products may be changed or updated by Teradata at any time without notice. Teradata may also make changes in the products or services described in this information at any time without notice.

Machine-Assisted Translation

Certain materials on this website have been translated using machine-assisted translation software/tools. Machine-assisted translations of any materials into languages other than English are intended solely as a convenience to the non-English-reading users and are not legally binding. Anybody relying on such information does so at his or her own risk. No automated translation is perfect nor is it intended to replace human translators. Teradata does not make any promises, assurances, or guarantees as to the accuracy of the machine-assisted translations provided. Teradata accepts no responsibility and shall not be liable for any damage or issues that may result from using such translations. Users are reminded to use the English contents.

Feedback

To maintain the quality of our products and services, e-mail your comments on the accuracy, clarity, organization, and value of this document to: docs@teradata.com.

Any comments or materials (collectively referred to as "Feedback") sent to Teradata Corporation will be deemed nonconfidential. Without any payment or other obligation of any kind and without any restriction of any kind, Teradata and its affiliates are hereby free to (1) reproduce, distribute, provide access to, publish, transmit, publicly display, publicly perform, and create derivative works of, the Feedback, (2) use any ideas, concepts, know-how, and techniques contained in such Feedback for any purpose whatsoever, including developing, manufacturing, and marketing products and services incorporating the Feedback, and (3) authorize others to do any or all of the above.

Contents

Chapter 1: Preparing for Installation 5
Welcome to Teradata Vantage 5
Prerequisites 5
Supported Releases 5
What's New 5
Installation Kit Contents 7
About Teradata Community 7
Teradata Tools and Utilities Prerequisites 7
System Requirements 8
Recommendation for Upgrading Teradata Tools and Utilities 8

Chapter 2: Installing New Software 10
About Installing Third-Party Software 10
About Installing Teradata Tools and Utilities 11
Installing Teradata Tools and Utilities Products 14

Chapter 3: Deploying Teradata Tools and Utilities 18
About Deploying Products 18
Uninstalling Existing Products from Client Computers 18
Creating a Software Repository 18
Creating a Batch File to Install the Products 20
Deploying the Batch File and the Packages via SCCM 21

Chapter 4: Configuring New Software 24
About Setting Network Parameters 24
About Configuring Teradata Call-Level Interface version 2 24
Verifying the Existence of the `hosts` File 24
Identifying a Database 25
Adding and Editing AP Entries in the `hosts` File 25
Setting Up Environment Parameters 26
Defining Paths to System Files 26
Setting Up Environment Variables 26
Speeding Up the CLIv2 Logon Process 27
Setting Up Port Number, Netrace, and Threadlogging 27
About clispb.dat 28
Setting System Parameters 28
Configuring ODBC Driver for Teradata 29
Configuring Teradata Parallel Transporter 29
Configuring Teradata Access Module for JMS 30

Chapter 5: Verifying the Complete Installation 31
About Verifying the Installation 31
Verifying Installation of the .NET Data Provider for Teradata 34
Setting the JDBC Driver Classpath 35
Accessing Lists of Installed Files 35

Chapter 6: Validating Product Readiness 37
TTU Install Directory Path 37
Validating AccessModuleAmazonS3 37
Validating AccessModuleAzure 37
Validating Kafka Access Module 38
Validating AccessModuleMQ 38
Validating AccessModuleNP 39
Validating AccessModuleOLEDB 39
Validating DataConnector API 41
Validating FastExport 41
Validating FastLoad 42
Validating Geospatial 42
Validating MultiLoad 43
Validating Teradata Java Message Service (JMS) 43
Validating Teradata Parallel Transporter 43
Validating Teradata Wallet 44
Validating TPump 44

Chapter 7: Uninstalling Software 45
About Uninstalling Previous Versions of Teradata Tools and Utilities 45
Generating a List of Installed Products 45
Uninstalling Teradata Tools and Utilities 14.10 and Later 46
Uninstall Teradata Tools and Utilities Pre-14.10 46
Interactively Uninstalling Teradata Tools and Utilities 50

Appendix A: Supporting Information 51

Appendix B: Product Parameter Values for Silent Installs 58

Appendix C: Installation Software 60

Appendix D: Troubleshooting the Teradata Tools and Utilities Installation 62

Appendix E: Installing .NET Data Provider for Teradata 64

Appendix F: About Teradata Community 68

Appendix G: Additional Information 70

1

Preparing for Installation

Welcome to Teradata Vantage

Teradata Vantage™ is our flagship analytic platform offering, which evolved from our industry-leading Teradata® Database. Until references in content are updated to reflect this change, the term Teradata Database is synonymous with Teradata Vantage.

Advanced SQL Engine is a core capability of Teradata Vantage, based on our best-in-class Teradata Database. Advanced SQL refers to the ability to run advanced analytic functions beyond that of standard SQL.

Prerequisites

You should be familiar with:

- Computer technology and terminology
- Relational database management systems

Supported Releases

This document supports the following versions of Teradata products:

- Teradata Vantage™ Release 2.0
- Teradata Tools and Utilities Release 17.10

The most current information about supported operating systems, supported Vantage versions, and product version numbers for all Teradata Tool and Utilities is available (at https://docs.teradata.com/) in a single spreadsheet titled *Teradata® Tools and Utilities Supported Platforms and Product Versions* (B035-3119).

What's New

FastLoad, FastExport, MultiLoad, and TPump Utilities

The Teradata FastLoad, Teradata FastExport, Teradata MultiLoad, and Teradata TPump utilities are now 64-bit instead of 32-bit.

TPTBase and TPTStream

The TPTStream package has been removed and the files have been added to the TPTBase package. The TPTBase package is 64-bit only (no 32-bit support).

Release Changes

- There is no longer a Database Management suite.
- Teradata Workload Analyzer has been discontinued.
- Teradata System Emulation Tool is now included in the Base suite.

Dynamically Configure the Features Screen

Starting with TTU 16.20.06, the ability to customize a Suite Feature Screen has been added. For more information, see Customizing a Suite.

User Selectable and Multi-Version Support

Starting with TTU 16.20, you can now install the Bundles in any location (see User Selectable Install Location for details). TTU 17.10 can co-exist with 15.10.10.00 and later releases (see Multi-Version Support for details).

Product Suites

The Teradata Tools and Utilities products are bundled in the following suite:

- Teradata Tools and Utilities - Base 17.10.06

For more information, see Software Packages.

Installing Product Suites

The Installer window contains a list of each product name in the suite. Use the checkboxes next to each product to install a subset of products. The following dependent products are automatically installed:

- Shared ICU Libraries for Teradata
- Teradata Call-Level Interface version 2
- Teradata Data Connector API

Open Database Connectivity (ODBC) Driver for Teradata is always selected and installed, unless it has already been installed.

The ability to customize a Suite has been added. For more information, see Customizing a Suite.

Uninstalling Product Suites

When you uninstall a suite, all primary products are uninstalled. The name of the suite is removed from **Programs and Features**. All dependent products are uninstalled when the last suite is uninstalled.

Accessing efixes

Suites that include efixes are available for download from https://support.teradata.com. For products that are available in a 32-bit version and a 64-bit version, each suite contains both versions (bitnesses) of the products. When you install a new version of the suite, efixes are applied automatically. Efixes can be installed using SCCM.

Downloading Standalone Products

You can download the following standalone products from https://downloads.teradata.com:

- .NET Data Provider for Teradata
- Open Database Connectivity (ODBC) Driver for Teradata
- Teradata Call-Level Interface version 2
- Teradata JDBC Driver
- Teradata Wallet

Note:

These products are intended to be used only to connect to Vantage using non-Teradata products, not for updating the Teradata Tools and Utilities suite of products.

Installation Kit Contents

The installation kit contains the following documentation media and application media:

- *Teradata Vantage™ User Documentation CD-ROM* (Vantage 2.00, Teradata Tools and Utilities17.10) (B035-1909)
- *Teradata Tools and Utilities Base Release* 17.10.xx (IBM AIX, Solaris, Linux, Ubuntu, Windows, Apple macOS), BCD0-2458-0000

If you did not receive all the installation media that you ordered, contact your Teradata Sales Representative.

About Teradata Community

Some Teradata Tools and Utilities products can be downloaded from the Teradata Community Downloads page.

Note:

These products are intended to be used to connect to Vantage using non-Teradata products, not to update the Teradata Tools and Utilities suite of products.

Teradata Tools and Utilities Prerequisites

Before you perform the procedures in this document, you must:

- Have Internet access
- Have network access
- Have Administrator privileges on the target machine
- Know which Teradata Tools and Utilities applications you are going to install
- Be familiar with naming conventions for Windows files and paths
- Have access to a bill of materials or similar list of Teradata Tools and Utilities software that was ordered
- Be familiar with software installation

System Requirements

You must have Administrator privileges for the computer on which you install TTU software.

The client computer must be able to communicate with a workstation-attached Vantage system that uses the TCP/IP protocol with Winsock version 2.0 support.

To find out which version of Vantage is compatible with Teradata Tools and Utilities, see the *Teradata® Tools and Utilities Release Definition*, B035-2029.

Hardware Specifications for Client and Server Systems

Component	Specification
CPU	1.5 GHz Pentium 4 (2.0 GHz recommended)
RAM	1 GB (2 GB recommended)
Disk space, free	800 MB minimum
Video display resolution	1024 x 768 minimum
Network card	Required
Media drive	Required
Operating system	For a list of supported operating systems, see *Teradata® Tools and Utilities Supported Platforms and Product Versions* (B035-3119).

Disk Space Requirements

The following amount of disk space is required to install this release:

350MB

Recommendation for Upgrading Teradata Tools and Utilities

Teradata notifies existing customers whenever software upgrades, higher-level releases, or maintenance releases occur. Upgrades are available from the Teradata Community. These upgrades or software patches can be downloaded by authorized customers or Teradata field support personnel with access.

For upgrades on the Teradata Community, go to Teradata Community Downloads, download the software, and install it.

For patches, log in to the Teradata Support Portal at https://support.teradata.com, download the software, and install it.

Installing New Software

About Installing Third-Party Software

Some Teradata Tools and Utilities products require the installation of third-party software. You can download most third-party software from the developer's website.

Determining the Current Version of .NET Framework

Prerequisite:

Install Microsoft .NET Framework 4.1 or later before you install .NET Data Provider for Teradata.

1. Do one of the following to determine the installed version of Microsoft .NET Framework:
   - Open a command prompt and type `APPWIZ.CPL`. The version of Microsoft .NET Framework is displayed in the list of installed programs.
   - See the article How to determine which versions and service pack levels of the Microsoft .NET Framework are installed.
2. If the system does not have the correct version installed, download one of the following installers:
   - Offline installer
   - Web installer
3. Follow recommendations on the installation page.

Installing Third-Party Software for Teradata Parallel Transporter Base

Download and install Progress DataDirect ConnectXE ODBC driver for Teradata.

Installing Third-Party Software for Teradata Parallel Transporter Wizard

The Teradata Parallel Transporter Wizard software must use the 32-bit Java SE Runtime Environment (JRE) version 1.4.2_06 or later on the 32-bit and 64-bit Windows platforms, but the PATH to the JRE is not added automatically.

On the 64-bit Windows platform, the Teradata Parallel Transporter Wizard cannot use the 64-bit JRE version.

1. Download and install the 32-bit JRE from Java SE Downloads on the 32-bit or 64-bit Windows platform.

2. On a 64-bit Windows platform, add the path to the 32-bit JRE `java.exe` command to the front of the `PATH` system environment variable:
   `C:\Program Files\Java\jre<version#>\bin;C:\Program Files\Teradata\client\17.10\Teradata Parallel Transporter\bin;`

Installing Third-Party Software for Teradata Access Module for WebSphere MQ

Note:

To display the version of WebSphere that is installed on the system, type `dspmqver` at the command line.

1. Download IBM WebSphere MQ 7.5 and install it according to the instructions in the downloaded package.

Installing Third-Party Software for Teradata JDBC Driver

The Teradata JDBC Driver requires one of the following environment versions:

- JDK/JRE 1.4.2
- JDK/JRE 5
- JDK/JRE 6
- JDK/JRE 7
- JDK/JRE 8
- JDK/JRE 9

The environment can be downloaded from Java Downloads.

Installing Third-Party Windows Help Program

Perform this task on all clients running Windows 7, Windows 8, or Windows Server 2008 to view online help files with a `.HLP` extension.

1. Install Teradata Tools and Utilities software before proceeding.
2. Go to Microsoft Support (Windows Help).
3. Select the `WinHlp32.exe` link for your version of Windows.
4. Select your language and click **Download**.
5. Select the 32-bit or 64-bit version then click **Next**.
6. Click **OK**.

About Installing Teradata Tools and Utilities

Installing in a different location is now fully supported. If the destination folder is changed from the default, all products are installed in that location. If the system is 64-bit, a folder named *prefix* is created for the 64-bit applications and one named '*prefix* (x86)' is created for the 32-bit applications. This follows the Windows convention of **Program Files** and **Program Files (x86)**.

All products should be installed using the Suite. Interactive and silent installs are supported and described in this section. Installing products outside of a Suite results in failures in the TTU products.

Note:

At the end of the install, the Teradata Tools and Utilities 64-bit bin and the Teradata Tools and Utilities 32-bit bin paths will be the only paths added to the PATH environment variable.

TTU products no longer depend on the PATH environment variable to find and locate the Teradata Tools and Utilities-dependent libraries. However, third-party applications can find TTU shared libraries and executables using the PATH variable.

Teradata Tools and Utilities package files are digitally signed.

Known Caveats

- Admin credentials are required to install.
- Installing products outside the Suite is not supported.
- If upgrading from a pre-14.10 TTU version, see Uninstalling Software. This version must be uninstalled before installing 17.10.
- If upgrading from a 14.10 or later version, the installation of 17.10 prompts to remove the existing version before installing 17.10.

  Note:

  If installing silently, the pre-existing version is removed automatically unless a parameter is provided to keep the previous version. This is only valid for 15.10.10 and later version. For more information, see Silently Installing Teradata Tools and Utilities Products.

Disk Space Requirements

The following amount of disk space is required to install this release:

~995 MB

Note:

Some products have both 32-bit and 64-bit packages. For a list of products and the package each one resides in, see Software Packages.

User Selectable Install Location

Teradata Tools and Utilities (TTU) Products can now be installed at a location provided by the user. All products of the same release (for example, 16.20, 17.00, 17.10 and so on.) must be installed at the same prefix.

For example, if the prefix is a drive (for example, `D:`), the 32-bit products will be installed in `D:\Program Files (x86)\Teradata\Client\17.10` and the 64-bit products will be installed in `D:\Program Files\Teradata\Client\17.10`. This follows the Windows convention. The same applies to a prefix like *drive*`:\TTUProducts`.

Multi-Version Support

The 17.10 release can co-exist with the 15.10.10.00 release and later versions. The *active release* is the most recent release installed. For example, if 15.10.10.00 is installed, then 15.10.10.00 is the active release. If 17.10 is installed after installing 15.10.10.00, 17.10 becomes the active release. The releases can be installed in any order. Installing 17.10 then 15.10.10.00 makes 15.10.10.00 the active release.

Clear the **Uninstall Previous versions** check box on the **Select Features** when installing 17.10 if the 15.10.* (* is 10 or greater) release is not uninstalled. This option is also available in silent install. See Silently Installing Teradata Tools and Utilities Products for details.

The TTUListProducts application can be used to set the active release if more than one release is installed. This application is available on the **Teradata Client 17.10** start menu item. TTUListProducts indicates the current active release and lists all the products that are installed on the system. The difference between the active release and non-active release is that the `bin` folder will be on `PATH`. A logoff or reboot will be required when the `PATH` is updated. This is a Windows restriction. The non-active release products can be used from the Start menu item for that release.

A bat file named `ttu_1710_PATH.bat` is provided in the `bin` folder to access the 17.10 release in a command prompt window if 17.10 is not the active release. This `bat` file will update the `PATH` to include the 17.10 `bin` folder.

Changing the Active Version using the TeradataListProducts Application

1. Select **Start > All Programs > Teradata Client 17.10 > TeradataListProducts**.

   Note:

   Right click on the **TeradataListProducts** menu item and select **Run as administrator** to change the active version. This is necessary even though the user is a member of the Administrator group.

   Alternatively, in the 32-bit install location; right-click `TTUListProducts.exe` in the `bin` folder and select **Run as administrator** to change the active version.

Installed Files

Starting with TTU 15.10.01, the Teradata Tools and Utilities products use relative locations to find and load dependent libraries. Do not move the installed files to any other location. The `bin` path of the release is the only path added to the `PATH` environment variable to enable third-party applications to locate the TTU shared libraries. All public executables and libraries are now located in the `bin` folder.

Installing Teradata Tools and Utilities Products

Installation can be done interactively or silently using the main installer (`setup.exe`) that is provided in the install suite `Windows\TTU` folder. See Installation Software for more details.

Installation Use Cases

Use cases for installing Teradata Tools and Utilities are listed below.

1. Individual User with admin privilege installing on their own PC.

   Interactively Installing Teradata Tools and Utilities Products or Silently Installing Teradata Tools and Utilities Products can be used.

2. Admin installing on the multiple servers (Dev, QA and then Production) to validate new release before updating all servers and client systems. Interactively Installing Teradata Tools and Utilities Products or Interactively Installing Teradata Tools and Utilities Products can be used.
3. Admin updating or installing TTU on client systems using a deployment tool such as SCCM.

   This needs to be done using deployment. See Deploying Teradata Tools and Utilities for instructions on how to deploy the TTU client software.

About Log Files

When you install Teradata Tools and Utilities, you have the option to have a log file named `InstallShield.log` created. Use the log file to troubleshoot any problems with the installation.

Setting Up a Log File for Interactive Installation

Do the following to create a log when installing interactively.

1. cd to the `Windows\TTU` folder.
2. Run the following command:

   ```
   >setup.exe /debuglog
   ```

   A `InstallShield.log` file is created in the same folder as `setup.exe`.

Use the following command to create the `install.log` file in a different location:

```
>setup.exe /debuglog"C:\PathToLog\setupexe.log"
```

Log Files for Silent Installation

When you install silently using `silent_install.bat`, the following two log files are created in `%temp%`.

- `TTUSuiteSilent_Install.log` contains logs for `TTUSuiteSilent.exe`, which is launched by `silent_install.bat`.
- `TTUSilentUninstall.log` contains the steps for the suite `setup.exe`,which is launched by `TTUSuiteSilent.exe`.

Log files are created for the individual products that are installed and are located in `%temp%`.

Interactively Installing Teradata Tools and Utilities Products

The TTU Windows Suites are available on media (request to IPP) or it can be downloaded from TSS or IPP. If using media, then insert it into the drive. If the **Welcome** dialog box does not appear shortly after the media is inserted, use Windows Explorer to navigate to the main folder of the media. Double-click `setup.bat` to start the installation process.

When you install Teradata Tools and Utilities products, all dependent products for your selection are installed automatically. On a 64-bit system, both 32-bit and 64-bit versions of the dependent products are installed.

Close all active Windows programs and temporarily disable any virus protection software running on the client computer.

1. Double-click `setup.bat`.

   Note:

   For a list of products and the media/package each one resides on, see Software Packages.

2. In the **Welcome to Teradata Tools and Utilities** dialog box, click **Install Product**.
3. Select the installation language, then click **Next**.
4. In the **Welcome** dialog box, click **Next** .
5. Read and accept the license agreement, then click **Next**.
6. Click **Next** to install to the default directory; to change directories, click **Browse**.
7. In the **Select Features** dialog box, select the products that you want to install.

   Note:

   The checkbox for Open Database Connectivity (ODBC) Driver for Teradata is always selected; this product is installed with the Base and DBM Suites.

8. Choose whether you want the installation process to uninstall previous versions of the following products.
9. Click **Install**.
10. When the installation is complete, click **Finish**.

11. If media was used, remove the media from the drive.

Note:

The above steps apply to the first Suite installed for a particular version. Steps 5 and 6 are skipped when installing another Suite of the same version.

Silently Installing Teradata Tools and Utilities Products

You can use a single DOS command to silently install Teradata Tools and Utilities products for a particular suite. Each Suite has a unique `silent_install.bat` file.

Note:

For a list of products and the media/package each one resides on, see Software Packages.

1. Close all active Windows programs and temporarily disable any virus protection software running on the client computer.
2. Uninstall previous versions of the Teradata Tools and Utilities software if the version is 14.10 or lower. See Uninstalling Software for more information.
3. If the installation source files are not on the system disk, insert the appropriate media.
4. From a command prompt, change directory to `\Windows\TTU\`.
5. Run the batch file:
   ```
   silent_install.bat "PRODUCT" <installDir> <keepPrevVer>
   ```

   The **installDir** parameter is optional. The installDir can be a drive or fully qualified folder. All Suites of the same release (for example, 17.10) are installed to the same prefix. If the installDir parameter is not specified, the default prefix is `C:\Program Files` [or `C:\Program Files (x86)`]. Because products from the same release must be installed to the same prefix, if a product has already been installed to a different prefix, the prefix entered here is ignored.

   The **keepPrevVer** parameter is optional. If specified, then the previous release will not be removed.

   Note:

   This applies only to 15.10.10 or later releases. All pre-15.10.10.00 releases will always be removed since those releases cannot co-exist with 17.10. The 15.10.10.00 release and later can co-exist with 17.10.

   For a list of **PRODUCT** parameter values and other details about the command syntax to run the batch file, see Product Parameter Values for Silent Installs.

Note:

- Some products include a 32-bit version and a 64-bit version. On a 32-bit system, only the 32-bit version of the product is installed. On a 64-bit system, both versions of the product are installed.
- Beginning with Teradata Tools and Utilities 14.10, you can start Teradata Wallet from the **Start** menu.
- See Verifying the Complete Installation and/or Validating Product Readiness to verify the installed products are installed correctly.

Deploying Teradata Tools and Utilities

About Deploying Products

To provide complete examples, several sections such as the summary below, "Creating a Software Repository", and "Creating the Batch File to Install the Products", use SCCM as an example. However, the steps for deployment are the same steps required for any deployment process.

A summary of tasks required to deploy products is listed below:

1. Install third-party product dependencies on the client computers. See About Installing Third-Party Software for details.
2. Uninstall existing products from the client computers.

   Note:

   Alternatively, you can have the batch file uninstall existing products before the new products are installed.

3. Create a software repository on the system that is running the SCCM server.
4. Create the batch file to install the products during SCCM deployment.
5. Use System Center Configuration Manager to deploy the batch file and the product packages to the client computers.

Uninstalling Existing Products from Client Computers

If you plan to include the command below in the batch file, it is not necessary to complete this procedure.

1. On each client computer, run one of the following commands:

Option	Description
14.10 or later	`silent_uninstall.bat ALL`
pre-14.10	`cscript uninstall_TTU.vbs ALL`

   All existing Teradata Tools and Utilities products are uninstalled. Each Suite has a unique `silent_uninstall.bat` file.

Related Information:

Creating a Batch File to Install the Products

Creating a Software Repository

1. On the system that runs the SCCM server software, create a folder for use as the root of the software repository.
   For example, name the folder `TTU17.10_SoftwareRepository`.
2. Check https://support.teradata.com to see if a newer version of the package file for the Teradata Tools and Utilities bundled suite is available. If a newer version is available, download it, extract it, and use it as the source instead of the installation media.
3. In the file structure of the source, browse to the `\Windows\TTU` folder.
4. Create a `TTU` directory in the directory created in step 1. Create an `x86` folder and an `x64` directory in the `TTU` directory.
5. For each Teradata Tools and Utilities product, copy the folder to the `x86` or `x64` folder of the software repository based on the package bitness.
   The 32-bit packages must be in the `x86` folder, and the 64-bit packages in the `x64` folder. For example, the CLIv2 32-bit folder should be copied to the `x86` folder at the root of the repository.

   After you copy each folder, the folder and file structure should look like this.

Note:

The files (directories are bold) listed below are mandatory. The packages listed under the `x86` and `x64` directories are the dependent packages. Please add the required primary packages.

```
TTU17.10_SoftwareRepository
        ThirdPartyLicensesTTU.txt
        TTU
                setup.exe
                silent_install.bat
                SuiteSetup.exe
                TTUListProducts.exe
                TTUSuiteSilent.exe

        x86
                CLIv2-32bit
                        setup.exe
                        ...
                DataConnector-32bit
                        setup.exe
                        ...
                SharedICU-32bit
                        setup.exe
                        ...
        x64
                CLIv2-64bit
```

```
                    setup.exe
                    ...
            DataConnector-32bit
                    setup.exe
                    ...
            SharedICU-64bit
                    setup.exe
                    ...
```

6. Copy the following files from the `\Windows\TTU` folder to the root of the `software repository\TTU` directory:

   ```
   setup.exe
   silent_install.bat
   SuiteSetup.exe
   TTUListProducts.exe
   TTUSuiteSilent.exe
   ```

7. Copy the `ThirdPartyLicensesTTU.txt` from the `\Windows` folder to the root of the `software repository` directory.

Creating a Batch File to Install the Products

The batch file that you create in this procedure will be used to automatically perform the following tasks:

- Uninstall existing Teradata Tools and Utilities products (if you include that instruction)
- Install new Teradata Tools and Utilities products

1. Create a batch file on the root of the software repository.
   For example, name the file `Deploy_TTU.bat`.
2. Add the appropriate text to the file.

Option	Description

Already uninstalled existing products	`REM Input: No input required`
	`REM Logging: The log file for each package will be located in the %temp% folder`
	
	`silent_install.bat "PRODUCT"`

Need to uninstall 14.10 or later	`REM Input: No input required`
	`REM Logging: The log file for each package will be located in the %temp% folder`
	
	`silent_uninstall.bat ALL`
	
	`silent_install.bat "PRODUCT"`

Option	Description

Need to uninstall 14.10 or earlier	`REM Input: No input required`
	`REM Logging: The log file for each package will be located in the %temp% folder`
	
	`cscript uninstall_TTU.vbs ALL`
	
	`silent_install.bat "PRODUCT"`

where **PRODUCT** is a set of parameter values associated with the products that will be installed. The parameter values, including a set of strings you can copy, are found at: Product Parameter Values for Silent Installs.

Deploying the Batch File and the Packages via SCCM

1. Create a package and configure it for deployment.
   a. Select **Start > All Programs > Microsoft System Center 2012 > Configuration Manager**. Allow a few seconds for the software to connect to SQL Server.
   b. Select **Software Library**.
   c. Expand **Overview > Application Management > Packages**.
   d. Right-click **Packages** and select **Create Package**. The **Create Package and Program Wizard** appears.
   e. Add information to the following text boxes:

Text Box	Value
Name	The name of the package you are creating (for example, `TTU Base 17.10`)
Manufacturer	`Teradata Corporation`
Language	`EN`
Version	`17.10`

   f. Select **This package contains source files**. The **Set Source Folder** dialog appears.
   g. For the **Source folder** text box, browse to the folder containing the source files (the software repository).
   h. In the **Create Package and Program Wizard**, click **Next**.
   i. Click **Standard program** then click **Next**.
   j. Enter the **Name** of the program, for example, `TTU Base 16.20`.
   k. In the **Command line** text box, enter the following command:

batch_filename **"PRODUCT"**

where:

- *batch_filename* is the name of the batch file on the root of the software repository
- **PRODUCT** is a set of parameter values associated with the products you want to install. The parameter values, including a set of strings you can copy, are found at: Product Parameter Values for Silent Installs

---

Important:

If the batch file you specify contains a command to execute `silent_install.bat`, do not add **`PRODUCT`** parameter values to the **Command line** text box. An example of a batch file that contains a command to execute `silent_install.bat` is the file named `Deploy_TTU.bat` mentioned in Creating a Batch File to Install the Products. This file already contains a list of **`PRODUCT`** parameter values.

Command Example: Install all products that are part of the Base suite:

```
silent_install.bat
"BTEQ,FastExport,FastLoad,JMSAM,MultiLoad,KAFKAAM,AZUREAM,S3AM,NamedPip
esAM,
OLEDBAM,Preprocessor,TDNETDP,TDWallet,TPTBase,TPump,
TSET,WebsphereAM,TDGeoSpatial"
```

Open Database Connectivity (ODBC) Driver for Teradata is always installed. While you do not need to specify the value for this product, copy the product package file to the root of the software repository.

There is no need to specify the values for the dependent products as they will automatically be installed. However, be sure to copy the following dependent product package files to the root of the software repository:

- Shared ICU Libraries for Teradata
- Teradata Call-Level Interface version 2
- Teradata Data Connector API

l. Select **Normal** from the **Run** drop-down list.
When the package is deployed to the client, a command prompt appears. If you do not want the command prompt to be displayed, select **Hide**.

m. Select **Whether or not user is logged in** from the **Program can run** drop-down list, then click **Next**.

n. Click **Next** to accept the default Requirements settings.

o. Click **Next** again, then **Close**.

2. Deploy the package (make the package available to clients).
   a. Right-click the name of the package (for example, TTU Base 17.10) and select **Deploy**.
   The **Deploy Software Wizard** is displayed.
   b. Next to **Collections**, click **Browse**.
   c. Select **All Systems** then click **OK**.
   d. Click **Next**.

e. Click **Add** then select **Distribution Point**.
f. In the **Name** column, select the checkbox next to the distribution point then click **OK**.
g. Click **Next**.
h. Click **Purpose** and select one of the following options:
   - **Available**: During deployment, installation on the client will be interactive.
   - **Required**: During deployment, installation on the client will be silent. If a dialog box appears, you can select **Allow user to interact with installation**.
i. Click **Next**.
j. [Optional] Schedule when the deployment will become available and when it will expire.
k. Click **Next**.
l. Select **Software installation** then click **Next**.
m. Select one of the following options from the **Deployment options** drop-down list:
   - For a fast network, select **Download content from distribution point and run locally**.
   - For a slow network, select **Do not run program**.
n. Click **Next**.
o. Click **Next**.
p. Click **Close**.

Note:

It might take some time for the deployed software to become available on the client.

---

3. [Optional] Manually install the package at the client.
   a. At the client, select **Start > All Programs > Microsoft System Center 2012 > Configuration Manager > Software Center**.
   b. Select the package you want to install.
   c. Click **Install**.

Configuring New Software

About Setting Network Parameters

The Internet address and the symbolic names for Vantage are considered the network parameters. It is assumed that network software is installed correctly.

About Configuring Teradata Call-Level Interface version 2

Note:

The parameters to resolve the host name to IP address must be configured for Open Database Connectivity (ODBC) Driver for Teradata and any other connection driver.

Set the following parameters:

Network Parameters

Setting the correct network parameters guarantees that Teradata Call-Level Interface version 2 communicates with the target Vantage database.

Environment Parameters

The environment parameters provide Teradata Call-Level Interface version 2 with parameter file location information and enables Teradata Call-Level Interface version 2 to run in parallel mode. Teradata strongly recommends that you set the environment parameters.

System Parameters

The Teradata Call-Level Interface version 2 system parameters are used internally by Teradata Call-Level Interface version 2 to customize services and features. Set the system parameters only if special requirements exist that the default setting cannot meet.

Verifying the Existence of the `hosts` File

1. Verify the `hosts` file is located in the `%windir%\system32\drivers\etc` directory.
2. Do one of the following:
   - If you are running a Name Server, verify the `hosts` file is on the Name Server and that network software is configured to access the Name Server.

     The Teradata Tools and Utilities software uses generic TCP/IP GetHostByName services.
   - If not running a Name Server, verify the `hosts` file contains a list of all systems on the LAN that can be accessed.

Vantage must be identified in this file if a system is going to communicate with it. A `hosts` file exists on each computer system in the network.

Identifying a Database

1. Identify each TCP/IP Communications Processor (COP), Applications Processor (AP), or UNIX node running the Teradata Gateway that the client computer will be communicating with Vantage.

   In the table below, AP refers to all of the following:

   - TCP/IP Communications Processor (COP)
   - Applications Processor (AP)
   - Node running the Teradata Gateway
2. Make an entry in the `hosts` file for each TCP/IP COP, AP, or UNIX node running the Teradata Gateway.

Platform	hosts File Entries
Vantage on an AP platform (System 3600)	APs
Vantage on a V2 computer (UNIX)	nodes

Adding and Editing AP Entries in the `hosts` File

1. Once the APs are identified, use a text editor to open the client `hosts` file and add or edit the AP entries. Use the following syntax when adding entries to the `hosts` file:

```
a.b.c.d          dbnameCOPn      [aliasCOPn]
```

a.b.c.d

The Internet address is a four-byte numeric value that identifies an AP and its location on the network. An Internet address is assigned to each AP when Vantage is configured.

dbcname

The database using the AP is 1 to 70 characters in length. The first character must be alphabetic; where the remaining characters can be alphanumeric. The System Parameter Block file, `%Program Files%\Teradata\Client\`*version*`\clispb.dat`, lists dbc as the default dbcname. dbc can be assigned as the dbcname of a database with which the client computer communicates. This simplifies network debugging and avoids problems when Name servers become available. If the client computer communicates with more than one database on the same TCP/IP LAN, use dbc as the dbcname for the more heavily used database. Any other names can be used for any remaining instances of Vantage.

COP*n*

The parameter that assigns a number to a particular AP that Vantage uses. COP is a fixed string of characters that must always be specified. The value of n can range from 1 to the total number of "APs" in a COP group. A COP group is a set of "APs" that operate together for a particular database. "APs" in the same COP group share session balancing and session recovery. Each AP within a COP group must be assigned a unique number starting with 1.

alias*COP*n

The alias name, specifies an alternate name for a database AP. Its use is optional. More than one aliasname is permitted. An aliasname need not be unique across a network, but the `hosts` file for a computer must not contain the same aliasname for more than one AP. Suggestion: the *n* in each alias COP*n* equals *n* in the corresponding dbcname COP*n*.

For example, a client computer accesses one database (named dbc) through two TCP/IP APs with addresses 101.0.0.21 and 101.0.0.22 on the same logical TCP/IP LAN. Each AP is assigned two alias names. The following lines show example entries in the `hosts` file for the two APs:

```
101.0.0.21 dbcnode1 dbccop1 tdcop1
101.0.0.22 dbcnode2 dbccop2 tdcop2
```

Setting Up Environment Parameters

Environment parameters allow the runtime environment to be specified for each Teradata Tools and Utilities user. If system environment parameters are changed, the client computer must be rebooted for the changes to take effect.

Defining Paths to System Files

1. Define the path to the Teradata Call-Level Interface version 2 system parameter block file by setting the `COPLIB` environment variable.
   `COPLIB` is the path to the Teradata Call-Level Interface version 2 system parameter block file, `clispb.dat`.
2. Define the path to the Teradata Call-Level Interface version 2 anomaly log file by setting the `COPANOMLOG` environment variable.
   `COPANOMLOG` is the path to the anomaly log file, a trace log file that displays Teradata Call-Level Interface version 2 activities including network-related error information. This environment variable should be defined only for debugging purposes.

Setting Up Environment Variables

1. Open the **System Properties** dialog box.

Tip:

Open a command prompt and type `SYSDM.CPL` .

2. Click the **Advanced** tab.
3. Click **Environment Variables** to add a new variable or edit an existing variable.

Speeding Up the CLIv2 Logon Process

Increase the speed of the CLIv2 logon process by limiting the number of host name resolution requests issued by the applications.

1. Set an environment variable with the name equal to the machine name with a specified number of COPS available for it.
   set *machine_name=cops*

   machine_name
   name of the Vantage system, or tdpid

   cops
   number of COPS

2. Open the **System Properties** dialog box.

Tip:

Open a command prompt and type `SYSDM.CPL` .

3. Click the **Advanced** tab.
4. Click **Environment Variables** and add or edit the *machine_name* system environment variable.
5. Set the `SHOGUN` system environment variable to 3 as follows:
   `set SHOGUN=3`
   Setting SHOGUN=3 causes CLIv2 to issue GetHostByName for SHOGUNCOP1, SHOGUNCOP2, and SHOGUNCOP3 only. If one of the GetHostByName requests fails, the successive requests are not issued.
6. Add or edit the `SHOGUN` system environment variable.

Setting Up Port Number, Netrace, and Threadlogging

Note:

Before you change the `TDMSTPORT` system environment variable to use a port number other than the default of 1025, contact Teradata Support to understand the ramifications.

1. Specify the port number *nnnn* used by the Teradata Tools and Utilities software by setting the TDMSTPORT system environment variable:
   `set TDMSTPORT=nnnn`
2. Open the **System Properties** dialog box.

   Tip:
   Open a command prompt and type `SYSDM.CPL` .
3. Click the **Advanced** tab.
4. Click **Environment Variables** and add or edit the TDMSTPORT system environment variable.
5. Enable Detailed COPANOMLOG Logging by setting up the system environment variable NETRACE:

   ```
   set NETRACE=1
   ```

   Setting NETRACE=1 enables detailed network information between the client computer and Vantage. This information is logged in the file defined in the system environment variable COPANOMLOG.
6. Add or edit the NETRACE system environment variable.
7. Enable Multiple COPANOMLOG File Creation by setting up the system environment variable THREADLOGGING:

   ```
   set THREADLOGGING=1
   ```

   Threadlogging supports multi-threaded Windows applications. Multi-threaded logging is enabled by setting THREADLOGGING=1. For example, if COPANOMLOG is `c:\temp\coplog` and THREADLOGGING is enabled, logs are created as `coplog.xxx.txt`, where *xxx* is the thread_id in hexadecimal:

   ```
   c:\temp\coplog.1a4.txt
   c:\temp\coplog.db8.txt
   c:\temp\coplog.b34.txt
   etc.
   ```

   Each thread that uses a CLI connection creates its own individual log file.
8. Add or edit the THREADLOGGING system environment variable.

About clispb.dat

The `clispb.dat` file (the Teradata Call-Level Interface version 2 System Parameter Block) contains default values for variables used by CLI application programs. In `clispb.dat`, Vantage sets the initial values to duplicate the Teradata Call-Level Interface version 2 internal defaults.

Setting System Parameters

1. Edit the `clispb.dat` file to meet system requirements.

   This file is located in the `etc` folder of the release.

For more information, see the *Teradata® Call-Level Interface Version 2 Reference for Workstation-Attached Systems*, B035-2418.

Configuring ODBC Driver for Teradata

To use the Open Database Connectivity (ODBC) Driver for Teradata, you must first configure the ODBC driver and an ODBC data source.

The *ODBC Driver for Teradata® User Guide*, B035-2526 contains the following information:

- Complete procedures to configure the ODBC Driver for Teradata
- Information about configuring the data source

Configuring Teradata Parallel Transporter

Editing the Teradata Parallel Transporter Configuration File

If you installed Teradata Parallel Transporter Base, you can modify the local configuration file (`twbcfg.ini`) to specify the paths to the following directories (folders):

- `checkpoint` directory: Contains the checkpoint files for Teradata Parallel Transporter jobs. Teradata Parallel Transporter uses checkpoint files to store temporary information needed to restart a job that does not complete successfully.
- `log` directory: Contains the binary logs for Teradata Parallel Transporter jobs. Each Teradata Parallel Transporter job writes a log in the `log` directory.

Note:

You do not need to reboot after you edit the configuration file.

---

1. Clear the read-only attribute for the configuration file.

   The configuration file is located in the following folder:

   ```
   %PROGRAMFILES%\Teradata\Client\version\Teradata Parallel Transporter\
   ```

   where *version* is the short version (for example, `17.10`).
2. Specify the path to the checkpoint directory in the configuration file.
   - 32-bit

     ```
     CheckpointDirectory= 'C:\Program Files
     (x86)\Teradata\Client\version\Teradata Parallel Transporter\checkpoint'
     ```

   - 64-bit

```
CheckpointDirectory= 'C:\Program Files\Teradata\Client\version\Teradata
Parallel Transporter\checkpoint'
```

3. Specify the path to the log directory in the configuration file.
   - 32-bit

     ```
     LogDirectory='C:\Program Files (x86)\Teradata\Client\version\Teradata
     Parallel Transporter\logs'
     ```

   - 64-bit

     ```
     LogDirectory='C:\Program Files\Teradata\Client\version\Teradata
     Parallel Transporter\logs'
     ```

Editing the Teradata Parallel Transporter Properties File

If the folder for the Log Directory to `C:\Logs\Teradata Parallel Transporter` *version* is modified and the Teradata Parallel Transporter Wizard is installed, use the following procedure to update the *LOG_DIRECTORY* entry in the following file:

TPT_install_directory`\tptwizard\config\tptwizard.properties`

1. Open the `tptwizard.properties` file.
2. Update the `LOG_DIRECTORY` line to `LOG_DIRECTORY=c:\Program Files\Teradata\Client\`*version*`\Teradata Parallel Transporter\logs\\`.

   The folder path is an example. You can specify a different path.

   Note:

   The folder for the LOG_DIRECTORY entry in the `tptwizard.properties` file must be separated by double backslashes (`\\`) and must end with double backslashes (`\\`).

   For information about how to interpret the job log, see the *Teradata® Parallel Transporter Reference*, B035-2436.

Configuring Teradata Access Module for JMS

For instructions on how to configure Teradata Access Module for JMS, see the `readme` file.

1. Open the `readme` file located in `%Program Files(x86)%\Teradata\Client\`*version*`\JMSAXSMOD` folder.

Verifying the Complete Installation

About Verifying the Installation

Related Information:

Validating the BTEQ Installation
Verifying the Installation Using CLI Sample Program
Generating a List of Installed Products

Validating the BTEQ Installation

See the *Basic Teradata® Query Reference*, B035-2414 for detailed information on how to use BTEQ.

1. Start BTEQ:

   Start > All Programs > Teradata Client 17.10 > Teradata BTEQ

   Output similar to the following appears:

   ```
   Teradata BTEQ 17.10.00.00 for     platform.
   Copyright 1984-2017, Teradata Corporation. ALL RIGHTS RESERVED.
   Enter your logon or BTEQ command:
   ```

2. Log on to Vantage:

   ```
   .logon dbcpath/username
   ```

   Enter the BTEQ logon command with a logon ID. Include the dbcpath if Vantage dbcpath is not the default i_dbcpath specified in `clispb.dat`.

3. When prompted, enter the password.
   For example, if Vantage has a dbcpath of Finance, the username is `fml`, and the password is `fml`, type:

   ```
   .logon Finance/fml
   Password: fml
   ```

   If the logon is successful, output similar to the following appears:

   ```
   *** Logon successfully completed.
   *** Teradata Database Release is 17.10.00.00
   *** Teradata Database Version is 17.10.00.00
   *** Transaction Semantics are BTET.
   *** Session Character Set Name is 'ASCII'.

   *** Total elapsed time was 1 second.

   BTEQ -- Enter your DBC/SQL request or BTEQ command:
   ```

Note:

For information about security and connecting to Vantage, see *Teradata Vantage™ - Advanced SQL Engine Security Administration*, B035-1100.

4. Submit a sample Teradata SQL query:

   ```
   SELECT time;
   ```

   If the test passes, output similar to the following appears.

   ```
   *** Query completed. One row found. One column returned.
   *** Total elapsed time was 1 second.
                       Time
                   --------
                   11:09:11
   ```

5. To see the version numbers for BTEQ and the key Vantage software it depends on:

   ```
   .SHOW VERSIONS
   ```

   which produces output similar to the following:

   ```
   .SHOW VERSIONS
    BTEQ Version 17.10.00.00 for platform
    CLIV2      : 17.10.00.00
    COPERR.H   : 17.10.00.00
    COPTYPES.H : 17.10.00.00
    DBCAREA.H  : 17.10.00.00
    DBCERR.H   : 17.10.00.00
    DBCHQEP.H  : 17.10.00.00
    MOSIIF.H   : 17.10.00.00
    PARCEL.H   : 17.10.00.00
    PIOM       : 17.10.00.00
    TDICU      : 17.10.00.00
    TDWALLET   : 17.10.00.00
    Database   : 17.10.00.00

     BTEQ linking date is Feb  4 2014
   ```

6. Log off of Vantage and exit BTEQ.

Validating the ODBC Installation

Refer to *ODBC Driver for Teradata® User Guide*, B035-2526 for detailed information on how to use ODBC.

The tdxodbc tool can be used to validate the installation of ODBC Driver for Teradata. It is located in the directory:

prefix`\Teradata\Client\`*TTUversion*`\bin`

- Run the command:
  `tdxodbdc -c SQLDriverConnect -C`
  `"DRIVER={Teradata};DBCName=`*databasename*`;UID=`*username*`;PWD=`*password*`;" -t`

  where

databasename – Name or IP address of the database.
username – Database username.
password – Database password.

If ODBC Driver for Teradata is installed correctly, the tool displays information that it successfully connected to the database.

Verifying the Installation Using CLI Sample Program

1. In the makefile `makesamp.win`, add the correct paths to the `include` files.

   By default the `makesamp.win` file is located in the following folder:

   `\`*program-files*`\Teradata\Client\`*version*`\CLIv2\clisamp`

2. Run the makefile to create the executable module `clisamp`:

   ```
   # nmake -f makesamp.win
   ```

   The default configuration is **CFG="clisamp - Win32 Debug"**. Also supported is **CFG="clisamp - Win32 Release"**.

3. Change directory to the appropriate subfolder:

   `Release`

   `Debug`

4. Run the sample program by entering one of the following commands:

   - Run the sample program using the default logon information (`dbc/systemfe,service`):

     ```
     clisamp
     ```

   - Show help for the sample program using the **-h** option:

     ```
     clisamp -h
     ```

   - Run the sample program using a user-provided input string:

     `clisamp proddev/dbc,`*password*

     where:

     - `proddev` is the tdpid
     - `dbc` is the user
     - *password* is the password

The program does the following:

- Logs on using the default logon information (or the user-specified logon information) as shown above.
- Executes a simple query: `SELECT * from dbc.sessioninfo;`
- Writes the results of the query to a `CLISAMP.DAT` file and displays information similar to the following:

```
clisamp
CLIv2    version is     version
```

```
MTDP     version is     version
MOSIOS   version is     version
MOSIDEP  version is     version
OSERR    version is     version
Logging on to --> dbc/systemfe,service
writing out to --> CLISAMP.DAT
submitting --> select * from dbc.sessioninfo;
writing out to --> CLISAMP.DAT
Logging off.
```

Note:

For information about security and connecting to Teradata Vantage, see *Teradata Vantage™ - Advanced SQL Engine Security Administration*, B035-1100.

Generating a List of Installed Products

1. Start the application that generates a list of products using one of the following methods:

Method	Procedure
From the Start menu	• Select **Start > All Programs > Teradata Client 17.10 > Teradata List Products**. Alternatively, in the <*prefix*>\Teradata\Client\17.10\bin folder, double-click TTUListProducts.exe.
From a command prompt	a. Open a command prompt. b. Change directory to the \Windows\TTU\ folder (or to the appropriate folder on the physical media): `cd Windows\TTU\` c. Run the following command: `TTUListProducts.exe -s -f` *filename* where: • **-s** runs the application in silent mode • **-f** *filename* allows you to specify the name of the output file You can specify **-s** and **-f** *filename* at the same time.

A file named TTUListProducts*timestamp*.txt is created on the Desktop, where *timestamp* is the time the file was created. It contains a list of all of the Teradata Tools and Utilities products that are installed on the system.

Note:

If you used the **-f** *filename* option at the command prompt, the filename matches what you specified.

Verifying Installation of the .NET Data Provider for Teradata

The complete installation of .NET Data Provider for Teradata includes an unsupported ad-hoc query tool `NQuery.exe`. Use this tool to test successful installation of .NET Data Provider for Teradata and to verify connection to Vantage.

1. Navigate to `%PROGRAMFILES%\Teradata\NET Data Provider for Teradata\`*version* `\Samples\Bin\` or `%PROGRAMFILES(x86)%\Teradata\NET Data Provider for Teradata\`*version* `\Samples\Bin\`, depending on the Windows platform.
2. Double-click `NQuery.exe`.
3. Select **Connection > Connect**.
4. Select .NET Data Provider for Teradata for the **Data Provider**.
5. Enter values in the **Connection String** box.
   The connection string is a semicolon-separated list of name=value pairs. It uses the format Data Source = *system_name* ; User Id = *username* ; Password = *password* ; where:
   - Data Source = *system_name* (the tdpid of the Vantage system)
   - User Id = *username* (the logon name by which the user of this client computer accesses Vantage)
   - Password = *password*, the password for the *username*

   For example, to connect to database vixen with username jackieo and password sVnglass3s, type:

   ```
   Data Source = vixen ; User Id = jackieo ; Password = sVnglass3s ;
   ```
6. Click **Connect**.
7. Type `Select Date;` in the **SQL Command** pane and click **Execute**.
   The Vantage system you connect to returns the current system date in the **Command Result** pane.
8. Click **Connection > Disconnect**.
9. Click **File > Exit**.

Setting the JDBC Driver Classpath

The Teradata JDBC Driver is distributed as platform-independent jar files.

1. Copy the jar files into a directory.
2. Set the classpath to refer to the necessary jar files, including:
   - `terajdbc4.jar`
   - `tdgssconfig.jar`

Accessing Lists of Installed Files

Complete the following steps to verify the installation created all the necessary files.

1. Log into https://support.teradata.com.
2. Enter the Knowledge Article Number in the search field.

Operating System	Knowledge Article Number
Apple macOS	KAP2C009E
IBM AIX	KAP2C00A6
CentOS, OEL, RedHat, SLES, Ubuntu	KAP2C00AA
Oracle Solaris on SPARC and AMD Opteron	KAP2C00AE
Windows	KAP2C00B2

Validating Product Readiness

TTU Install Directory Path

See User Selectable Install Location for details on the directory path you are prompted to specify for the procedures in this section.

Install location:

- 32-bit INSTALL_DIRECTORY is `<prefix>\Program Files (x86)\Teradata\Client\17.10`
- 64-bit INSTALL_DIRECTORY is `<prefix>\Program Files\Teradata\Client\17.10`

where the default prefix is `C:`.

Validating AccessModuleAmazonS3

For more information about Teradata Access Module for S3, see *Teradata® Tools and Utilities Access Module Reference*, B035-2425.

1. Open a command prompt.
2. Create the `HOME` environmental variable.
3. Create the `%HOME%/aws` directory.
4. Create the `config` file and the `credentials` File under `%HOME%/aws` directory.
5. Change directory to `TTU_install_directory/sample/validate`.
6. Run the command `/validate.bat` *S3BucketName S3DirectoryName S3ObjectName* where the following is true:
   - *S3BucketName* is the S3 bucket name.
   - *S3DirectoryName* is the folder name created under S3 bucket to store the S3 object.
   - *S3ObjectName* is the S3 object name.

Validating AccessModuleAzure

Prerequisite:

Verify Teradata Parallel Transporter is installed on the system.

For more information about Teradata Access Module for Azure Blob, see *Teradata® Tools and Utilities Access Module Reference*, B035-2425.

1. Open a command prompt.

2. Create the `HOME` environment variable.
3. Create the `%HOME%/azureaxsmod` directory.
4. Create the `credentials` File under `%HOME%/azureaxsmod` directory.
5. Change directory to `TTU_install_directory/sample/validate`
6. Run the command `azureaxsmod_validate.bat` *<NODEID>* *<UserName>* *<UserPassword>* *<CONTAINERNAME>* *<OBJECTNAME>* where the following is true:
   - *<NODEID>* is a database name id.
   - *<UserName>* is a database User Name.
   - *<CONTAINERNAME>* is the azure blob storage container name.
   - *<OBJECTNAME>* is the azure blob object name.

   The script executes `azureaxsmod_qstart.tpt` in the `quickstart` directory.

Validating Kafka Access Module

Prerequisite:
Verify Teradata Parallel Transporter is installed on the system.

For more information about Teradata Access Module for Kafka, see *Teradata® Tools and Utilities Access Module Reference*, B035-2425.

1. Open a command prompt.
2. Create a database username and password.
3. Change directory to `INSTALL_DIR/AXSMOD/quickstart`.
4. Change directory to `INSTALL_DIR/AXSMOD/validate`.
5. Run the command `kafkaaxsmodvalidate.bat` *<Tpdid>* *<UserName>* *<UserPassword>* *<TOPICNAME>* *<BROKER>* *<PARTITIONS>* where the following is true:
   - *<Tpdid>* is a database name id.
   - *<UserName>* is database User Name.
   - *<TOPICNAME>* is a Kafka Message Server topic name.
   - *<BROKER>* is a kafka Message Server Broker, e.g., KAFKASERVERIPADDRESS:PORTNO.
   - *<PARTITIONS>* is partitions of the topic, e.g., 0,1.

The validation script executes `kafkaaxsmod_qstart.tpt` in the `quickstart` directory.

Validating AccessModuleMQ

For more information about Teradata Access Module for WebSphere MQ, see *Teradata® Tools and Utilities Access Module Reference*, B035-2425.

1. Open a command prompt.
2. Create a database username and password.
3. Change directory to `INSTALL_DIR/AXSMOD/quickstart`.
4. Update `mqaxsmod_parmfile` accordingly.
5. Change directory to `INSTALL_DIR/AXSMOD/validate`.
6. Run the command `mqaxsmodvalidate.bat` *Tdpid UserName UserPassword*, where the following is true:
   - *Tdpid* is a database ID
   - *UserName* is a database username
   - *UserPassword* is a database user password

   The validation script executes `mqaxsmodqstart.fl` in the `quickstart` directory.

Validating AccessModuleNP

For more information about Teradata Access Module for Named Pipes, see *Teradata® Tools and Utilities Access Module Reference*, B035-2425.

1. Open a command prompt.
2. Create a database username and password.
3. Change directory to `INSTALL_DIRECTORY/AXSMOD/validate`.
4. Run the command `npaxsmodvalidate.bat` *Tdpid UserName UserPassword*, where the following is true:
   - *Tdpid* is a database ID
   - *UserName* is a database username
   - *UserPassword* is a database user password

   The validation script executes `npaxsmodqstart.fl` and `npaxsmodqstart.fe` in the `quickstart` directory.

Validating AccessModuleOLEDB

Prerequisite:

- Verify ODBC driver has been installed.
- In this procedure, either the source or target must be Vantage.

For more information about Teradata Access Module for OLE DB, see *Teradata® Tools and Utilities Access Module Reference*, B035-2425.

1. Open a command prompt.

2. Change directory to `INSTALL_DIR/AXSMOD/validate`.
3. Run the command `oleload.exe`.
   The OLELOAD GUI appears.
4. From **Select a source**, select a source; then, depending on the source selected, make the appropriate selections as described below.

Source	Selections
Vantage	**Teradata host**: Machine name or IP address of the Teradata host **User id**: Username for Vantage **Password**: Password for Vantage
Oracle Database	**Microsoft OLEDB Provider for Oracle (MSDAORA)** **Enter a server name**: System name or IP address **User name**: Username for the Oracle Database **Password**:Password for the Oracle Database
SQL Server	**Microsoft OLEDB Provider for SQL Server (SQLOLEDB)** **Select or enter a server name**: System name or IP address **Enter information to log on to the server**: • **Use Windows NT Integrated Security**: If SQL has been installed on the local host, or • **Use a specific user name and password**: If SQL has been installed on a different machine **Select the database on the server**: Any particular user from SQL server

5. Follow the steps below, depending on source type:

Source	Selections
Vantage	Click **OK**.
Oracle Database or SQL Server	a. Click **Test Connection**. b. Click **OK**.

6. From **Select a destination**, select a target; then, depending on the target selected, make the selections below:

Source	Selections
Teradata host	**Teradata host**: Machine name or IP address of the Teradata host **User name**: Username for Vantage **Password**: Password for Vantage
Oracle Database	**Microsoft OLEDB Provider for Oracle (MSDAORA)** **Enter a server name**: System name or IP address **User name**: Username for the Oracle Database **Password**:Password for the Oracle Database
SQL Server	**Microsoft OLEDB Provider for SQL Server (SQLOLEDB)**

Source	Selections
	Enter information to log on to the server: • **Use Windows NT Integrated Security**: If SQL has been installed on the local host, or • **Use a specific user name and password**: If SQL has been installed on a different machine **Select the database on the server**: Any particular user from SQL server

7. Do the following, depending on target type:

Source	Selections
Vantage	Click **OK**.
Oracle Database or SQL Server	a. Click **Test Connection**. b. Click **OK**.

8. Load the tables:
   a. In the navigator pane of the OLELOAD GUI, select the database associated with the credentials used to log in.
      Under the database selected, the list of tables appears.
   b. Select the desired table, and then in the table pane on the right, select the desired columns.
   c. In the bottom-right corner of the window, click **Settings**.
   d. Make the desired changes, then click **OK**.
   e. Click **Launch.**
      The loaded records are visible in the destination database.

Validating DataConnector API

For more information about the Teradata Data Connector API, see the *Teradata® Tools and Utilities Access Module Reference*, B035-2425.

1. Open a command prompt.
2. Create a database username and password.
3. Change directory to `INSTALL_DIR/piom/validate`.
4. Run the command `piomvalidate.bat` *Tdpid UserName UserPassword*, where the following is true:
   - *Tdpid* is a database ID
   - *UserName* is a database username
   - *UserPassword* is a database user password

   The validation script executes `piomqstart.fl` in the `quickstart` directory.

Validating FastExport

For more information about Teradata FastExport, see *Teradata® FastExport Reference*, B035-2410.

1. Open a command prompt.
2. Create a database username and password.
3. Change directory to `INSTALL_DIR/fexp/sample/validate`.
4. Run the command `fexpvalidate.bat` *Tdpid UserName UserPassword*, where the following is true:
   - *Tdpid* is a database ID
   - *UserName* is a database username
   - *UserPassword* is a database user password

   The validation script executes `fexp_sample.fe` in the `quickstart` directory.

Validating FastLoad

For more information about FastLoad, see the *Teradata® FastLoad Reference*, B035-2411.

1. Open a command prompt.
2. Create a database username and password.
3. Change directory to `INSTALL_DIR/FastLoad/sample/validate.`
4. Run the command `flodvalidate.bat` *Tdpid UserName UserPassword*`,` where the following is true:
   - *Tdpid* is a database ID
   - *UserName* is a database username
   - *UserPassword* is a database user password

   The validation script executes `flod_sample.ml` in the `quickstart` directory.

Validating Geospatial

For more information about Teradata Geospatial, see the *Teradata® Geospatial Utilities User Guide*, B035-2519.

1. Open a command prompt.
2. Create a database username and password.
3. Change directory to the working directory.
4. Run the commands:

   ```
   Set TDgeospatial path
   Set GDAL path
   Set TeradataJDBC path
   SetTDgeospatial jar file path
   ```
5. Change directory to `TDGeospatial_install_directory/tdgeospatial/validate.`

6. Run the command `geospatialvalidate.bat` *Tdpid UserName UserPassword*, where the following is true:
   - *Tdpid* is a database ID
   - *UserName* is a database username
   - *UserPassword* is a database user password

   The validation script executes `geospatialqstart.bteq` in the `quickstart` directory.

Validating MultiLoad

For more information about Teradata MultiLoad, see the *Teradata® MultiLoad Reference*, B035-2409.

1. Open a command prompt.
2. Create a database username and password.
3. Change directory to `INSTALL_DIR/mload/sample/validate`.
4. Run the command `mloadvalidate.bat` *Tdpid UserName UserPassword*, where the following is true:
   - *Tdpid* is a database ID
   - *UserName* is a database username
   - *UserPassword* is a database user password

   The validation script executes `mload_sample.ml` in the `quickstart` directory.

Validating Teradata Java Message Service (JMS)

For more information about Teradata Access Module for JMS, see the *Teradata® Tools and Utilities Access Module Reference*, B035-2425.

1. Open a command prompt.
2. Create a database username and password.
3. Change directory to `INSTALL_DIR/AXSMOD/quickstart`.
4. Update `jmsaxsmod_parmfile` accordingly.
5. Change directory to `INSTALL_DIR/AXSMOD/validate`.
6. Run the command `jmsaxsmodvalidate.bat` *Tdpid UserName UserPassword*, where the following is true:
   - *Tdpid* is a database ID
   - *UserName* is a database username
   - *UserPassword* is a database user password

   The validation script executes `jmsaxsmodqstart.fl` in the `quickstart` directory.

Validating Teradata Parallel Transporter

For more information about Teradata Parallel Transporter, see the *Teradata® Parallel Transporter Reference*, B035-2436.

1. Open a command prompt.
2. Change directory to `TPT_install_directory`.
3. Run the command `tptvalidate.bat`.

Validating Teradata Wallet

For more information about Teradata Wallet, see *Teradata Vantage™ - Advanced SQL Engine Security Administration*, B035-1100.

To see the Teradata Wallet help, enter `help` from the command prompt.

1. Open a command prompt.
2. Change directory to Teradata Wallet install directory.
3. Run the command `validate.bat`.

Validating TPump

For more information about Teradata TPump, see the *Teradata® Parallel Data Pump Reference*, B035-3021.

1. Open a command prompt.
2. Create a database username and password.
3. Change directory to `INSTALL_DIR/tpump/sample/validate`.
4. Run the command `tpumpvalidate.bat` *Tdpid UserName UserPassword*, where the following is true:
   - *Tdpid* is a database ID
   - *UserName* is a database username
   - *UserPassword* is a database user password

   The validation script executes `tpump_sample.tp` in the `quickstart` directory.

Uninstalling Software

About Uninstalling Previous Versions of Teradata Tools and Utilities

If a pre-14.10 version of TTU is installed then please use the uninstall vb script to uninstall before installing 17.10. See Uninstall Teradata Tools and Utilities Pre-14.10 for details.

Any pre-15.10.10 version of TTU must be uninstalled before installing 17.10. See Uninstalling Teradata Tools and Utilities 14.10 and Later for details.

If a 15.10.10 or later release is installed on the system, then 17.10 can also be installed on the same system without uninstalling the 15.10.10 or later release. See Multi-Version Support for details.

Note:

If you need to reinstall 17.10, you must use **Programs and Features** to uninstall the existing product instance. For more information, see Interactively Uninstalling Teradata Tools and Utilities.

Generating a List of Installed Products

1. Start the application that generates a list of products using one of the following methods:

Method	Procedure
From the Start menu	• Select **Start > All Programs > Teradata Client 17.10 > Teradata List Products**. Alternatively, in the <*prefix*>\Teradata\Client\17.10\bin folder, double-click TTUListProducts.exe.
From a command prompt	a. Open a command prompt. b. Change directory to the \Windows\TTU\ folder (or to the appropriate folder on the physical media): `cd Windows\TTU\` c. Run the following command: `TTUListProducts.exe -s -f` *filename* where: • **-s** runs the application in silent mode • **-f** *filename* allows you to specify the name of the output file You can specify **-s** and **-f** *filename* at the same time.

A file named TTUListProducts*timestamp*.txt is created on the Desktop, where *timestamp* is the time the file was created. It contains a list of all of the Teradata Tools and Utilities products that are installed on the system.

Note:

If you used the **-f** *filename* option at the command prompt, the filename matches what you specified.

Uninstalling Teradata Tools and Utilities 14.10 and Later

This is optional if installing 17.10. The 17.10 installation will prompt to remove a 14.10 or through 15.10.09 release before installing the 17.10 release.

You can use a single DOS command to silently uninstall Teradata Tools and Utilities 14.10 and later products.

1. Close all active Windows programs and temporarily disable any virus protection software running on the client computer.
2. Open a command prompt.
3. Change directory to the folder that contains the `silent_uninstall.bat` file.
   The `silent_uninstall.bat` file is specific to a Suite and can only be used with that suite. Each suite has its own silent_uninstall.bat file.
4. Enter the appropriate command to uninstall the products.
   - To uninstall all products:

     ```
     silent_uninstall.bat ALL
     ```

   - To uninstall fewer than all products:

     ```
     silent_uninstall.bat "PRODUCT"
     ```

   For a list of **PRODUCT** parameter values and other details about the command syntax to run the batch file, see Product Parameter Values for Silent Installs.

Uninstall Teradata Tools and Utilities Pre-14.10

Uninstallation Script Prerequisites

Note:

Use the `uninstall_TTU.vbs` script to uninstall pre-14.10 products. To uninstall 14.10 and later products, use **Programs and Features** to interactively uninstall or use `silent_uninstall.bat`.

To run the `uninstall_TTU.vbs` script, the following software is required on the target machine.

- Windows Installer version must be 3.0 or above. To see the version of Windows Installer, enter the `msiexec` command at the command prompt.

- Windows Script Host (WSH) version must be 5.6 or above. To see the version of Windows Script Host, enter the `cscript` command at the command prompt.

Related Information:

Interactively Uninstalling Teradata Tools and Utilities
Uninstalling Teradata Tools and Utilities 14.10 and Later

Uninstallation Script Overview

Note:

It is not necessary to uninstall previous versions of Teradata Wallet before you install another version of it.

Use `uninstall_TTU.vbs` to remove pre-14.10 products. Use the Add/Remove entry for the 14.10 or later Suite or the Suite `silent_uninstall.bat` file to remove the Suite. The script is in the installation package at `\Windows\uninstall_TTU.vbs`.

It creates a log file (`%TEMP%\uninstall_script_`*timestamp*`.log`) that contains a record of all of the steps taken during execution.

`msi` logging is turned on during package installation. The `msi` uninstallation log files are stored in the `%TEMP%` folder with the following file name:

`uninstall_`*package_name_package_version*`.log`

These logs provide useful information to support personnel if any issues are observed in the uninstallation of Teradata Tools and Utilities packages.

Run the uninstallation script before you do any of the following:

- Install a new version of a package
- Install a specific version of a package (a version other than the latest version)
- Decommission Teradata Tools and Utilities software on a particular system

Note:

The version of the following products that use Teradata Wallet and the version of Teradata Wallet must match:

- Teradata Call-Level Interface version 2
- Open Database Connectivity (ODBC) Driver for Teradata

When you run the script, you can specify individual packages to be uninstalled. Teradata recommends that you uninstall the dependency products (CLIv2, PIOM, TDICU, and TDODBC) last; however, they can be uninstalled individually regardless of the order.

NOTICE

If a dependency product is uninstalled without reinstalling a new version, the products that depend on that product stop functioning. For example, if CLIv2 is uninstalled, all products that depend on CLIv2 stop functioning.

In order for the script to run, the command must include the **PRODUCT** parameter. Without it, the script displays usage information. The script supports the following two options:

- **/PRIORTOVERSION**
- **/FORCE**

As an argument to the script, a version can be specified using the **/PRIORTOVERSION** parameter (for example: `/PRIORTOVERSION:14.10.00.00`). The uninstallation script only uninstalls versions of the specified packages that are earlier than the version specified in the **/PRIORTOVERSION** parameter. When the **/PRIORTOVERSION** parameter is not specified, the default value is /PRIORTOVERSION:16.00.00.00.

The default behavior handles uninstallations of Teradata Tools and Utilities packages, except corrupted installations which are handled using the **/FORCE** parameter. When this parameter is specified, the uninstallation script performs an extensive registry search for all versions and components that were installed by a particular Teradata Tools and Utilities package. If the registry entries are found, the script deletes them. This parameter should be used with care because it takes time to search the registry for versions and components of a package. If multiple packages are passed as arguments, the script can take a long time to run.

When both parameters are specified, the `/FORCE:TRUE` parameter performs uninstallation and cleanup of only the versions of the packages that are earlier than those specified using the **/PRIORTOVERSION** parameter. The /FORCE:TRUE should not be used initially. If a product does not get uninstalled without using the FORCE flag, then use it only when a product or list of products are specified.

The script uses system packaging commands to uninstall packages, so it must be executed by an admin user (or by someone with admin privileges) .

Using a Script to Silently Uninstall Teradata Tools and Utilities

1. Run the `uninstall_ttu.vbs` script.

Command	Description
`cscript uninstall_ttu.vbs`	Displays usage information
`cscript uninstall_ttu.vbs` *PRODUCT*	Uninstalls the products you specify
`cscript uninstall_ttu.vbs ALL`	Uninstalls all products

PRODUCT

One or more products that will be uninstalled

The product parameter values must be listed in the same order as shown in the table below. The values are case-sensitive.

Example of uninstalling BTEQ and its dependencies:

```
cscript uninstall_ttu.vbs BTEQ PIOM CLIv2 TDICU
```

Products and their Parameter Values

Product Name	Parameter Value
(All products)	ALL
Teradata Wallet	TDWALLET
Teradata Parallel Transporter Base	TPTBASE
Teradata System Emulation Tool	TSET
Teradata Access Module for JMS	JMSAXSMOD
Teradata Access Module for OLE DB	OLEDBAXSMOD
Teradata Access Module for WebSphere MQ	MQAXSMOD
Teradata Access Module for Named Pipes	NPAXSMOD
Teradata C Preprocessor2	SQLPP
OLE DB Provider for Teradata	TDOLEDB
Teradata Parallel Data Pump	TPUMP
Teradata FastExport	FASTEXP
Teradata MultiLoad	MLOAD
Teradata FastLoad	FASTLD
Basic Teradata Query	BTEQ
Teradata Data Connector API	PIOM
Open Database Connectivity (ODBC) Driver for Teradata	TDODBC
.NET Data Provider for Teradata	TDNETDP
Teradata Call-Level Interface version 2	CLIv2
Teradata Wallet	TDWALLET
Shared ICU Libraries for Teradata	TDICU
Teradata Geospatial Utilities	TDGEOSPATIAL

PRIORTOVERSION

Removes products prior to a specified version

For example, if you are installing BTEQ from Teradata Tools and Utilities 17.00 and you need to uninstall packages prior to 14.10, enter the following command:

```
cscript uninstall_ttu.vbs BTEQ PIOM CLIv2
TDICU /PRIORTOVERSION:14.10.00.00
```

The version parameter *must* contain all four pairs of digits, and can be specific enough to remove all packages before, for example, **`14.00.00.01`**. Typically the last two pair of digits would just be `00.00`.

FORCE

Removes all traces of the products by force

The default is `/FORCE:`**`FALSE`**. If you specify `/FORCE:`**`TRUE`**, the installation process performs a thorough investigation of the registry for all known product keys and removes all traces. The process can take a long time. It is recommended that you use the `/FORCE:`**`TRUE`** option only if the default `/FORCE:`**`FALSE`** option did not successfully uninstall the products that you specified.

For example, to force uninstallation of BTEQ, enter the following command:

```
cscript uninstall_ttu.vbs BTEQ PIOM CLIv2 TDICU /FORCE:TRUE
```

Interactively Uninstalling Teradata Tools and Utilities

1. Go to **Control Panel > Programs and Features**.
2. Uninstall the Teradata Tools and Utilities products.

   When you uninstall the Base suite, Teradata Tools and Utilities - Base 17.10, all of the Teradata Tools and Utilities products are uninstalled.

Supporting Information

Software Packages

The table below contains a list of the Teradata Tools and Utilities products that are in the Base suite. Products that support a specific operating system are noted.

Note:

In the table, the package titles are abbreviated:

Abbreviation	Software Package
Base	*Teradata Tools and Utilities Base Release*, BCD0-2458-0000 (IBM AIX, Linux, Ubuntu, Solaris, Windows, Apple macOS)

Products in Base Software Package

	Package Title (Abbreviated)	
Product	**64-bit**	**32-bit**
.NET Data Provider for Teradata	●	
Basic Teradata Query	●	
Open Database Connectivity (ODBC) Driver for Teradata	●	●
Shared ICU Libraries for Teradata	●	●
Teradata Access Module for JMS	●	
Teradata Access Module for Kafka	●	
Teradata Access Module for Named Pipes	●	
Teradata Access Module for OLE DB	●	
Teradata Access Module for WebSphere MQ	●	
Teradata C Preprocessor2	●	
Teradata Call-Level Interface version 2	●	●
Teradata Data Connector API	●	●
Teradata FastExport	●	
Teradata FastLoad	●	

Product	Package Title (Abbreviated)	
	64-bit	32-bit
Teradata Geospatial Utilities		●
Teradata MultiLoad	●	
Teradata Parallel Data Pump	●	
Teradata Parallel Transporter Base	●	
TPT API		●
Teradata System Emulation Tool		●
Teradata Wallet	●	●
Teradata JDBC Driver	●	●

Related Information:

Products Available from Teradata Community

Product Descriptions

Product	Description
.NET Data Provider for Teradata	Library of routines that Microsoft .NET Framework applications use for uniform and standard-based (ADO.NET) access to data sources such as Vantage. This is the data access component of the .NET Framework Class Library.
Basic Teradata Query	General-purpose utility that allows users on a workstation to communicate with one or more Vantage systems and format reports for output to print and screen. This utility is accessible from the command line or a script.
Open Database Connectivity (ODBC) Driver for Teradata	Enables ODBC applications running on Microsoft Windows, UNIX/Linux, and Apple macOS operating systems to communicate with the database using the open standard ODBC interface.
Shared ICU Libraries for Teradata	Library of routines based on IBM International Components for Unicode. The library provides Unicode support.
Teradata Access Module for JMS	Command line utility that provides a fast, reliable, asynchronous way to transfer data between any JMS-enabled messaging system and Vantage. This utility works with the Teradata Parallel Transporter products as well as Teradata FastExport, Teradata FastLoad, Teradata MultiLoad, and Teradata Parallel Data Pump.
Teradata Access Module for Kafka	Component that allows to export data from Apache Kafka and to import data to Apache Kafka. This utility works with the Teradata Parallel Transporter products.
Teradata Access Module for Named Pipes	Component that provides an interprocess communication link between a writer process (such as Teradata FastExport) and a reader process (such as Teradata FastLoad). This utility works with the Teradata Parallel Transporter products as

Product	Description
	well as Teradata FastExport, Teradata FastLoad, Teradata MultiLoad, and Teradata Parallel Data Pump. It is accessible from the command line.
Teradata Access Module for OLE DB	Command line utility that acts as an interface between Teradata load and export utilities (Teradata FastLoad, Teradata FastExport, Teradata MultiLoad, Teradata Parallel Data Pump, Teradata Parallel Transporter, and Basic Teradata Query) and data sources for which an OLE DB provider is available. The access module quickly moves data between an OLE DB data source and Vantage without requiring intermediate storage.
Teradata Access Module for WebSphere MQ	Component that allows Teradata utilities to import data using IBM WebSphere MQ Access Module message queuing middleware. This utility works with the Teradata Parallel Transporter products as well as Teradata FastLoad, Teradata MultiLoad, and Teradata Parallel Data Pump. It is accessible from a command line.
Teradata C Preprocessor2	Command line utility that provides the programmer of a C-based application a simplified way to read and change information stored in Vantage. This utility interprets and expands SQL statements that have been incorporated in an application program.
Teradata Call-Level Interface version 2	Library of routines that enable an application program to access data in Vantage.
Teradata Data Connector API	Component that reads and writes flat files to provide the input/output interface for access modules and BTEQ. For additional information about Teradata Data Connector API, see the following documents: • *Teradata® Tools and Utilities Access Module Programmer Guide*, B035-2424 • *Teradata® Tools and Utilities Access Module Reference*, B035-2425
Teradata FastExport	Uses multiple sessions to quickly transfer large amounts of data from Vantage tables and views to a client-based application. This utility is accessible from the command line or a script.
Teradata FastLoad	Quickly loads large amounts of data to empty tables in Vantage. This utility uses multiple sessions to load data into only one table in the database per job. This utility is accessible from the command line or a script.
Teradata Geospatial Utilities	Loads Shape file data into tables in Vantage. It uses multiple sessions to load data, but it loads data into one table per layer.
Teradata JDBC Driver	Library of routines that enables Java applications to connect to Vantage.
Teradata MultiLoad	Provides an efficient way to deal with batch maintenance of large databases. Useful for performing fast, high-volume maintenance on multiple Vantage tables and views. This utility is accessible from the command line or a script.
Teradata Parallel Data Pump	Data loading utility that maintains (update, delete, insert, and atomic upsert) data in Vantage. This utility uses standard Teradata SQL to achieve moderate to high data loading rates. Multiple sessions and multi-statement requests are typically used to increase throughput. It is accessible from the command line.

Product	Description
Teradata Parallel Transporter Base	Package containing the script-driven interface (Teradata Parallel Transporter) and the application programming interface (Teradata Parallel Transporter API). Teradata Parallel Transporter is an object-oriented and scalable client application that perform high-speed, parallel data operations (extraction, loading, and updating) on Vantage. **Note:** The following Teradata Parallel Transporter Operators are installed as part of the Teradata Parallel Transporter Base: • DataConnector Operator • DDL Operator • Export Operator • FastExport OUTMOD Adapter Operator • FastLoad INMOD Adapter Operator • Load Operator • MultiLoad INMOD Adapter Operator • ODBC Operator • Operator Support Library • OS Command Operator • Schema Mapping Operator • SQL Inserter Operator • SQL Selector Operator • Teradata Parallel Transporter Stream Operator • Teradata Parallel Transporter API • Teradata Parallel Transporter ICU Library • Teradata Parallel Transporter Wizard • Update Operator
Teradata Studio Express	Formerly known as Teradata SQL Assistant Java Edition, an information discovery tool that retrieves data from Vantage systems and allows the data to be manipulated and stored on the desktop. It is built on top of the Eclipse Rich Client Platform (RCP). This allows the product to take advantage of the RCP framework for building and deploying native GUI applications to a variety of desktop operating systems.
Teradata System Emulation Tool	Windows-based GUI application that emulates Optimizer-generated data from a target system, generates query plans, and then imports that information to a test system where queries can be run without impacting the production system.
Teradata Wallet	Package containing a library and a command line tool (tdwallet) that lets users securely store private information such as Vantage passwords on client computers. The stored information can then be recalled when logging in to Vantage.

Product Dependencies

Product	Teradata Tools and Utilities Dependencies	Third-Party Dependencies
Basic Teradata Query	Shared ICU Libraries for Teradata Teradata Call-Level Interface version 2 Teradata Data Connector API	None
Teradata Access Module for Named Pipes	None	None
.NET Data Provider for Teradata	None	Microsoft .NET Framework Version 4.1 Redistributable Package
Open Database Connectivity (ODBC) Driver for Teradata	None	None
Shared ICU Libraries for Teradata	None	None
Teradata C Preprocessor2	Shared ICU Libraries for Teradata Teradata Call-Level Interface version 2	None
Teradata Call-Level Interface version 2	Shared ICU Libraries for Teradata	None
Teradata Data Connector API	None	None
Teradata FastExport	Shared ICU Libraries for Teradata Teradata Call-Level Interface version 2 Teradata Data Connector API	None
Teradata FastLoad	Shared ICU Libraries for Teradata Teradata Call-Level Interface version 2 Teradata Data Connector API	None
Teradata JDBC Driver	None	JDK 1.4.2 , JDK 5.0, JDK 6.0, JDK 7.0, JDK 8.0 **Note:** • JDK 5.0 is certified for Windows 32-bit. • JDK 5.0 is certified for Windows XP Professional x64 using EM64T. • JDK 6.0, JDK 7.0 and JDK 8.0 are certified for Windows 64-bit.

Product	Teradata Tools and Utilities Dependencies	Third-Party Dependencies
		Note: The environment can be downloaded from Java Downloads.
Teradata Access Module for JMS	None	JRE 1.4.2, JRE 1.5, JRE 1.6, JRE 1.7, JRE 1.8
Teradata MultiLoad	Shared ICU Libraries for Teradata Teradata Call-Level Interface version 2 Teradata Data Connector API	None
Teradata Access Module for OLE DB	Open Database Connectivity (ODBC) Driver for Teradata Shared ICU Libraries for Teradata	None
Teradata Access Module for WebSphere MQ	Teradata Data Connector API	IBM WebSphere MQ 7.5 **Note:** To display the version of WebSphere that is installed on the system, type `dspmqver` at the command line.
Teradata Parallel Data Pump	Shared ICU Libraries for Teradata Teradata Call-Level Interface version 2 Teradata Data Connector API	None
Teradata Parallel Transporter Base	Shared ICU Libraries for Teradata Teradata Call-Level Interface version 2	32-bit, Java 2 Runtime Environment 1.4.2_06 **Note:** The environment can be downloaded from Java Downloads.
Teradata Studio Express Teradata Studio Teradata Plug-in for Eclipse	None	Java Runtime Environment (JRE) 1.8 **Note:** JRE can be downloaded from Java Downloads.
Teradata System Emulation Tool	Shared ICU Libraries for Teradata Teradata Call-Level Interface version 2 or Open Database Connectivity (ODBC) Driver for Teradata	None
Teradata Wallet	None	None

Product	Teradata Tools and Utilities Dependencies	Third-Party Dependencies
	For Teradata Wallet help, at the command prompt enter `help` . For more information about Teradata Wallet, see *Teradata Vantage™ - Advanced SQL Engine Security Administration*, B035-1100.	

Dependent Products

During installation of Teradata Tools and Utilities, the following dependent products are automatically installed:

- Shared ICU Libraries for Teradata
- Teradata Call-Level Interface version 2
- Teradata Data Connector API
- Open Database Connectivity (ODBC) Driver for Teradata

Product Parameter Values for Silent Installs

Product Parameter Values for Silent Installs

You can use batch files to silently uninstall and install Teradata Tools and Utilities products. The command to run the batch file must include a value for the **PRODUCT** parameter.

Product Parameter Values

Name of Product Suite	Name of Product	Parameter Value
Uninstalling only	(All products)	ALL (no quotes)
Base	.NET Data Provider for Teradata	TDNETDP
	Basic Teradata Query	BTEQ
	Teradata Access Module for JMS	JMSAM
	Teradata FastExport	FastExport
	Teradata FastLoad	FastLoad
	Teradata JDBC Driver	TeraJDBC
	Teradata Access Module for Kafka	KAFKAAM
	Teradata Access Module for Azure Blob	AZUREAM
	Teradata Access Module for S3	S3AM
	Teradata MultiLoad	MultiLoad
	Teradata Access Module for Named Pipes	NamedPipesAM
	Teradata Access Module for OLE DB	OLEDBAM
	Teradata C Preprocessor2	Preprocessor
	Teradata Parallel Data Pump	TPump
	Teradata Parallel Transporter Base	TPTBase
	Teradata Wallet	TDWallet
	Teradata Access Module for WebSphere MQ	WebsphereAM
	Teradata Geospatial Utilities	TDGeoSpatial
	Teradata System Emulation Tool	TSET

Batch Install/Uninstall Text Strings

Name of Product Suite	Text String (set of parameter values)
Base	BTEQ,FastExport,FastLoad,JMSAM,KAFKAAM,MultiLoad,NamedPipesAM, OLEDBAM,Preprocessor,TDNETDP,TDWallet,TPTBase,TPump,WebsphereAM, TDGeoSpatial,TeraJDBC,TSET

The command syntax for running a batch file is as follows:

batch_file_name **"PRODUCT"**

where *batch_file_name* is the name of the batch file:

- `silent_uninstall.bat`
- `silent_install.bat`

and where **PRODUCT** is a set of parameter values associated with the suite of products you are uninstalling.

The parameter values are case-sensitive. Be sure to include the quote marks ("") which must be formatted as straight quotes. Teradata recommends that you generate install scripts using a text editor to ensure straight quotes are used. The installation will fail otherwise. Separate the parameter values with a comma, but do not include a space.

Note:

When you run a batch file, the ODBC driver and all of the dependent products are installed or uninstalled automatically. You do not have to specify their parameter values. These products include the following:

- Open Database Connectivity (ODBC) Driver for Teradata
- Shared ICU Libraries for Teradata
- Teradata Call-Level Interface version 2
- Teradata Data Connector API

Installation Methods

Installation Method	Step
Installation media	Insert the appropriate media into the drive. **Note:** If the **Welcome** dialog box does not appear shortly after the media is inserted, use Windows Explorer to navigate to the main folder of the media. Double-click `setup.bat` to start the installation process.
Downloaded package	1. Unzip the file of the package you're ready to install (for example, BCD0-2374-0000). 2. Select one of the following methods: a. Go to the main folder (for example, `Database_Mgmt`) and double-click `setup.bat`. b. To create an installation log file, perform the following steps: • Be sure the folder where the `setup.exe` file is located is writable (for example, `Database_Mgmt\Windows\TTU`. • From a command prompt, change directory to the folder where the `setup.exe` file is located. • Run either of the following commands: ◦ `setup.exe/debuglog` A log file named `InstallShield.log` will be created in the folder where the `setup.exe` file is located. ◦ `setup.exe /debuglog"`*path*`\`*filename*`"` where *path* is the path (absolute or relative) where you want the log file to be created and *filename* is the name of the log file. Be sure to add the quote marks ("").

Customizing a Suite

1. Do the following to create a Suite that contains a subset of the primary products in a Suite:
   `TTU\create_subset_suite.bat suitelist=`*<comma separated list of primary products>* *<location of new Suite>*

   where the following is true:

   - *<location of new Suite>* is a fully qualified directory, e.g., "`C:\temp`"
   - *<comma-separated list of primary products>* is one or more of the following:
     - Base: See the Base row in Product Parameter Values for Silent Installs for a list of the primary products in the Base Suite.

Note:

The Suite feature UI screen will only list the products in the list provided. The dependent products (tdicu, piom and cliv2) are always installed and not listed on the features screen.

The `create_subset_suite.bat` is specific to a Suite, so please use the file in the Suite TTU directory that you would like to customize. The custom suite can also be installed/uninstalled silently. The silent install/uninstall bat files are specific to a suite.

Troubleshooting the Teradata Tools and Utilities Installation

Error: ActiveX component can't create object: 'Scripting.FileSystemObject'

Source

Microsoft VBScript runtime error.

Resolution

1. Check the following registry key:

   `HKEY_CLASSES_ROOT\CLSID\{0D43FE01-F093-11CF-8940-00A0C9054228}`

   The InprocServer32, ProgId, TypeLib and Version keys must be present. If those keys are not present, then do the following:
2. Grant Administrators and System full control to the above key.
3. Re-register the scrrun.dll manually in an elevated command prompt (regsvr32 scrrun.dll).
4. Check if the sub-keys were created.

   If the keys are created, then try the install again.

Error: 2738 could not access vbscript runtime for customer action

Consider the following options to resolve the error.

Check for vbscript and jscript DLLs in the Registry

1. Check that vbscript.dll and jscript.dll are not registered in HKEY_CURRENT_USER (HKCU), checking for the registry keys below.
   - `VBScript, HKCU\SOFTWARE\Classes\CLSID\{ B54F3741-5B07-11CF-A4B0-00AA004A55E8}`
   - `JScript, HKCU\SOFTWARE\Classes\CLSID\{ F414C260-6AC0-11CF-B6D1-00AA00BBBB58}`
2. Remove these keys if they exist in HKEY_CURRENT_USER.

Re-register vbscript or jscript DLLs

Make sure that if you need to re-register vbscript.dll or jscript.dll, you run regsvr32.exe in an elevated console with User Account Control (UAC) enabled. Otherwise, you will end up registering the runtimes in HKCU.

Check for the following keys:

- VBScript, HKEY_CLASSES_ROOT \CLSID\{ B54F3741-5B07-11CF-A4B0-00AA004A55E8}
- JScript, HKEY_CLASSES_ROOT \CLSID\{ F414C260-6AC0-11CF-B6D1-00AA00BBBB58}

Installing .NET Data Provider for Teradata

About Installing .NET Data Provider for Teradata

Installing .NET Data Provider for Teradata involves several tasks:

1. Review security requirements for using Microsoft .NET Framework.
2. Check that the required version of Microsoft .NET Framework is installed on your computer.
3. Install .NET Data Provider for Teradata from the Base suite:

   Teradata Tools and Utilities Release 17.00.11, BCD0-2458-0000 (Windows, Linux, Amazon Linux 2, Ubuntu, IBM AIX, Solaris, Apple macOS)
4. Verify the installation.

The .NET Data Provider for Teradata has no dependencies on other Teradata Tools and Utilities client products.

About Security Requirements for .NET Data Provider for Teradata

The Microsoft Developer Network Library provides extensive documentation for Microsoft .NET Framework. The following topics are recommended reading before you start developing applications to access Vantage with Microsoft .NET Framework, Microsoft Visual Studio, and .NET Data Provider for Teradata:

- ADO.NET Specification and .NET Data Providers at ADO.NET Overview
- FullTrust permission set at Security in the .NET Framework
- Additional security information at Administering Security Policy

Uninstalling .NET Data Provider for Teradata

Uninstall previous versions of .NET Data Provider for Teradata before installing the most recent version.

1. Go to **Control Panel > Programs and Features**.
2. If .NET Data Provider for Teradata appears in the list of installed programs and is older than the version you are installing, right-click the entry and select **Uninstall** or **Change/Remove**, depending on the Windows version.
3. In the **.NET Data Provider for Teradata - InstallShield Wizard** dialog box, click **Remove** then **Next**.
4. Click **Finish**.

Installing Third-Party Microsoft .NET Framework

Microsoft .NET Framework must be installed before installing .NET Data Provider for Teradata. See Determining the Current Version of .NET Framework for more details.

Downloading .NET Data Provider for Teradata

Prerequisite:

The .NET Data Provider for Teradata (tdnetdp) is available in the Base Suite. If installing only the tdnetdp product, then the individual product can be obtained from Teradata Community.

You must be a registered user of Teradata Community before you can download installation files for .NET Data Provider for Teradata.

The executable file for .NET Data Provider for Teradata, along with a README in text format and a `.zip` archive of help files in `.chm` format, are available for download from Teradata Community.

1. Go to .NET Data Provider for Teradata download files.
2. Log on to Teradata Community.
3. Click the link to download the executable file `tdnetdp__windows_indep.`*version*`.exe`.
4. Accept the license agreement.
5. [Optional] Click the link to download a `.zip` archive of the help files.
6. Accept the license agreement.
7. [Optional] Click the link to view the README in the browser.
   The README provides detailed information about this version of .NET Data Provider for Teradata as well as information on security requirements, supported platforms, development environments, and versions of Vantage.

Installing .NET Data Provider for Teradata

1. Browse to the installation file `tdnetdp__windows_indep.`*version*`.exe` that you downloaded.
2. Double-click the executable to launch the installer.
3. If you are prompted to allow the program to make changes to the computer, click **Yes**.
4. In the **Choose Setup Language** dialog box, select the desired language, then click **Next** .

   Note:

   English and Japanese are the only languages currently supported.

5. In the **Welcome** dialog box, click **Next** .
6. Accept the default **Destination Folder** or select a different destination folder and click **Next**.

- On 32-bit systems, the default installation folder is `%PROGRAMFILES%\Teradata\NET Data Provider for Teradata\`*version* `\`.
- On 64-bit systems, the folder is `%PROGRAMFILES(x86)%\Teradata\NET Data Provider for Teradata\`*version* `\`.

7. Select one of the following in the **Setup Type** dialog box:
   - **Complete** to install all program files for .NET Data Provider for Teradata.
   - **Custom** to install all program files for .NET Data Provider for Teradata and include the following optional features:

Feature	Description	Notes
Microsoft Visual Studio 2010 Integration	If Microsoft Visual Studio 2010 is installed on the system, .NET Data Provider for Teradata is integrated into Microsoft Visual Studio.	Integration with Microsoft Visual Studio products is selected by default if the installer detects that they are on the system. You can clear the option to integrate .NET Data Provider for Teradata with Microsoft Visual Studio if integration is not required. When integration is not selected, the installation time is shortened.
Microsoft Visual Studio 2008 Integration	If Microsoft Visual Studio 2008 is installed on the system, .NET Data Provider for Teradata is integrated into Microsoft Visual Studio. If Microsoft SQL Server Business Intelligence Development Studio is also installed on the system, .NET Data Provider for Teradata is also integrated into the Development Studio.	
Publisher Policy	A publisher policy assembly configures the policy that is used when the Microsoft .NET Framework runtime binds to an assembly. The publisher policy assembly is installed into the Global Assembly Cache.	The **Publisher Policy** feature is selected by default. ◦ Leave **Publisher Policy** selected if you are upgrading a previous version of .NET Data Provider for Teradata or other applications are installed that are dependent on an earlier version of .NET Data Provider for Teradata. ◦ Clear **Publisher Policy** if no previous installations of .NET Data Provider for Teradata exist on the system and no other applications are dependent on an earlier version of .NET Data Provider for Teradata. Suppose you upgrade from version 01.02.00 to version 15.10.00 of .NET Data Provider for Teradata and you bind all existing applications to the 15.10.00 version. You must place the publisher policy assembly `policy.1.2.Teradata.Client.Provider.dll` in the Global Assembly Cache to redirect (bind) an existing application built with the 01.02.00 version of .NET Data Provider for Teradata to the 15.10.00 version.

8. Click **Install**.
9. [Optional] To display the readme file, select **Show the readme file**.
10. Click **Finish**.
    To verify the installation, see Verifying Installation of the .NET Data Provider for Teradata.

About Teradata Community

Some Teradata Tools and Utilities products can be downloaded from the Teradata Community Downloads page.

Note:

These products are intended to be used to connect to Vantage using non-Teradata products, not to update the Teradata Tools and Utilities suite of products.

Products Available from Teradata Community

Note:

This list is limited to Teradata Tools and Utilities applications.

Connectivity

- Teradata Call-Level Interface version 2 suite (includes Teradata International Components for Unicode)
- Teradata JDBC Driver
- .NET Data Provider for Teradata
- Open Database Connectivity (ODBC) Driver for Teradata suite (includes Teradata International Components for Unicode)

Tools

- Teradata Plug-in for Eclipse
- Teradata Studio Express
- Teradata Wallet

Downloading a Product from the Teradata Community

1. Go to Teradata Community Downloads.
2. Click the product name or the platform next to the product name.
3. Click the software package or text file to download.
4. If prompted, enter a username and password or create a username and password.
5. Agree to the **License Agreement**.
6. If a readme file is included with the download, review the file and print any special download or installation instructions before continuing.
7. Return to the **Downloads** page and click the package to download.

8. Click **Save** and select a location where to save the file.

Additional Information

Audience

This document is intended for use by:

- Database administrators and relational database developers
- System administrators
- System and application programmers
- System operators and other database specialists

Changes and Additions

For a complete list of changes to the TTU Installer, see the *Teradata® Tools and Utilities Release Definition* (B035-2029) associated with this release.

Additional Information

If you need additional information or assistance, contact your Customer Service Representative. If you purchased an Enterprise System Support (ESS) contract, a customized support plan was created for you. For contact information, refer to your support plan.

Related Documents

Title	Publication ID
Teradata® Tools and Utilities Release Definition	B035-2029

Teradata Links

Link	Description
https://docs.teradata.com/	Search Teradata Documentation, customize content to your needs, and download PDFs. Customers: Log in to access Orange Books.
https://support.teradata.com	Helpful resources in one place: • Support requests • Account management and software downloads • Knowledge base, community, and support policies • Product documentation • Learning resources, including Teradata University

Link	Description
https://www.teradata.com/University/Overview	Teradata education network
https://support.teradata.com/community	Link to Teradata community