

# **Teradata Vantage™ - Advanced SQL Engine Node Software**

## **Migration Guide**

---

Release 17.10

July 2021

# Copyright and Trademarks

Copyright © 2016 - 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
	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.
	Indicates a hazardous situation which, if not avoided, could result in minor or moderate personal injury.
	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](mailto: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

<b>Chapter 1: Introduction to Node Software Migration Guide</b>	<b>5</b>
Changes and Additions	5
<b>Chapter 2: Migration Planning</b>	<b>6</b>
Migration Paths	6
About Platform Requirements	7
Software Requirements	7
Plan Additional Space on Destination System	9
Restoring Data to a Smaller System	10
Delete Crashdumps	11
Join and Hash Indexes	11
Rewrite UDFs, UDTs, Stored Procedures and External Stored Procedures	11
Database Query Logging Data Setup and Save	12
Security Configurations	12
Security Hardening	12
Internet Connection Requirement	13
Migration Timetable	13
<b>Chapter 3: Preparing Source Data for Migration</b>	<b>15</b>
Running the Pre-Migration Preparation Script	15
Inspecting Pre-Migration Preparation Script Output	15
Saving Security Configuration	19
About SCANDISK and CheckTable	19
Checking for Pending Operations	20
<b>Chapter 4: Archiving Source Data</b>	<b>21</b>
Basics of Archiving Data	21
Using DSA to Archive Data	22
<b>Chapter 5: Preparing the Destination System</b>	<b>26</b>
Initializing the Destination System	26
Installing Site Security Configuration	26
The tdlocaledef Utility	27
Recording DBC.Hosts Information	27
Copying Site-Defined Client Character Sets	28
<b>Chapter 6: Restoring Data</b>	<b>29</b>
Basics of Data Restoration	29
Time Zone String Management	29

Logically Deleted Rows .....	31
Data Restoration Using DSA .....	32
<b>Chapter 7: Setting Up Advanced SQL Engine .....</b>	<b>44</b>
About Modifying Security Settings .....	44
Inspecting Stored Procedures .....	44
Inspecting Java External Stored Procedures .....	44
Inspecting UDFs and UDTs .....	45
Verifying DBC.Hosts Configuration .....	45
Verify the Integrity of Database File System and Tables .....	45
Resetting the DBC Password .....	46
Enabling Logons .....	46
<b>Appendix A: Additional Information .....</b>	<b>47</b>

# Introduction to Node Software Migration Guide

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.

The instructions in this book assume the destination system has been properly set up and Advanced SQL Engine is tested and running properly. The instructions also assume that the TTU packages are at the correct release level on the destination system and that the DSC server is functioning correctly. For installing details, refer to the applicable installation documents for the particular platform and operating system. See *Teradata Vantage™ - Advanced SQL Engine Release Definition*, B035-1725 for a complete listing of compatible platforms and disk arrays.

This book describes the procedure to migrate a full Teradata system to another using Teradata Data Stream Architecture (DSA). Steps, commands, and configuration file (XML) examples are provided for an actual full system migration. However, there are many details and options that can be specified by the user that are not covered in this book. See *Teradata® DSA User Guide*, B035-3150 for more details.

Software Package	Minimum Version
DSA	17.02 (Must match or exceed the Advanced SQL Engine version.)

## Changes and Additions

Date	Description
July 2021	The DSC repository now uses Postgres instead of the Teradata database.
June 2020	ARC is no longer supported for system migrations. Material on Data Stream Architecture (DSA) has been enhanced.

# Migration Planning

## Migration Paths

When contracted, all migration and upgrade services are performed by the Teradata Services team.

This document focuses only on migrating an entire system, including access rights. To perform a full system migration, two archive operations must be performed, DBC All followed by DBC Only. The restores must be in the sequence of DBC Only, SYSLIB, DBC All, TD\_SERVER.

Migrations are only supported from two major releases back. The minimum version for migrating directly to Advanced SQL Engine 17.10 is Advanced SQL Engine 15.0. Migration from releases prior to 15.0 require an intermediate migration to a supported release.

---

**Note:**

For more details on Full System migrations or information on performing a partial data migration involving some or all user data without database DBC, see *Teradata® DSA User Guide*, B035-3150.

---

## Supported Migrations and Operating Systems

Advanced SQL Engine release 17.10 can run on the following operating systems:

- Novell SUSE LINUX Enterprise Server 11 Service Pack 3
- Novell SUSE LINUX Enterprise Server 12 Service Pack 3

Advanced SQL Engine supports migrations from Advanced SQL Engine 15.0 and later releases, to release 17.10.

Advanced SQL Engine supports direct migration between releases when the source release is no more than two major releases earlier than the destination release. Otherwise, you must perform an intermediate migration that meets this criterion.

---

**Note:**

Advanced SQL Engine does not support backwards migrations to an earlier major or minor release.

---

**NOTICE**

This document focuses only on direct migration from an Advanced SQL Engine release no more than two major releases earlier than the release to which this document applies. If you need to perform an intermediate migration before migrating to this release, additional actions may be necessary to ensure a successful migration. In such scenarios, be sure to review the appropriate version of the *Teradata Vantage™ - Advanced SQL Engine Release Definition* for the intermediate release in addition to this document before undertaking a migration.

## Considerations for Advanced SQL Engine Migration on Amazon Web Services (AWS) or Microsoft Azure (Azure)

Use the Data Stream Controller (DSC) feature of Teradata Data Stream Architecture for migrations on AWS or Azure. See *Teradata® DSA User Guide*, B035-3150 for more information, especially the section “Using Teradata DSC in the Public Cloud.”

## Advanced SQL Engine Migration in the Public Cloud

Teradata offers a complete line of services to help you use Teradata products in the cloud. These include help with database migrations and upgrades, tailored to your specific needs. For more information, see <https://www.teradata.com/Consulting/Velocity>.

For additional information about Teradata migration services, contact your Teradata representative.

## About Platform Requirements

The instructions in this book assume the destination system has been properly set up and Advanced SQL Engine is tested and running properly. For installing details, refer to the applicable installation documents for the particular platform and operating system. Refer to the *Teradata Vantage™ - Advanced SQL Engine Release Definition*, B035-1725 for a complete listing of compatible platforms and disk arrays.

## Software Requirements

### Teradata Tools and Utilities

Teradata Tools and Utilities (TTU) software permits communication between a Teradata Client workstation and Advanced SQL Engine, and includes many important utilities. Many of the Teradata Tools and Utilities products are installed on all Client platforms.

Each new major release of Advanced SQL Engine will operate properly with the previous major release of TTU. Upgrades can be phased in as needed after the migration. However, Teradata recommends that you use the release of TTU utilities that matches the target Advanced SQL Engine release. If you use a prior

release of TTU, you may not have access to all of the security and other features available in the current Advanced SQL Engine release.

The following TTU utilities must be installed on the target system and must match the new release level:

- BTEQ
- CLI
- TDICU
- TDGSS
- PIOM

## Backup and Restore Network Software

### Hardware Compression Driver Requirement

When restoring an archive made on a source system with hardware block-level compression, you must install the hardware compression driver package (teradata-expressdx) on all TPA nodes of the destination system. The installation requires a reboot of these nodes. Because it allows the destination system to read the compressed archive, this requirement applies even if the destination system is not set up for hardware compression.

---

#### Note:

The teradata-expressdx driver package is provided with systems that are equipped with compression hardware.

---

## Teradata DSA

Teradata DSA allows you to archive and restore data using Teradata Viewpoint portlets or the command line.

### DSA Configuration

Before using DSA for data backup and restore, you must install and configure DSA. Teradata recommends using Viewpoint portlets for configuration, but command-line instructions are also provided. Use the Viewpoint Monitored Systems portlet to make systems available to the BAR Setup portlet. In BAR Setup you must configure the systems, nodes, backup solutions, and target groups before you can use BAR Operations for backup and restore.

See the following documents:

- *Teradata® Viewpoint User Guide*, B035-2206
- *Teradata® DSA User Guide*, B035-3150
- *Teradata® DSA - DSE for Veritas NetBackup Installation, Configuration, and Upgrade Guide*, B035-3151



- *Teradata® DSA - DSU Installation, Configuration, and Upgrade Guide*, B035-3153
- *Teradata® DSA - DSE for IBM Spectrum Protect Installation, Configuration, and Upgrade Guide*, B035-3155

## Plan Additional Space on Destination System

Teradata recommends you have at least 20% free permspace available on the destination system. Additional permspace on the destination system may be required for the following reasons:

- Disabling BLC can cause space requirements to grow nearly 250% in extreme cases.
- Changing a table from NO FALLBACK to ALWAYS FALLBACK will increase the space requirements by 100% for that table.
- New features and functionality add fields to existing system tables or may require new system tables.
- Table headers may increase in size.
- The new system may be configured with more AMPs. A copy of every table header is on every AMP.
- Hash function changes may result in data rows moving to different AMPs causing an out of space condition that did not exist previously.

If the migration involves a hash function change, there must be enough free space in each database to hold an extra temporary copy of the largest table in that database. A hash function change may result in many rows being located on different AMPs on the destination system. When a row is redistributed to another AMP, the row is copied into a buffer to be sent to the new AMP. The space for the original copy of the row is not freed until the entire table has been restored and redistributed.

- The source and destination systems have different DBS Control settings. Some DBS Control settings, especially those related to block-level compression, affect space usage.

It is not only large databases that can run out of space. Use the following query on the source system to show the used and available space for each database:

```
SELECT DatabaseName,SUM(CurrentPerm), SUM(MaxPerm)
FROM DiskSpace GROUP BY 1
WITH SUM(CurrentPerm),SUM(maxperm) ORDER BY 1;
```

If  $\text{SUM(CurrentPerm)/SUM(MaxPerm)} > 80\%$ , the available free space on the system is below the recommended minimum threshold for migration.

In any restore it is possible, due to the issues listed above, that the physical space (Current Perm) used by a user/database will exceed the allocated Max Perm for that user/database after the restore. If there is adequate physical storage available, DSA ignores space accounting during the restore operation. This allows the restore to complete successfully. A non-fatal 1196 will be reported:

```
Database USRP_32448 has exceeded it's logical space limit
```

but the restore will succeed. A system administrator must adjust the Max Perm for those users/databases before the system is turned over to end users or certain queries will abort with out-of-space errors.

## Fallback Considerations

After sysinit is run on the destination system, it will default to ALWAYS FALLBACK. If the source system contains non-fallback tables and the destination system requires that all tables have fallback, the destination system will need to have sufficient space to hold a fallback copy of all of these tables. The max permspace on the destination system should at least double the permspace amount consumed by non-fallback tables on the source system.

Use the following query to calculate total permspace consumed by non-fallback tables on the source system.

```
SELECT SUM(DataBaseSpace.CurrentPermSpace)
(NAMED CurrentPerm, FORMAT
'-,---,---,---,---,---,---,--9')

FROM DBC.Dbase, DBC.DataBaseSpace, DBC.TVM
WHERE DataBaseSpace.TableID <> '000000000000'XB
AND DataBaseSpace.TableID = TVM.tvmid
AND TVM.DatabaseId = Dbase.DatabaseId
AND TVM.TableKind NOT IN ('G','M','V')
AND TVM.ProtectionType = 'N';
```

The destination system will need to have at least this much more space than the source system for these tables.

## Block-Level Compression Considerations

If the source and destination systems have different DBS Control settings relating to block-level compression (BLC), and the migration job is not set up to overwrite BLC DBS Control settings when data is restored to the destination system, the permspace consumed by tables will be different on the two systems.

An extreme case would be if the source system uses BLC to compress all tables, but the destination system does not compress any. In this case, the same set of tables, uncompressed, may consume up to and even more than 250% of the permspace on the destination system that they consumed on the source system. (This is based on Teradata Engineering estimates that the typical reduction in space from using BLC is about 60% per table. Tables compress to, on the average, about 2.5 times smaller than when they are not BLC-compressed.) The reduction rate varies, depending on user data.

## Restoring Data to a Smaller System

Most full-system migrations involve destination systems with sufficient space for the operation. In rare exceptions, or when a disaster-recovery system is not the same configuration as the production system, the destination system may be smaller than the source system. In such cases, special steps are required for a successful restore. Teradata recommends that you contact Teradata Services to ensure a successful migration.

In many cases of migrating to a smaller system, the Max Perm allocations on the destination system will have to be reduced by a system administrator after a successful restore. DSA will allocate to each user/database the same Max Perm that was allocated on the source system before the backup. This means the total of the Max Perm allocations from the larger system may cause the Max Perm allocated to DBC on the smaller system to become negative. In that case, immediately following a successful restore to a smaller system, a system administrator must reduce the Max Perm allocations of each user/database to more closely match the actual Current Perm used by each user/database. Until the DBC Max Perm allocation is positive, the system will be virtually unusable.

If the source system has more physical data (Current Perm) than can fit on the target system, you must exclude some databases from the restore operation until the amount of data being restored is less than the physical disk space available on the target system.

It is more common that a restored user/database exceeds its allocated Max Perm. This is the case when the physical disk space on the target system is actually sufficient to hold all of the existing data from the source system, but the amount of allocated space (Max Perm) on the source system is larger than the physical disk space on the target system. This situation causes the Max Perm for DBC on the target system to become negative. When that happens, any SQL statement that requires any additional space fails, and reports an out of space error.

## Delete Crashdumps

Unless migrating from one Linux system to another running the same Advanced SQL Engine release, the crashdump files from the source system are neither compatible with nor relevant to the destination system. They should be deleted from the source system before creating the archive. As a best practice, always exclude the Crashdumps database when archiving and restoring data as part of the migration process.

## Join and Hash Indexes

DSA manages to save and rebuild all join and hash indexes as part of the archive and restore operations.

DSA performs join and hash index maintenance during the post-script phase of a restoration job, and statistics maintenance during the dictionary phase. As long as the restoration reaches the data phase of restoration, statistics information is not lost if the job subsequently fails.

## Rewrite UDFs, UDTs, Stored Procedures and External Stored Procedures

User-Defined Functions (UDFs), User-Defined Types (UDTs), Stored Procedures and External Stored Procedures may execute code that is platform-dependent. Such objects may not recompile or execute correctly when migrated to another platform. UDTs provided by Teradata are platform-independent.

During the Restore operation, UDFs, UDTs, Stored Procedures and External Stored Procedures are recompiled automatically. The Restore operation will complete even if there are compilation failures.

However, depending on the failed objects some programs, objects and possibly entire databases may be unusable until the failed objects can be recompiled successfully.

## Database Query Logging Data Setup and Save

Database Query Logging (DBQL) data is not archived or restored, with the exception of the DBQLRuleTbl and DBQLRuleCountTbl which are archived/restored as part of database DBC.

If you want to migrate the DBQL logging data to the new system the data must be copied from the DBQL tables to user tables prior to the archive/restore.

Teradata suggests you take advantage of the Teradata Professional Services offering to set up standard DBQL rules and logging on your new system.

## Security Configurations

If you have created a custom security configuration for the source system by editing the default TdgssUserConfigFile.xml file, the configuration is not automatically migrated to the new destination system. After migrating data to the freshly installed Advanced SQL Engine on the destination system, only the new default security configuration is available. To transfer a custom security configuration to the new destination system, you must complete the following steps:

1. Save the edited TdgssUserConfigFile.xml file from the source system. This file is located in the following directory on the Control Node: /opt/teradata/tdat/tdgss/site
2. Replace the TdgssUserConfigFile.xml file on the destination system with the one saved from the source system.
3. Activate the custom configuration on the destination system.

Row-level security is part of the Mandatory Access Control (MAC) feature. If row level security is applied in the source system, you must restore database SYSLIB before you restore any user data on the target system. You must also restore SYSLIB a second time after the migration of the user data is complete.

---

### Note:

Prior to copying row level security tables, re-associate each copied constraint with its respective constraint function.

---

After the migration, refer to *Teradata Vantage™ - Advanced SQL Engine Security Administration*, B035-1100, to plan and implement your new security features.

### Related Information:

[Saving Security Configuration](#)

[Installing Site Security Configuration](#)

## Security Hardening

Security hardening restricts access to certain ports on the Advanced SQL Engine node, which can cause problems when trying to perform a migration to the system.

The message `BYPASS_AR_MONITOR CNS connect timeout` indicates a failure to connect to the CNS subsystem on the cnshost control node.

If this occurs during a migration, it is likely that access to port 3015 is restricted. To correct the problem, reopen port 3015 because it is needed to perform the migration.

For more information, see *Teradata Vantage™ on AWS (DIY) Installation and Administration Guide*, B035-2800.

## Internet Connection Requirement

You need to use an Internet connection during the migration to access online sources of script files and documentation. The connection can be on one of the system nodes, or it can be on a separate computer system. If you use a separate computer system, you will need to download files to the separate computer system temporarily. You must then transfer all downloaded files to the Control Node on the source system.

## Migration Timetable

Planning and performing a migration is a complex process requiring many activities to make sure that all the required software and information has been gathered.

All migrations are performed under the guidance of Teradata Customer Services staff, who work with you to develop a plan for system migration. The key to a successful migration is preparation.

Prior to the scheduled migration complete the following steps:

- Download all necessary documentation.
- Install latest PUTTools package on system.
- Run the Pre-Migration Preparation script to identify and fix any problems. See [Running the Pre-Migration Preparation Script](#).
- Prepare the destination system if different from the source system.
- Install the desired Advanced SQL Engine release and the latest Teradata Tools and Utilities on the destination system.

The time required to complete a migration will vary between systems. The most time consuming step is the actual transfer of all of the data from the source to the target system. The migration will take at least as long as it takes to perform a full system backup. Using DSA to restore data to a target system that has a different configuration than the source system requires significantly more time than was required to create the archive. This is because the restored data must be written one row at a time, while the archive was created using block-level I/O operations.

If you are changing the row format from packed to aligned, or from aligned to packed, it will require at least twice as long as a full system backup, because every data row must be rewritten, field-by-field, one row at a time.

After all of the data has been migrated to the target system the remaining steps and system validation will also vary, depending on the size of the system and the types of data objects.

# Preparing Source Data for Migration

## Running the Pre-Migration Preparation Script

### Prerequisite:

Download and install the latest PUTTools package on all TPA nodes.

DBC must have full access rights for DBC prior to archiving/restoring DBC. If DBC does not have full access rights for DBC, those rights need to be granted prior to performing the archive. If the source system is not available, when DBC has been restored, you must explicitly grant full access rights for each object, view, and macro under DBC before running the `post_dbc_restore` script and DIP.

Run `pre_migration_prep.pl` on the *source* system. The script performs a variety of checks for potential problems that must be resolved prior to continuing the migration process.

Each Advanced SQL Engine release has a unique subdirectory in the PUTTools directory for the pre-migration script applicable to that release. The folder is `/opt/teradata/PUTTools/td17.version/preupgrade/pre_migration_prep.pl`.

1. Create a temporary directory and change directory (`cd`) to that directory.
2. In the temporary directory, run the perl command to start the script:  

```
perl /opt/teradata/PUTTools/td17.version/preupgrade/pre_migration_prep.pl
system_name/dbc,dbc_password output_filename
```

The script generates output report files to the temporary directory.

## Inspecting Pre-Migration Preparation Script Output

The script tests for several types of issues that can prevent or halt the migration. The script outputs results for each type of check to the standard output. As the script runs, results for each of the individual checks are preceded by an output line such as `**** Checking for Reserved Words ....`

The script saves the results of each type of check to a separate output file in the directory where the script is located. Inspect these files and resolve any reported problems before continuing.

There are checks for each of the following types of items:

Item Type	What the Script Checks	What You Must Do
Reserved Words	Use of any reserved words in the user-defined tables and stored procedures.	Remove reserved words from tables and procedures. For more information, see <a href="#">Working With Reserved Words</a> .

Item Type	What the Script Checks	What You Must Do
DBQL Data	Non-empty tables are reported because DBQL tables are not archived/restored.	Copy DBQL table data to a user table, if you want to migrate the DBQL data to the destination system. For more information, see <a href="#">Saving DBQL Log Data</a> .
Stored Procedures with No Source Code	Stored procedures without source code cannot be recompiled on the destination system.	Rewrite SQL procedures to include the WITH SPL option in order to store the source code in the database. For more information, see <a href="#">Fixing SQL Stored Procedures</a> .
DBC Startup String	Checks for a startup string assigned to user DBC. A startup string may prevent logons after the version switch to a new release.	Set StartupString to NULL for user DBC before the migration or upgrade.
Triggers on TDWM. RULEBYPASS	Checks for triggers on the TDWM. RULEBYPASS table.	Remove any triggers on the TDWM. RULEBYPASS table before the migration or upgrade.
Orphaned Access Rights	Checks for orphaned rows in DBC.AccessRights.	Open an incident for Teradata Services and attach the orphaned rows report file.
Secure Zones	Checks for the use of secure zones on the source system.	See Secure Zone documentation in <i>Teradata Vantage™ - Advanced SQL Engine Security Administration</i> , B035-1100.
PPI tables using hashbuckets function to define partition ranges.	Checks for PPI tables.	This is not supported as part of the migration. You must partition the tables differently prior to the migration.
Geospatial types	Checks for old geospatial type definitions which are not compatible with the latest geospatial types provided by Teradata.	In order to use the latest geospatial types and functionality provided by Vantage, you must drop these old geospatial types.
HUT locks	Checks for any HUT locks that have not been released.	You must release the HUT lock manually.
Cnsrun	This check is used to detect whether there is something wrong with the way the system is configured that will cause problems with the migration.	You should examine the /etc/hosts file and resolve the inconsistencies.

A summary section at the end of the report indicates whether the system is ready to proceed with the migration or upgrade.



## Working With Reserved Words

### About Reserved Words

Reserved words are used as keywords by Advanced SQL Engine or ANSI/ISO SQL. They cannot be used as identifiers to name database objects, such as databases, tables, columns, or stored procedures. They also must not be used as macro or stored procedure parameters or local variables, host variables, or correlation names.

Each new release of Advanced SQL Engine adds new reserved words. For every release, the *Teradata Vantage™ - Advanced SQL Engine Release Summary*, B035-1098 includes a list of the new reserved words. Teradata provides a view and table function that you can use to generate a complete list of reserved, nonreserved, and future reserved words. For more information, see *Teradata Vantage™ - SQL Fundamentals*, B035-1141.

The pre-migration script runs the `check_reserved_words.bteq` and `check_reserved_words_tpt.bteq` scripts to check for reserved words used by the database and by the Teradata Parallel Transporter (TPT) load utility. If you use TPT, you should check the results in `reservedwords_tpt.rpt`.

### Fixing Reserved Words

Eliminate reserved words from your database prior to migration or upgrade by either changing them to something other than reserved words or enclosing them in quotation marks where possible.

- To eliminate the reserved words, do one of the following:
  - Change the word to something else. For example, change reserved word `FUNCTION` to `THEFUNCTION`.
  - For some uses, such as in column names, you can enclose the reserved word in quotes to prevent the database from interpreting it as a reserved word. For example, change reserved word `FUNCTION` to `"FUNCTION"`.

---

#### Note:

The Reserved Words script may not find all instances of the reserved words, because Client databases are not checked and the names can be obscured through aliases and embedded SQL statements.

---

### Saving DBQL Log Data

The `DBQLRuleTbl` and `DBQLRuleCountTbl` are archived and restored only if database DBC is migrated. To migrate data from other DBQL tables, the data from these tables must first be copied into a user database/table. For more information about saving DBQL log data, see *Teradata Vantage™ - Database Administration*, B035-1093.

**Note:**

Definitions of the DBQL log tables often change between major releases, and the data in DBQL log tables from a prior release does not get converted to match the data collected using the definitions in the subsequent release.

All of the DBQL rules that were present on the source system are save/restored, so the same information is collected on the destination system.

Teradata Professional Services can set up standard DBQL rules and logging on the destination system.

The `dbql_out.rpt` file, output by the pre-processing script, details how many rows exist in each of the DBQL tables. The table below shows an example.

Table	Count
DBQLExplainTbl	152478
DBQLObjTbl	3078221
DBQLogTbl	219991
DBQLSqlTbl	220024
DBQLStepTbl	1073967
DBQLSummaryTbl	0

## Load Isolated Tables and Join Indexes

Load isolated (LDI) tables that participate in join indexes must be re-created without the `WITH ISOLATED LOADING` clause prior to migration.

## Fixing SQL Stored Procedures

If you are migrating between major releases, such as moving from a 16.xx release to a 17.xx release, the database restoration process automatically recompiles SQL procedures if the corresponding source code has been saved in the database.

To prevent unauthorized modification, replication, or distribution of the source code, procedures provided by Teradata and other vendors are often distributed without source code. Teradata procedures are recreated on the new release, but third-party vendors must be contacted to provide new versions of their procedures because without source code, procedures cannot be recompiled during migrations or upgrades.

The pre-migration preparation script generates a file, `sp_nospllist.txt`, which lists SQL stored procedures that were not saved with their source code. These procedures must be either modified prior to the migration or upgrade, or must be recreated after the migration or upgrade.

1. View the `sp_nospllist.txt` report.

This report shows the stored procedures that do not have their source code stored in the database.

---

**Note:**

The report shows stored procedure names in both ASCII text and hexint forms.

---

2. Recreate each listed stored procedure in one of the following ways:

- Procedures that are exclusive to your site, and that do not include source code, should be modified using ALTER PROCEDURE with the WITH SPL clause to store their source code in the database. This allows these procedures to be recompiled during the migration or upgrade. For more information on ALTER PROCEDURE, see *Teradata Vantage™ - SQL Data Definition Language Syntax and Examples*, B035-1144.
- Teradata-provided SQL stored procedures that do not have source code are recreated during the upgrade/migration process during the execution of DIP scripts. For more information on the DIP utility, see *Teradata Vantage™ - Database Utilities*, B035-1102.
- Stored procedures provided by third party vendors usually do not include source code. These require re-installation of a software package, and possibly assistance from the vendor.

## Saving Security Configuration

If you have created a custom security configuration on the source system by editing the TdgssUserConfigFile.xml file, the file must be saved in preparation for later installation and activation of the custom configuration on the destination system.

---

**Note:**

This is unnecessary if you use the default security configuration.

---

1. Locate the TdgssUserConfigFile.xml on the Control Node in the /opt/teradata/tdat/tdgss/site file.
2. Make a copy of the file and save it to secure removable media.

## About SCANDISK and CheckTable

Prior to a migration or upgrade, you should validate the integrity of the database to insure against any loss of data that might result from errors in tables or a corrupted file system. Pre-existing problems on the source system can cause issues, such as the following:

- Failure of the entire migration or upgrade operation
- Failure to detect new problems introduced by the migration on the destination system
- Apparent migration or upgrade problems that are not real
- Database failures on the destination system

SCANDISK and CheckTable are system utilities that check the integrity of the database file system and tables, and report on any problems they find. For more information on these utilities and how to run them, see *Teradata Vantage™ - Database Utilities*, B035-1102.

---

**Note:**

These utilities may take a long time to run, depending on the priority level at which they are run, the degree of checking they do, and how much data exists on the system. To speed the utilities and minimize the effect on currently logged on users, do the following:

- Run SCANDISK at a low priority level.
- Run CheckDisk at a medium priority doing Level 2 checking with the following command:

```
check dbc at level two with no error limit skiplocks priority=m  
error only;
```

CheckTable commands require a semicolon at the end.

- If there are no users logged on to the system, use the IN PARALLEL option of the CheckTable command to speed the checking.
- 

## Checking for Pending Operations

A Pending Operation indicates that a table is currently in the middle of a data load operation. During this time, the table is unusable until the load operation completes. A load operation cannot begin on one release and complete after the migration on a different release. You must allow any active load operations to finish before starting the migration.

Most tables that are in a pending state are the result of previously aborted load operations. These operations will never complete, so these tables should be deleted because they are unusable.

Coordinate with the site database administrator to make sure all pending operations are finished, and no new transfer or load operations will be started until after the migration.

1. Run CheckTable using the following command:

```
check all tables at level pendingop skiplocks priority=h;
```

2. Drop any tables that have pending operations.

# Archiving Source Data

## Basics of Archiving Data

### Archive Mechanisms

Starting in release 17.00, DSA is the only supported backup and restore mechanism.

### Database DBC

Two archive operations are required for a full system archive. First is a DBC All archive, followed by a DBC Only archive.

### Databases to Exclude

Some databases that are created by DIP scripts or replaced on a newer release should not be archived. DSA automatically excludes these databases:

- TD\_SYSFNLIB
- SYSBAR
- SYSUIF
- TD\_SYSGPL

The CRASHDUMPS database contains system- and release-specific crash information that will not be relevant to the destination system, so it should also be excluded from the migration.

SYSLIB database contains the GetTimeZoneDisplacement and algorithmic compression UDFs that may be referenced during the restoration process. SYSLIB is archived as part of the DBC All archive but needs to be restored immediately after the DBC Only restore and before the rest of the DBC All restore. After all other user data is restored, the DIP scripts ensure that the SYSLIB database is current.

The SYSSPATIAL database contains information about the geospatial data types.

- If you were using non-Teradata geospatial data types on the source system, and want to keep using those data types on the destination system, you should archive and restore this database.
- If you were using non-Teradata geospatial data types on the source system, but want to use Teradata geospatial versions of the geospatial data types on the upgraded destination system, exclude SYSSPATIAL from the migration. The SYSSPATIAL database will be automatically created on the new system, and will contain definitions for the Teradata geospatial data types.

### Secure Zones

The Secure Zones feature allows creation of one or more separate and exclusive database hierarchies, called zones, within a single Advanced SQL Engine system. Access and administration for each zone is handled separately from the Advanced SQL Engine system and from other zones.

Only the owner of a Secure Zone can archive or restore databases within that zone, and that owner cannot be DBC. Each Secure Zone owner must create archive and restore jobs for the databases in their respective zones in order to perform a full-system migration. For more information, see *Teradata Vantage™ - Advanced SQL Engine Security Administration*, B035-1100.

## Source Data Preservation

As a precaution, keep the source system operational with the original data until you are certain that the destination system is running correctly with the migrated data.

## Using DSA to Archive Data

To archive source data using DSA, you can use either the BAR Operations portlet in Teradata Viewpoint or the command line.

Before using DSA for data backup and restore, you must install and configure DSA. Teradata recommends using Viewpoint portlets for configuration, but command-line instructions are also provided. Use the Viewpoint Monitored Systems portlet to make systems available to the BAR Setup portlet. In BAR Setup you must configure the systems, nodes, backup solutions, and target groups before you can use BAR Operations for backup and restore.

See the following documents:

- *Teradata® Viewpoint User Guide*, B035-2206
- *Teradata® DSA User Guide*, B035-3150
- *Teradata® DSA - DSE for Veritas NetBackup Installation, Configuration, and Upgrade Guide*, B035-3151
- *Teradata® DSA - DSU Installation, Configuration, and Upgrade Guide*, B035-3153
- *Teradata® DSA - DSE for IBM Spectrum Protect Installation, Configuration, and Upgrade Guide*, B035-3155

## Backing Up DBC and User Data Using the BAR Portlets

You must create two backup jobs. One includes all the databases under DBC and excludes the DBC database automatically and one includes only the DBC database.

1. From the BAR Operations **Saved Jobs** view, create a backup job that saves the databases under DBC:
  - a. Click **New Job**.
  - b. Select the **Backup** job type, then click **OK**.
  - c. In the **New Backup Job** view, enter a job name, such as Backup-DBC-A11.
  - d. Select a **Source System** from the list.
  - e. Enter the user credentials.
  - f. Select a **Target Group** from the list.
  - g. Select the DBC database in the **Objects** tab.

- h. Click ☐ next to the DBC database.  
The **Settings** dialog box appears.
  - i. Check the **Include all children databases and users** box.
  - j. Click **OK**.
  - k. Click **Save**.
  - l. Click ☐ on Backup-DBC-A11 and select **Run**.
2. From the BAR Operations **Saved Jobs** view, create a backup job that saves only the DBC database:
    - a. Click **New Job**.
    - b. Select the **Backup** job type, then click **OK**.
    - c. In the **New Backup Job** view, enter a job name, such as Backup-DBC-Only.
    - d. Select a **Source System** from the list.
    - e. Enter the user credentials.
    - f. Select a **Target Group** from the list.
    - g. [Optional] Add a **Description**.
    - h. Select the DBC database in the **Objects** tab.
    - i. [Optional] To verify the parent and objects selected, click the **Selection Summary** tab.

**Note:**

Size information is not available for DBC backup jobs. N/A displays as the size value for DBC backup jobs.

- j. [Optional] To adjust job settings for the job, click the **Job Settings** tab.
  - k. Click **Save**.
  - l. Click ☐ on Backup-DBC-Only and select **Run**.
3. Save the name of both backup job save sets.  
The backup jobs must complete with a COMPLETED\_SUCCESSFULLY or WARNING status before you can create a restore job.

## Backing Up DBC and User Data Using the Command Line

Two backup jobs are required: one for all DBC user data and one for DBC only.

Each config or create step uses an XML file. The examples below include a representative XML file. See *Teradata® DSA User Guide* for details about the XML files.

1. Create a backup job for user data that includes all children of DBC but excludes the DBC database.  
Example: `dsc create_job -n DBC-A11 -f DBCA11Job.xml`

This example of DBCA11Job.xml includes all children of DBC by setting the `<includeAll>` attribute to true.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<dscCreateJob xmlns="http://schemas.teradata.com/v2012/DSC">
  <job_instance>
```

```

    <job_name>DBC-All</job_name>
    <job_description>Backup user data and all objects in
DBC</job_description>
    <job_type>BACKUP</job_type>
    <job_state>ACTIVE</job_state>
    <auto_retire>>false</auto_retire>
    <objectlist>
      <objectinfo>
        <object_name>DBC</object_name>
        <object_type>DATABASE</object_type>
        <parent_name></parent_name>
        <parent_type>BACKUP_JOB</parent_type>
        <object_attribute_list>
          <includeAll>true</includeAll>
        </object_attribute_list>
      </objectinfo>
    </objectlist>
  </job_instance>
  <source_tdpid>systemname</source_tdpid>
  <target_media>1_5_drives</target_media>
  <job_options>
    <online>>false</online>
    <data_phase>DATA</data_phase>
    <query_band></query_band>
    <dsmain_logging_level>Error</dsmain_logging_level>
  </job_options>
</dscCreateJob>

```

2. If prompted, enter login credentials.

3. Run the job.

Example: `dsc run_job -n DBC-All`

4. Create a backup job that includes only the DBC database.

A DBC only backup job cannot be run as online.

Example: `dsc create_job -n DBC-Only -f DBCOnlyJob.xml`

This example of DBCOnlyJob.xml excludes all children of DBC by setting the `<includeAll>` attribute to false.

```

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<dscCreateJob xmlns="http://schemas.teradata.com/v2012/DSC">
  <job_instance>
    <job_name>DBC-Only</job_name>
    <job_description>Backup DBC only and exclude all child
objects</job_description>
    <job_type>BACKUP</job_type>
    <job_state>ACTIVE</job_state>
    <auto_retire>>false</auto_retire>
    <objectlist>
      <objectinfo>
        <object_name>DBC</object_name>
        <object_type>DATABASE</object_type>
        <parent_name></parent_name>
        <parent_type>BACKUP_JOB</parent_type>
        <object_attribute_list>
          <includeAll>>false</includeAll>
        </object_attribute_list>
      </objectinfo>
    </objectlist>
  </job_instance>
  <source_tdpid>systemname</source_tdpid>

```



```

    <target_media>1_5_drives</target_media>
    <job_options>
      <online>false</online>
      <data_phase>DATA</data_phase>
      <query_band></query_band>
      <dsmain_logging_level>Error</dsmain_logging_level>
    </job_options>
  </dscCreateJob>

```

5. When prompted, enter login credentials.

6. Run the job.

Example: `dsc run_job -n DBC-Only`

7. [Optional] Check the status of any running job:

`dsc job_status -n JobName`

Job Status: COMPLETED\_SUCCESSFULLY indicates the job is finished. A job might also complete with warning or errors.

# Preparing the Destination System

## Initializing the Destination System

Before migrating data to the destination system, and while no users are logged on, initialize the system as described below.

1. Use the Vproc Manager to verify that all virtual processors (VPROCS) on are online.
2. Install the correct Advanced SQL Engine release and Teradata Tools and Utilities release according to the procedures in the corresponding installation guides.
3. On client platforms, install the correct Teradata Tools and Utilities release according to the procedures in the corresponding installation guide.
4. On the destination system, do the following:
  - a. Run (or have your Teradata Services representative run) the sysinit utility to initialize the system.

### NOTICE

Sysinit will remove any existing data on the destination system.

#### Note:

The only time you can change the database row format and hash function is during system initialization. This is also the only time you can enable Japanese language support. If you require assistance, consult with Teradata Customer Services.

- b. Run the DIPMIG script (which runs the DIPERRS, DIPDB, DIPVIEWS, DIPVIEWSV, and DIPBAR scripts).

#### Note:

Do not run the DIPALL script after running sysinit. Run only the DIPMIG script. If this script is not run, or the DIPALL script is run, DBC database restoration will fail and the sysinit process must be repeated on the destination system.

5. See *Teradata® DSA User Guide*, B035-3150 for instructions on adding the required system configuration, restarting DSMain, and activating the added system configuration.
6. Enable DBC logons and prevent user logons on the database by using the Database Window utility or an equivalent console utility, such as cnstern or cnstool, to enter the following console command:

```
enable dbc logons
```

## Installing Site Security Configuration

If you are transferring custom security configurations from the source system, you must replace the `TdgssUserConfigFile.xml` file on the destination system with the edited file that you saved from the source system, and then update the configuration GDO to activate the custom configuration on the destination system. If the default security configuration is instead being used, these steps can be omitted from the migration preparation process because the destination system automatically includes the new default security configuration.

1. Delete the existing default `TdgssUserConfigFile.xml` file on the destination file.

This file is located in the following directory on the Control Node:

```
/opt/teradata/tdat/tdgss/site
```

2. Transfer the saved file from the source system into the same directory on the destination system.

3. Use the `tdgssconfig` utility to update the `tdgssconfig.gdo` file as follows:

```
/opt/teradata/tdgss/bin/run_tdgssconfig
```

The `tdgssconfig` utility generates a new `tdgssconfig.gdo` file that contains the properties of the `TdgssConfigFile.xml` file.

4. Restart Advanced SQL Engine to activate the GDO by typing:

```
tpareset -f "new_tdgssconfig.gdo"
```

The security configuration that was on the source system is now on the destination system.

## The `tdlocaledef` Utility

The Locale Definition utility (`tdlocaledef`) command-line utility allows you to define or change how Advanced SQL Engine formats numeric, date, time, and currency output.

If changes to the locale definition need to be made, contact Teradata Services.

## Recording DBC.Hosts Information

The system DBC.Hosts table contains the logical host ids, host names and default character sets for each network and host connection. This information is specific to the destination system environment and was configured during the installation of Advanced SQL Engine.

During the migration, the DBC.Hosts file of the source system is transferred to the destination system, overwriting the destination system configuration data. The source system settings may not be correct for the destination system environment and may need to be updated.

Assuming that the destination system hosts table is set up correctly, the information must be saved before migrating the data from the other system, which will overwrite it. The `HostsInfo` view must be used to display the information in a readable format.

1. Log on to the destination system Control Node over the network.
2. Start BTEQ and log in.
3. Find the contents of the current `DBC.HostsInfo`:

```
SELECT * FROM dbc.hostsinfo;
```

The response is similar to:

LogicalHostId	HostName	DefaultCharSet
101	MVS1	KANJIEBCDIC5035_0I
1025	COP	KANJIEUC_OU

The output shows columns for Logical Host ID, Host Name, and Default Character Set for each host system configured in the destination system environment. COP is the network connection you logged in through for this session. In this example, the Default Character Set used for the network sessions is KANJIEUC\_OU. This example also shows there is a Channel-connected host, MVS1, connected to the destination system.

4. Save the DBC.Hosts query results to a text file for later reference.

## Copying Site-Defined Client Character Sets

If Advanced SQL Engine uses any Site-Defined Client Character Sets, you must copy some files manually to a new location. Site-Defined Client Character Sets are described in *Teradata Vantage™ - Advanced SQL Engine International Character Set Support*, B035-1125.

1. Ask your system administrator or database administrator if there are any Site-Defined Client Character Sets on the system and whether they need to be saved for future use.
2. Inspect the files in the `/opt/teradata/tdat/tdbms/starting_version/etc` directory (where *starting\_version* is the previous Advanced SQL Engine version) to see if there are any files that begin with "map."
3. Now inspect the files in the `/opt/teradata/tdat/tdbms/new_version/etc` directory (where *new\_version* is the new Advanced SQL Engine version) to see if there are any files that begin with *map*.
4. If any of the *map* files in the `/opt/teradata/tdat/tdbms/starting_version/etc` directory are not present in the `/opt/teradata/tdat/tdbms/new_version/etc` directory, copy the map files into the new directory.

# Restoring Data

## Basics of Data Restoration

Every site has established procedures for restoring data. Use the procedures established as standard for your site.

A full system restore requires both DBC All and DBC Only backups.

Some databases and objects cannot be restored on the destination system unless the related feature has been enabled on the destination system prior to restoration. Temporal and Columnar Partitioning as well as any other feature that must be purchased separately fall into this category.

If the DBC Only restore job fails, the entire procedure needs to be repeated starting at performing a sysinit of the system again. If a data restore job fails, that job can be repeated after the problem is fixed.

## Time Zone String Management

Time Zone Strings allow Advanced SQL Engine to automatically adjust for Daylight Savings Time changes, for locales that observe bi-annual time changes.

If the destination system has a Time Zone String setting, you must disable it before running the DIPMIG script. After running the DIPMIG script, you can re-enable the Time Zone String on the destination system.

## Checking for a Time Zone String

Before running the DIPMIG script, use the DBS Control utility to check if a Time Zone String is set on the destination system.

1. From a command prompt on the destination system, enter `dbscontrol` to start the DBS Control utility. For more information about the DBS Control utility, see *Teradata Vantage™ - Database Utilities*, B035-1102.
2. At the DBS Control prompt, enter `di gen` to display the General DBS Control fields.
3. Inspect General field 18 to see if there is a value for `TimeZoneString`.
  - If the value showing for `System TimeZoneString` is `Not Set`, there is no Time Zone String set on the destination system, and you do not need to make any Time Zone String changes.
  - If there is any other value showing for `System TimeZoneString`, you need to remove the Time Zone String prior to restoring data. For example:

```
18. System TimeZoneString = America Pacific
```

## Removing the Time Zone String

If the destination system has a Time Zone String, remove the string before running the DIPMIG script. The Time Zone String is defined in a text file, `tdlocaledef.txt`, that defines locale-specific information for the database system, such as the currency and numeric units, names of days of the week and months, and default formats for different data types. The `tdlocaledef.txt` file is compiled by the Teradata Locale Definition utility in order to take effect on the database system. For more information about the Teradata Locale Definition utility, see *Teradata Vantage™ - Database Utilities*, B035-1102.

1. On the destination system, locate the directory for the `tdlocaledef.txt` file, and change to that directory. For example:

```
# locate tdlocaledef.txt
/opt/teradata/tdat/tdbms/XX.XX.XX.XX/etc/tdlocaledef.txt

# cd /opt/teradata/tdat/tdbms/XX.XX.XX.XX/etc/
```

2. Save a copy of the current `tdlocaledef.txt` file. For example:

```
# cp tdlocaledef.txt tdlocaledef.txt.orig
```

3. Use a text editor to edit `tdlocaledef.txt`, and remove the value for `TimeZoneString`, leaving only the quotation marks.

For example, change a value that looks like this:

```
TimeZoneString {"America Pacific"; "-8"; "0"; "2"; "4"; "4"; "1"; "0"; "0";
"02:00:00"; "3"; "10"; "0"; "0"; "-1"; "02:00:00"; "1987"; "2006"; "-8";
"0"; "-7"; "0"; "4"; "3"; "8"; "0"; "0"; "02:00:00"; "4"; "11"; "1"; "0";
"0"; "02:00:00"; "2007"; "9999"; "-8"; "0"; "-7"; "0"}
```

to this:

```
TimeZoneString {""}
```

Save the modified `tdlocaledef.txt` file.

4. Run the `tdlocaledef` utility to compile the `tdlocaledef.txt` file.

```
# /usr/tdbms/bin/tdlocaledef -input tdlocaledef.txt -output new
```

5. Run `tpareset` to restart the database and put the new `tdlocaledef` settings into effect, removing the old Time Zone String:

```
# tpareset -f Removed Time Zone String
```

6. Run the DBS Control utility and display the General fields to confirm that the Time Zone String has been removed.

```
# dbscontrol
```

```
Enter a command, HELP, or QUIT:
```

```
di gen
```

Confirm that the value for field 18 shows as Not Set.

## Restoring the Time Zone String

If you removed the Time Zone String on the destination system before running the DIPMIG script, restore the string after you run it.

1. On the destination system, access the directory where you saved the copy of `tdlocaledef.txt`.  
For example:

```
# cd /opt/teradata/tdat/tdbms/XX.XX.XX.XX/etc/
```

2. Delete the `tdlocaledef.txt` file you edited to remove the Time Zone String, and rename the original `tdlocaledef.txt.orig` file to `tdlocaledef.txt`. For example:

```
# rm tdlocaledef.txt
```

```
# mv tdlocaledef.txt.orig tdlocaledef.txt
```

3. Run the `tdlocaledef` utility to compile the `tdlocaledef.txt` file.

```
# /usr/tdbms/bin/tdlocaledef -input tdlocaledef.txt -output new
```

4. Run `tpareset` to restart the database and put the new `tdlocaledef` settings into effect, restoring the original Time Zone String:

```
# tpareset -f set the Time Zone String
```

5. Run the DBS Control utility and display the General fields to confirm that the Time Zone String has been restored.

```
# dbscontrol
```

```
Enter a command, HELP, or QUIT:
```

```
di gen
```

Confirm that the value for field 18 is no longer Not Set.

```
18. System TimeZone String          = America Pacific
```

## Logically Deleted Rows

Rows can be logically deleted for load-isolated (LDI) tables and during certain join index operations. These rows are marked for deletion, but not immediately physically deleted from the database. After the data is restored to the destination system, tables with these rows cannot be revalidated until the rows have been physically deleted.

The `post_data_restore` script returns a `Failure 9899` error when it encounters logically deleted rows. Use the information in the script output to note the tables containing logically deleted rows, and issue an SQL `ALTER TABLE RELEASE ROWS` statement for each affected table. See *Teradata Vantage™ - SQL Data Definition Language Syntax and Examples*, B035-1144.

## Data Restoration Using DSA

Before using DSA for data backup and restore, you must install and configure DSA. Teradata recommends using Viewpoint portlets for configuration, but command-line instructions are also provided. Use the Viewpoint Monitored Systems portlet to make systems available to the BAR Setup portlet. In BAR Setup you must configure the systems, nodes, backup solutions, and target groups before you can use BAR Operations for backup and restore.

See the following documents:

- *Teradata® Viewpoint User Guide*, B035-2206
- *Teradata® DSA User Guide*, B035-3150
- *Teradata® DSA - DSE for Veritas NetBackup Installation, Configuration, and Upgrade Guide*, B035-3151
- *Teradata® DSA - DSU Installation, Configuration, and Upgrade Guide*, B035-3153
- *Teradata® DSA - DSE for IBM Spectrum Protect Installation, Configuration, and Upgrade Guide*, B035-3155



## Full System Restore Using the BAR Portlets

### Prerequisite:

When restoring an archive made on a source system with hardware block-level compression, you must install the hardware compression driver package (teradata-expressdx) on all TPA nodes of the destination system. The installation requires a reboot of these nodes. Because it allows the destination system to read the compressed archive, this requirement applies even if the destination system is not set up for hardware compression.

### Note:

The teradata-expressdx driver package is provided with systems that are equipped with compression hardware.

### NOTICE

These steps wipe the system. Use these steps for a full system restore.

The backup jobs must have successfully completed to create restore jobs from the save sets.

You must create four restore jobs from the two backup job save sets. One includes only the DBC database and three are from the save set that includes all the databases under DBC and excludes DBC.

1. On the destination system, run SYSINIT with options appropriate to your system.
2. On the destination system, [check the Time Zone setting status](#), and [disable the setting](#) if it is enabled.
3. On the destination system, start the DBS Database Initializing Program (DIP) and run the DIPMIG script, which runs DIPERRS, DIPDB, DIPVIEWS, DIPVIEWSV, and DIPBAR.
4. From the **BAR Setup** portlet, check the activation status of the destination system and do one of the following:
  - If the destination system is configured and enabled in the **BAR Setup** portlet, click **Update** for **Selector: Update system selector for JMS messages**.
  - If the destination system is not configured in the **BAR Setup** portlet, add the system and click **Apply**.
5. On the destination system, start DSMain from the Database Window (DBW) console supervisor screen by entering:  

```
start bardsmain -d DSCName
```

 where *DSCName* is the unique name of your DSC server
6. Enable logons for the DBC user only.  
 On the destination system, enter the following from the DBW console supervisor screen:  

```
enable dbc logons
```
7. Create a DBC-only restore job from the backup job save set that saved only the DBC database:

**Important:**

You need two sets of credentials for these steps: the DBC user and password for the *target* system and the DBC user and password for the *source* system.

- a. In the Bar Operations **Saved Jobs** view, click  for the DBC only backup job, and select **Create Restore Job**.
- b. Enter a **Job Name**, such as Restore-DBC-Only.
- c. Select a **Destination System** from the list.
- d. When prompted, enter logon credentials for the current DBC user and password for the target system.
- e. Select a **Target Group** from the list.
- f. Click the **Job Settings** tab, then select **Set Credentials** to enter the credentials for the DBC user and password of the source system at the time the backup save set was generated.
- g. Click **Save**.
- h. Click  for Restore-DBC-Only and select **Run**.

**Note:**

After the DBC restore is complete, the DBC password is set to the source system DBC password.

**Troubleshooting:**

If there are any errors, follow the instructions in the log file to correct the problem and run the post dbc script again.

- The post restore script output log files are saved in `/var/opt/teradata/tdtemp/post_restore_dbs version`.
- If the DBC restore job fails, check the DSC job status log to correct the issue.
- If the failure occurs before the postscript phase, you must run SYSINIT before rerunning the restore job.

If you ran `post_dbc_restore` manually, you must update the JMS system selector in the **BAR Setup** portlet once `post_dbc_restore` completes successfully.

- If the destination system is configured and enabled in the **BAR Setup** portlet, click **Update** for **Selector: Update system selector for JMS messages**.
- If the destination system is not configured in the **BAR Setup** portlet, add the system and click **Apply**.

8. Create a SYSLIB database restore job from the DBC ALL backup save set:
  - a. In the Bar Operations **Saved Jobs** view, click  on the backup job created for the databases under DBC, and select **Create Restore Job**.
  - b. Enter a **Job Name**, such as Restore-SYSLIB.
  - c. Select a **Destination System** from the list.

- d. When prompted, enter logon credentials for the current DBC user and password for the destination DBS.
  - e. Select a **Target Group** from the list.
  - f. On the **Objects** tab, clear the top checkbox, then expand the tree and select the checkbox for *only SYSLIB*.
  - g. Click **Save**.
  - h. Click ☐ on Restore-SYSLIB and select **Run**.
9. Create a restore job for all other data from the DBC ALL backup save set (excluding SYSLIB and TD\_SERVER\_DB):
- a. In the Bar Operations **Saved Jobs** view, click ☐ for the backup job for the databases under DBC, and select **Create Restore Job**.
  - b. Enter a **Job Name**, such as Restore-DBC-All.
  - c. Select a **Destination System** from the list.
  - d. When prompted, enter logon credentials for the current DBC user and password for the destination DBS.
  - e. Select a **Target Group** from the list.
  - f. Clear the checkbox for **TD\_SERVER\_DB** in the **Objects** tab.  
TD\_SERVER\_DB has dependencies that must be met before restoration.
  - g. Clear the checkbox for **SYSLIB** in the **Objects** tab.
  - h. Click **Save**.
  - i. Click ☐ for Restore-DBC-All and select **Run**.
  - j. If there are any errors, follow the instructions in the log file to correct the problem and run the post data script again.  
The post restore script output log files are saved in /var/opt/teradata/tdtemp/post\_restore\_dbs version.
10. From the BAR Operations **Saved Jobs** view, create a TD\_SERVER\_DB restore job from the backup job save set of the databases under DBC, excluding DBC:
- a. Click ☐ for the backup job for the databases under DBC, and select **Create Restore Job**.
  - b. Enter a **Job Name**, such as Restore-TD\_SERVER\_DB.
  - c. Select a **Destination System** from the list.
  - d. When prompted, enter logon credentials for the current DBC user and password for the destination DBS.
  - e. Select a **Target Group** from the list.
  - f. On the **Objects** tab, clear the top checkbox, then expand the tree and select the checkbox for *only TD\_SERVER\_DB*.
  - g. Click **Save**.
  - h. Click ☐ on Restore-TD\_SERVER\_DB and select **Run**.
11. On the destination system, start the DIP and run the DIPALL script.
12. If you disabled the Time Zone setting on the destination system, [enable it](#).
13. On the destination system, from the DBW console supervisor screen, enable logons for all users.

enable logons

## Full System Restore Using the Command Line

### Prerequisite:

When restoring an archive made on a source system with hardware block-level compression, you must install the hardware compression driver package (teradata-expressdx) on all TPA nodes of the destination system. The installation requires a reboot of these nodes. Because it allows the destination system to read the compressed archive, this requirement applies even if the destination system is not set up for hardware compression.

### Note:

The teradata-expressdx driver package is provided with systems that are equipped with compression hardware.

### NOTICE

These steps wipe the system. Use these steps for a full system restore.

Before you create a restore job, a backup job must have completed either successfully or with a warning. You must create four restore jobs, one for DBC only and three from the all DBC user data backup (SYSLIB, TD\_Server\_Database, and all other data).

1. On the destination system, run SYSINIT with options appropriate to your system.
2. On the destination system, [check the Time Zone setting status](#), and [disable the setting](#) if it is enabled.
3. On the destination system, start the DBS Database Initializing Program (DIP) and run the DIPMIG script.
4. Check the activation status of the destination system using the `dsc list_components` command, `dsc list_components -t system`, and do one of the following:
  - If the destination system is configured and enabled, run a `config_systems` command with the `skip` parameter set to `selector`, `dsc config_systems -f configsystem.xml -s selector`.
  - If the destination system is not configured, add the system using the `config_systems` command, `dsc config_systems -f configsystem.xml`. See the following sample `configsystem.xml` file.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
    <!--****The command below must be on one line,****-->
    <dscConfigSystems xmlns="http://schemas.teradata.com/v2012/DSC"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-
    instance" xsi:schemaLocation="DSC.xsd">
```

```

<system>
<!-- 'system_name' - Required, max 32 characters -->
<system_name>system7</system_name>

<!-- 'tdpid' - Required (unless skipped by option)-->
<tdpid>system7</tdpid>

<!-- 'database_query_method' -
Required, accepted values: BASE_VIEW/EXTENDED_VIEW -->
<database_query_method>BASE_VIEW</database_query_method>

<!-- 'streams_softlimit' - Required, number of streams per node per job -->
<streams_softlimit>20</streams_softlimit>

<!-- 'streams_hardlimit' - Required, max number of streams per node-->
<streams_hardlimit>20</streams_hardlimit>

<!-- 'reset_node_limit' - Optional, accepted values: true/false -->
<reset_node_limit>false</reset_node_limit>

<!-- 'skip_force_full' - Optional, accepted values: true/false -->
<skip_force_full>false</skip_force_full>
</system>
</dscConfigSystems>

```

5. On the destination system, start DSMain from the Database Window (DBW) console supervisor screen by entering:

start bardsmain -d *DSCName*, where *DSCName* is the unique name of your DSC server

6. Enable logons for the DBC user only.

On the destination system, enter the following from the DBW console supervisor screen:

enable dbc logons

7. Create a DBC-Only restore job, excluding all children of DBC, for example:

dsc create\_job -n Restore-DBC-Only -f RestoreDBCOnlyJob.xml

Sample RestoreDBCOnlyJob.xml file:

```

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<dscCreateJob xmlns="http://schemas.teradata.com/v2012/DSC">
  <job_instance>
    <job_name>Restore-DBC-Only</job_name>
    <job_description>Restore DBC only and exclude all child
objects</job_description>
    <job_type>RESTORE</job_type>
    <job_state>ACTIVE</job_state>
    <auto_retire>false</auto_retire>
    <backup_name>DBC-Only</backup_name>
    <backup_version>0</backup_version>
    <all_backup_objects>true</all_backup_objects>
  </job_instance>
  <source_media>1_5_drives </source_media>
  <target_tdpid>systemname</target_tdpid>
  <job_options>
    <enable_temperature_override>false</enable_temperature_override>
    <temperature_override>DEFAULT</temperature_override>
    <block_level_compression>DEFAULT</block_level_compression>
    <disable_fallback>false</disable_fallback>
    <query_band></query_band>
    <dsmain_logging_level>Debug</dsmain_logging_level>
    <reblock>false</reblock>
  </job_options>
</dscCreateJob>

```

```
<run_as_copy>false</run_as_copy>
</job_options>
</dscCreateJob>
```

8. When prompted, enter logon credentials:

Option	Description
Target username	Current DBC user for the target DBS.
Target password	Current DBC password for the target DBS.
Is this Restore job a DBC restore?	y
Password for the backup	DBC password of the source system at the time the backup save set was generated.

9. Run the job.

```
dsc run_job -n Restore-DBC-Only
```

After the DBC restore job is complete, the DBC password is set to the DBC password of the source system. The DBC database must restore successfully before you can restore user data.

10. Create a restore job for SYSLIB only:

```
dsc create_job -n Restore-SYSLIB -f R-Syslib.xml
```

---

**Important:**

`all_backup_objects` must be false when processing individual object restore or copy. If `all_backup_objects` is false and an objectlist is not provided, the job cannot be created.

---

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<dscCreateJob xmlns="http://schemas.teradata.com/v2012/DSC">
  <job_instance>
    <job_name>R-Syslib</job_name>
    <job_description></job_description>
    <job_type>RESTORE</job_type>
    <job_state>ACTIVE</job_state>
    <auto_retire>false</auto_retire>
    <objectlist>
      <objectinfo>
        <object_name>SYSLIB</object_name>
        <object_type>DATABASE</object_type>
        <parent_name>DBC</parent_name>
        <parent_type>USER</parent_type>
        <object_attribute_list>
          <includeAll>false</includeAll>
        </object_attribute_list>
      </objectinfo>
    </objectlist>
```

```

    <backup_name>DBC-ALL</backup_name>
    <backup_version>0</backup_version>
    <all_backup_objects>>false</all_backup_objects>
</job_instance>
<source_media>target1</source_media>
<target_tdpid>sdt12687</target_tdpid>
<job_options>
    <enable_temperature_override>>true</enable_temperature_override>
    <disable_fallback>>false</disable_fallback>
    <query_band></query_band>
    <dsmain_logging_level>Error</dsmain_logging_level>
    <reblock>>false</reblock>
    <run_as_copy>>false</run_as_copy>
    <skip_archive>>false</skip_archive>
    <skip_stats>>false</skip_stats>
    <concurrent_bld_per_tbl>5</concurrent_bld_per_tbl>
</job_options>
</dscCreateJob>

```

11. Run the job.

```
dsc run_job -n R-Syslib
```

12. Create a DBC-All restore job for user data, including all children of DBC except SYSLIB and TD\_SERVER\_DB:

```
dsc create_job -n Restore-DBC-All -f RestoreDBCAllJob.xml
```

```

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<dscCreateJob xmlns="http://schemas.teradata.com/v2012/DSC">
  <job_instance>
    <job_name>Restore-DBC-All</job_name>
    <job_description></job_description>
    <job_type>RESTORE</job_type>
    <job_state>ACTIVE</job_state>
    <auto_retire>>false</auto_retire>
    <objectlist>
      <objectinfo>
        <object_name>DBC</object_name>
        <object_type>DATABASE</object_type>
        <object_attribute_list>
          <includeAll>>false</includeAll>
        </object_attribute_list>
        <exclude>
          <excludeobjectinfo>
            <object_name>SYSLIB</object_name>
            <object_type>DATABASE</object_type>

```

```

        </excludeobjectinfo>
        <excludeobjectinfo>
            <object_name>TD_SERVER_DB</object_name>
            <object_type>DATABASE</object_type>
        </excludeobjectinfo>
    </exclude>
</objectinfo>
</objectlist>
<backup_name>DBC-ALL</backup_name>
<backup_version>0</backup_version>
<next_incremental_restore>>false</next_incremental_restore>
<all_backup_objects>>true</all_backup_objects>
</job_instance>
<source_media>target1</source_media>
<target_tdpid>abutera</target_tdpid>
<job_options>
    <enable_temperature_override>>true</enable_temperature_override>
    <disable_fallback>>false</disable_fallback>
    <query_band></query_band>
    <dsmain_logging_level>Debug</dsmain_logging_level>
    <nowait>>true</nowait>
    <reblock>>false</reblock>
    <run_as_copy>>false</run_as_copy>
    <skip_archive>>false</skip_archive>
    <skip_stats>>false</skip_stats>
    <concurrent_bld_per_tbl>5</concurrent_bld_per_tbl>
</job_options>
</dscCreateJob>

```

13. When prompted, enter logon credentials and run the job.

```
dsc run_job -n Restore-DBC-All
```

14. Create a restore job for the TD\_SERVER\_DB database:

```
dsc create_job -n R-TDSEVER -f R-TDSEVER.xml
```

---

### Important:

`all_backup_objects` must be false when processing individual object restore or copy. If `all_backup_objects` is false and an `objectlist` is not provided, the job cannot be created.

---

```

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<dscCreateJob xmlns="http://schemas.teradata.com/v2012/DSC">
    <job_instance>
        <job_name>R-TDSEVER</job_name>
        <job_description></job_description>
    </job_instance>
</dscCreateJob>

```



```

<job_type>RESTORE</job_type>
<job_state>ACTIVE</job_state>
<auto_retire>>false</auto_retire>
<objectlist>
  <objectinfo>
    <object_name>TD_SERVER_DB</object_name>
    <object_type>DATABASE</object_type>
    <parent_name>DBC</parent_name>
    <parent_type>USER</parent_type>
    <object_attribute_list>
      <includeAll>>false</includeAll>
    </object_attribute_list>
  </objectinfo>
</objectlist>
<backup_name>DBC-ALL</backup_name>
<backup_version>0</backup_version>
<all_backup_objects>>false</all_backup_objects>
</job_instance>
<source_media>target1</source_media>
<target_tdpid>sdt12687</target_tdpid>
<job_options>
  <enable_temperature_override>>true</enable_temperature_override>
  <disable_fallback>>false</disable_fallback>
  <query_band></query_band>
  <dsmain_logging_level>Error</dsmain_logging_level>
  <reblock>>false</reblock>
  <run_as_copy>>false</run_as_copy>
  <skip_archive>>false</skip_archive>
  <skip_stats>>false</skip_stats>
  <concurrent_bld_per_tbl>5</concurrent_bld_per_tbl>
</job_options>
</dscCreateJob>

```

15. Run the job.  

```
dsc run_job -n R-TDSERVER
```
16. On the destination system, start the DIP and run the DIPALL script.
17. If you disabled the Time Zone setting on the destination system, [enable it](#).
18. On the destination system, from the DBW console supervisor screen, enable logons.  

```
enable logons
```

## Troubleshooting a Full System Restore

If you have problems with a full system restore, try the following.

## Verify the Media Server IP Addresses

1. Ping the IP address from the media server to the database.
2. Ping the IP address from the database to the media server.
3. Trim any invalid or unused IP addresses.

## post\_dbc\_restore Script Fails to Launch

If the restore works, but fails to launch the `post_dbc_restore` script, you receive a message similar to the following:

```
1079 Post dbc restore script not started. Please run the post_dbc_restore
script manually with the following command:
/opt/teradata/PUTTools/td17<version>/IUMB_scripts/post_dbc_restore -s -N -j
3 TDT5WSDB/'dbc',<dbc_password>.
```

In addition, the table `SYSBAR.DSAConnectionsTbl` is not repopulated and you receive a message similar to this:

```
1001 SQL Exception 3807 3807: DBCSQL Exception: Object
'SYSBAR.DSAConnectionsTbl' does not exist.
DSAConnectionsTbl INSERT failed. All DSCs should activate system <system_name>
```

Follow these steps:

1. Manually launch the `post_dbc_restore` script:

```
/opt/teradata/PUTTools/td17<version>/IUMB_scripts/post_dbc_restore -s -N -j 3
TDT5WSDB/'dbc',<dbc_password>
```

2. Repopulate the `DSAConnectionsTbl`:

Interface	Steps
BAR Setup portlet	<p>Check the activation status of the destination system in the <b>BAR Setup</b> portlet and do one of the following:</p> <ul style="list-style-type: none"> <li>• If the destination system is configured and enabled, click <b>Update</b> for <b>Selector: Update system selector for JMS messages</b></li> <li>• If the destination system is not configured, add the system and click <b>Apply</b>.</li> </ul>
Command line	<p>Check the activation status of the system:  <code>dsc list_components -t system</code> and do one of the following:</p> <ul style="list-style-type: none"> <li>• If the destination system is configured and enabled, run this command:  <code>dsc config_systems -f configsystem.xml -s selector</code></li> <li>• If the destination system is not configured, add the system:  <code>dsc config_systems -f configsystem.xml</code> where <code>configsystem.xml</code> is similar to the example shown in <a href="#">Full System Restore Using the Command Line</a>.</li> </ul>

3. On the destination system, start DSMain from the Database Window (DBW) console supervisor screen by entering:

```
start bardsmain -d DSCName, where DSCName is the unique name of your DSC server
```

### Compile Errors Reported During a Restore

When restored by DSA, certain Teradata-provided stored procedures fail to recompile in the post\_data\_restore scripts. These procedures have dependencies that are not met until after DIPALL is run at the conclusion of the migration. You can manually recompile them using ALTER statements after DIPALL completes.

```
"SYSLIB"."JSON_SHRED_BATCH_U"
"SYSLIB"."JSON_SHRED_BATCH"
"TD_SYSEXML"."XSLT_SHRED_BATCH"
"TD_SYSEXML"."AS_SHRED_BATCH"
"TD_SYSEXML"."XSLT_SHRED"
"TD_SYSEXML"."XMLPUBLISH"
"TD_SYSEXML"."XMLPUBLISH_STREAM"
"LOCKLOGSHREDDER"."SP_LockLog_Shredder"
```

These errors are detected by the DSC when it runs the post\_data\_restore scripts and it concludes with a Warning job status rather than Complete. Check the log file noted in the error message provided by the DSC. The errors flagged by the DSC for these issues will not interfere with the rest of the restore.

The following commands will recompile the stored procedures after DIPALL has been run.

```
ALTER PROCEDURE "SYSLIB"."JSON_SHRED_BATCH_U" COMPILE;
ALTER PROCEDURE "SYSLIB"."JSON_SHRED_BATCH" COMPILE;
ALTER PROCEDURE "TD_SYSEXML"."AS_SHRED_BATCH" COMPILE;
ALTER PROCEDURE "TD_SYSEXML"."XSLT_SHRED" COMPILE;
ALTER PROCEDURE "TD_SYSEXML"."XSLT_SHRED_BATCH" COMPILE;
ALTER PROCEDURE "TD_SYSEXML"."XMLPUBLISH" COMPILE;
ALTER PROCEDURE "TD_SYSEXML"."XMLPUBLISH_STREAM" COMPILE;
ALTER PROCEDURE "LOCKLOGSHREDDER"."SP_LockLog_Shredder" COMPILE;
```

# Setting Up Advanced SQL Engine

## About Modifying Security Settings

Advanced SQL Engine initially starts with security defaults that are compatible with the security defaults in previous Advanced SQL Engine releases and with the newly installed Teradata Tools and Utilities programs. Do not modify any of the security settings in either Advanced SQL Engine or Teradata Tools and Utilities at this time. After the migration is complete, you can review the newly available security capabilities and revise your security setup if required. For more information, see *Teradata Vantage™ - Advanced SQL Engine Security Administration*, B035-1100.

## Inspecting Stored Procedures

Stored procedures are automatically recompiled after a restore. If any of the original stored procedures did not have the procedure source code (the source program language text) stored in the database with the stored procedure, the stored procedure cannot be recompiled. In these cases you must rewrite and recompile the stored procedure. For more information, see [Fixing SQL Stored Procedures](#).

If stored procedures exist on the system, the automatic recompile process generates several output log files for inspection.

- In `/var/opt/teradata/PUTTools/spconv_xx.xx.xx.xx`, inspect each of the following output log files, and take the identified corrective action(s), if necessary.

Recompile Output File	Purpose	Corrective Action
<code>spnorecomp.txt</code>	Lists all the stored procedures that failed to recompile for any reason.	Review the <code>spconvlog.out</code> file to identify the cause of failure.
<code>spconvlog.out</code>	Lists the specific cause of error for each stored procedure.	<ol style="list-style-type: none"> <li>1. Correct any errors in the SPL that may be preventing the recompile.</li> <li>2. Manually recompile the stored procedure.</li> </ol>
<code>sp_nospllist.txt</code>	Lists all the stored procedures that could not recompile because no SPL was stored with the procedure.	Re-create the stored procedure and store the SPL text with the stored procedure.

## Inspecting Java External Stored Procedures

The `post_data_restore` script recompiles Java external stored procedures (JXSPs), and generates several output files for inspection.

- In `/var/opt/teradata/PUTTools/jxspconv_xx.xx.xx.xx`, inspect each of the following output log files, and take the identified corrective actions, if necessary.

Recompile Output File	Purpose	Corrective Action
<code>jxspconv.pl.date_timestamp.log</code>	Records the execution of the <code>post_data_restore</code> script, including creation of files related to errors encountered, if any. For cleanup operations, this file shows which JXSPs and JARs have been removed. <b>Note:</b> If JXSPs do not exist on the system, this file reflects the following message: "No Teradata Java XSP /UDF or Jars...Returning from jxspconv.pl"	Review generated error files, if any.
<code>jxspteradata.bteq</code>	Contains the SQL statements that actually recompile all of the JXSPs found.	None.
<code>jxspteradata.log</code>	Lists the results of the <code>bteq</code> script execution.	Review this file for causes of recompile failures.

## Inspecting UDFs and UDTs

Recompilation of user-defined functions (UDFs) and user-defined types (UDTs) generates output files that report the results.

- Inspect the following log files to verify that UDFs and UDTs compiled properly:
  - `/var/opt/teradata/PUTTools/udfalter_xx.xx.xx.xx/udfalter.pl.date_timestamp.log`
  - `/var/opt/teradata/PUTTools/udtalter_xx.xx.xx.xx/udtalter.pl.date_timestamp.log`

## Verifying DBC.Hosts Configuration

If the new contents of the `DBC.Hosts` table are different from the original data, the data on the destination system must be changed back to the original data.

- Repeat the [Recording DBC.Hosts Information](#) task you performed when preparing the destination system.
- Compare the restored contents with the original contents recorded earlier.
- Change the new contents back to the original data.

## Verify the Integrity of Database File System and Tables

After a migration or upgrade, you should run the `SCANDISK` and `CheckTable` system utilities to verify the integrity of the database file system and tables. For more information, see [About SCANDISK and CheckTable](#) and *Teradata Vantage™ - Database Utilities*, B035-1102 .

## Resetting the DBC Password

After the migration, the DBA must reset the DBC user password from the temporary one that was used for the migration to the permanent password to be used on the new system.

If there is a system password rule that disallows reusing old passwords within a given period of time, the DBA may not be able to reset the DBC password back to the original value it had before the migration. The value in the PasswordReuse column of the Data Dictionary table DBC.SyssecDefaults determines the number of days that must elapse before you can reuse a password. If this value prevents the DBA from setting the password back to the original at the end of the migration, the DBA must either choose a different password or delete the old password from table DBC.OldPasswords, so the DBC password can be reset back to the original value that was in effect before the migration.

## Enabling Logons

1. Use the Database Window utility or an analogous console utility, such as cnstern or cnstool to enter the following console command:  
`enable logons`
2. From the console command line, enter the following command:  
`pdestate -a`  
The screen output should include `Logons are enabled`

## Additional Information

### Teradata Links

Link	Description
<a href="https://docs.teradata.com/">https://docs.teradata.com/</a>	Search Teradata Documentation, customize content to your needs, and download PDFs. Customers: Log in to access Orange Books.
<a href="https://support.teradata.com">https://support.teradata.com</a>	One-stop source for Teradata community support, software downloads, and product information. Log in for customer access to: <ul style="list-style-type: none"> <li>• Community support</li> <li>• Software updates</li> <li>• Knowledge articles</li> </ul>
<a href="https://www.teradata.com/University/Overview">https://www.teradata.com/University/Overview</a>	Teradata education network
<a href="https://support.teradata.com/community">https://support.teradata.com/community</a>	Link to Teradata community

### Related Documentation

Title	Publication ID
<i>Teradata® DSA User Guide</i>	B035-3150
<i>Teradata® DSA - DSE for Veritas NetBackup Installation, Configuration, and Upgrade Guide</i>	B035-3151
<i>Teradata® DSA - DSE for IBM Spectrum Protect Installation, Configuration, and Upgrade Guide</i>	B035-3155
<i>Teradata® DSA - DSU Installation, Configuration, and Upgrade Guide</i>	B035-3153
<i>Teradata® Viewpoint User Guide</i>	B035-2066