Customer Best Practices for Teradata Database Upgrades

Major Software Upgrades

Revision History

Summary of Changes			
Date	**Description of Change**	**Rev#**	**Author**
June 23,2008	Initial input from all areas	0.1	SFT Members
July 5, 2008	First integrated draft	0.2	Keith Jones
July 17, 2008	Second integrated draft	0.3	Keith Jones
July 12,2008	Third Integrated draft	0.4	Shelley Perrior
September 4, 2008	Fourth Integrated draft	0.5	Shelley Perrior
July 20, 2012	Updates to reflect current products & versions	1.0	Jerry Blevins
Nov 1, 2019	Update/Review for TD16.20	2.0	Kevin J Lewis

Important Notes	
Date	
July 5, 2008	This initial version of the document was written while Teradata V2R6.2 was widely accepted as the current stable version, and V12 was starting to be adopted by customers. Any instructions and software references in this document should be taken with that context in mind.
July 20, 2012	This updated version of the document was written while Teradata TD13 was widely accepted as the current stable version, and TD14 was starting to be adopted by customers. Any instructions and software references in this document should be taken with that context in mind.
Nov 1, 2019	Reviewed for applicability to TD16.20. Added information and links for Td Bench access.

Table of Contents

1. Introduction

> This paper is prepared for the use and benefit of Teradata customers. It is written by customers; for use by customers.
>
> Please read document first in its entirety before beginning any sections

A successful upgrade of any computing environment requires careful planning, coordination, and execution. A Teradata computing environment is no different. There are many factors that need to be considered and brought together for a successful plan. The success you experience is in part dependent on the success of your efforts throughout the process.

In general, there are three things that influence the success or failure of an upgrade:

- The quality of the product being added/upgraded
- The service procedures followed by Teradata associates
- The planning and execution of operational procedures by the customer

All three need to come together without flaw during the upgrade process in order to achieve success. The first two are the responsibility of Teradata. For the purposes of this paper, they are considered out of scope, but assumed to be present and bear no negative impact on the upgrade process.

This paper will focus on those operational procedures which are the responsibility of the customer. They include, but are not limited to:

- Planning
- Coordination
- Communication
- Testing
- Benchmarking
- Execution
- Project management

Upgrades/Updates can include database software (major, minor, maintenance, patches/fixes) and database hardware (upgrade (Floor/Cloud sweeps), expansion, migration). This white paper will focus only on database software upgrades. We plan to cover hardware upgrades (Floor/Cloud sweeps) and expansions in a separate document.

> ***Note to the reader.*** *This guide is provided as a courtesy of the Teradata Service Focus Team (SFT). It comes as is; with no implied support. The procedures represent the combined learning of the customer members who contributed to it. The procedures are meant to be high level guides and do not represent step-by-step procedures. They are accurate to the best combined knowledge of the team, but should always be*

followed with caution and good judgment as they are not guaranteed to be free from flaws.

2.   The Upgrade Process for Teradata Software

2.1   Major Database Software Upgrades, Defined

This paper deals primarily with major database software upgrades. A major database software upgrade is moving from one database version to another as represented by the following software version numbering format: xx.yy.zz.nn, with "xx" being a major release of the database, "yy" minor, "zz" maintenance and "nn" patches/fixes. Moving from Teradata TD15.20 to Teradata TD16.20 is a major upgrade. However, many of the points in this document could apply to minor, TD15.00 to TD15.10, software upgrades as well.

A Teradata Database *major* release generally includes significant new features and enhancements which may involve the conversion of system tables, user data or custom user features.

A Teradata Database *minor* release may include a number of new features and enhancements, as well as some Requests for Change (RFCs) and Discrepancy Reports (DRs).
A minor upgrade generally does not include the conversion of system tables, user data or custom user features.

2.2   Upgrade Process Overview Diagram

This diagram gives an overview of the upgrade process.   The body of this document will describe each task in more detail:
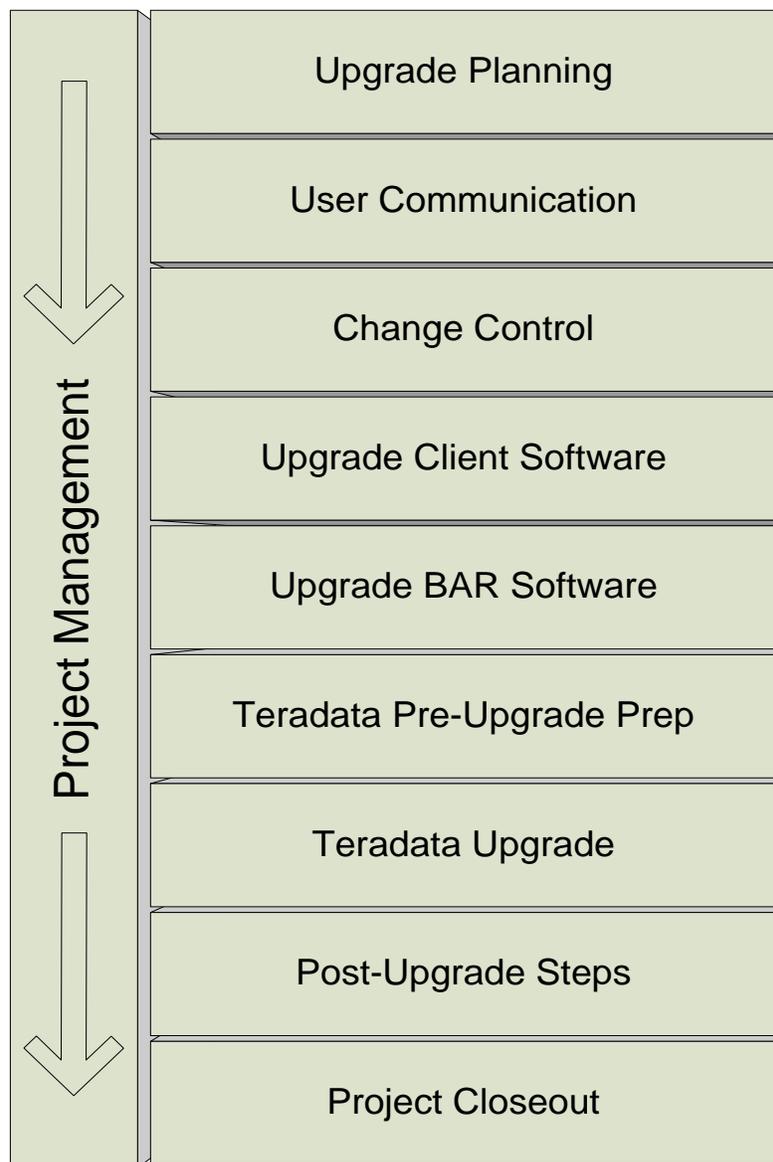
Figure 1: Upgrade Process Overview Diagram

3. Project Management

Any successful project requires strong and focused project management. A Teradata upgrade is a significant undertaking that is well worth treating as a formal project with both Customer's Project Managers and Teradata Project Managers aligned.

Assign a project manager to coordinate all the required tasks as a single project. This person will ensure that all involved parties communicate effectively, schedules are set and followed, and all tasks are completed successfully.

Create an upgrade team with DBAs, operations, developers and others to ensure cross-team participation and communication. Have a kick-off meeting to update everyone and then provide regular status updates to the members. You may want to include your Teradata System Support Engineer (SSE) and Service Experience Manager (SEM) in the team meetings.

Some SFT member DBAs have found it helpful to maintain an upgrade plan template document that covers both hardware and software changes. It lists items to remember for specific circumstances. For example, when adding new nodes you would want to remember to request firewall rules changes, obtain IP addresses, make DNS entries, etc. This is a living document to which you would add new memory joggers learned during each new upgrade.

4.   Upgrade Planning

An upgrade can be complex. Take the time up front to investigate the new version before taking the plunge. The time invested upfront will pay off later.

4.1   Research the new version

Find the differences between the new version and your current version. There are usually significant changes and improvements between major versions. Places to find information are:

- Partners or Teradata Analytic Universe (TAU) Presentations. Various presentations exist describing the new version and its features which are available via the TAU website to attendees of the conference for up to 2 years.
- Package Comparison Tool. This is available on the Teradata At Your Service website under the "Tools and Reports" menu. Given two package version numbers, this report displays the DRs (Discrepancy Reports) in the first package but not the second, and the DRs in the second package but not the first. This knowledge is invaluable in determining your particular risks and benefits of upgrading to a particular new version.
- Reserved Words Document. This will help you identify any necessary changes to existing application SQL. Teradata provides scripts to check existing database objects for use of reserved words. However, these scripts don't identify usage as aliases in the user queries.   These can be found by querying the DBQL tables. (*Note: if you are not logging SQL text in DBQL you really should be.*)
- SQL Fundamentals Appendix E (SQL Feature summary) will detail SQL differences.
- New version manuals, specifically the Release definition and Release Summaries for the new version. These are available on CD and on the Teradata website at https://docs.teradata.com
- You may also want to consider becoming involved in an EFT (Engineering Field Trial) or BETA testing program for the new version.
- Research DevX with an upgrade tag. https://www.teradata.com/Developer
- Reference applicable Orange Books available via the Teradata Service Portal

4.2   Identify your dependent applications

Be sure you know all the Teradata client applications in use and investigate whether they are able to use the new Teradata version. Some Teradata and/or third-party applications using Teradata will need to be upgraded in order to work with the new version. Once they are upgraded they will need to be tested. For critical applications, it may be necessary to delay the Teradata upgrade until a compatible version is available.

Some potential clients include:

- Third-party reporting tools such as Microstrategy, Cognos, SAS and Business Objects.
- Third-party ETL tools like ab initio and Informatica.

- Teradata Tools and Utilities (TTU) client software (See following section for details).
- Teradata Backup and Recovery (BAR) software (See following section for details).
- Teradata Managed Servers (Viewpoint, Data Mover, TMSM, BAR, Unity) See section 8 & 9
- Replication solutions (Unity, Golden Gate)
- DBA housekeeping scripts that are dependent on the data dictionary (DBC) schema. Check for changes to DBQL, resusage and adjust DBA scripts accordingly.

4.3   Pre-upgrade Benchmarking

If your management is concerned about the Return on Investment (ROI) of the upgrade, you'll want to identify queries you can use for before/after benchmarking. Types of SQL you may want to include are expedited queries, canary queries, heavy hitting queries, frequently run queries, ETL SQL and possibly some more complex queries that may have caused problems in the past. We recommend you identify at least 10-20 different priority queries.

Td Bench is a tool provided by Teradata that can assist with the Benchmarking task. The tool provides an integrated reporting against DBQL and Resusage. The install process dynamically adapts reporting views to different release and PDCR tables. A TestTracking table is maintained on Teradata for each test that records precise timestamps for the start/stop of each test along with information about the nodes present/up, AMPs, software release and TASM rule set. Views against DBLQL and Resusage allow comparison of tests by RunID.

TdBench is a tool specifically designed to simulate database workloads. With this tool, you can:

- Measure performance before vs after a change to add indexes, partitioning, compression, etc
- Measure the impact to your DBMS of changes to settings, a patch, or a new software release
- Simulate a workload for a new application or a proof of concept
- Compare the performance of one platform to anotherCompare performance of different data base vendor's product

Collect explain output, TSET information, elapsed times and details for each query. DBQL is a great resource to capture this information. Be sure you have a backup of DBQL or whatever table you use to store this information so you are sure to have it after the upgrade.

If possible, run the queries when the system is quiet before and after the upgrade to accurately compare results. Note that run-time should be long enough to retrieve comparable runtime figures. Keep in mind they might run faster than usual on a quiet system.

You may want to include TPT, Fastload, Fastexport, Multiload, T-Pump, archive and restore jobs in your benchmark suite.

Another important area you may need to consider is performance if you have a mixed workload. You will want to ensure that the various workloads will continue to perform as expected when the system is saturated. Creating a scenario to test this will help to prevent surprises after an upgrade.

For more detailed information regarding Benchmarking practices please refer to the SFT Customer Best Practices for Teradata Benchmarking whitepaper available at the Service Focus Team Website

4.4 Regression Testing

If you are lucky enough to have a test system, we strongly advise that you install the new Teradata version there first and do some regression testing. The benchmark queries mentioned above would be good contenders for the testing. Be sure to include priority queries from your most critical applications.

Use the TSET utility to replicate your production test bed including statistics to the test system. This will ensure that you get the same explains in test that you'll get in production.

If you can't copy down production data to test with, at least run explains on your critical queries to check for syntax errors and parser problems like segment violations (3610 errors).

4.5 Key Risk Questions to be answered

Whenever you undertake a major software upgrade, there will always be risks that need to be mitigated prior to execution of the upgrade. The following table provides a list of some common risk items and mitigation steps that can be taken to reduce or prevent any business impact associated with the risk.

Upgrade Risk Item	Description	Mitigation Steps	Teradata Guidance
PPI Table Handling	Some customers have experienced unpredictable variability in reconfig run times due to PPI data demographics, number of partitions, size of table, etc.	• Produce data demographic report and have it reviewed by Teradata Area change control to understand if any significant issues exist. • Consider using one of two approaches on subset of tables identified to reduce risk. Either: copy PPI tables to a non-PPI equivalent, delete rows, and re-insert after the merge; Or backup PPI tables, and restore tables after merge activities complete. • Add PPI contingency in project plan equivalent to 40-50% of the total estimated time for data redistribution and delete phases for reconfig to account for potential variability.	• Add PPI contingency to project plan based on reconfig run time estimates. • Final run of data demographic report within 7 days of scheduled system expansion. Not expected to find any major problems based on results of recent BCP expansion. • Review with customer that no significant table changes are planned on the system for PPI tables.
Statistics Collection	There are often questions as to whether statistics need to be collected before or after an upgrade. Teradata always recommends having fresh statistics for the best optimizer performance. However, statistics collection can take	• Develop a plan to focus on stats recollection during the outage window for mission critical tables, using a tiered approach. The customer needs to identify what tables are deemed mission critical. • Priority for what stats are collected on a given table should be in the following order: ○ Partition Stats (on system defined columns PARTITION on both PPI and non-PPI	• Assuming there isn't enough time to recollect all statistics, put focus on mission critical tables for statistics recollection during outage window. • Recollect all Partition and NUSI statistics on the system. • Recollect statistics

Upgrade Risk Item	Description	Mitigation Steps	Teradata Guidance
	some time, and it may not always be feasible to do a full statistics collection during an upgrade window.	tables) – this is mandatory ○ PI and NUSI stats (NUSI stats should always be recollected after a system expansion) ○ Single Columns stats (High Confidence in explains) ○ Multi Column stats (High Confidence in explains) ○ Single Column stats and Multi column stats (both with low Confidence) • Consider freshness of statistics as you develop your plan (i.e. when the last time statistics were collected on a given table).	that are known to be stale, and have exceeded their normal re-collection time. • Phase in remaining statistics either on a best effort basis during non-business impacting hours, or during their next scheduled refresh. Plan for isolated statistics refresh given unexpected performance issues.
Check Table @ Pending Ops Level during system outage window	This is designed to execute as a merge prep process, and is a standard utility to determine if issues exist with any objects that must be addressed prior to an upgrade process commencing. While the utility typically takes 10-20 minutes to execute on a system, successfully dealing with identified objects may extend the time required in the plan, which could add time to the outage window. The check table pending ops can take much longer to run on larger systems with many objects.	• Execute multiple checktable level 2 operations as part of normal planning to ensure that issues are resolved well ahead of time. • Execute checktable level 2 on dbc, level 1 on other tables post merge, as well as scandisk	• Execute checktable level 2 during normal prep activities and successfully resolve any identified issues. • Executed another checktable level 2 just prior to the system outage window and address any remaining issues. • Execute checktable @ pending ops level at beginning of outage window – there should be no significant errors.

Showspace and Packdisk Dependencies	Showspace is a standard utility that is designed to measure free cylinder space on a Teradata system. If the result shows less than 30% freespace then it is required to execute the Packdisk utility to recover space to support the reconfig operation. While Showspace runs quickly, Packdisk could take several hours to execute.	• Execute the Showspace utility multiple times during normal merge prep planning to trend free space availability. • Discuss with customer any major planned initiatives to be implemented that will impact freespace prior to the system merge weekend. • Executing Showspace utility one last time during final system backup activities prior to merge, and if Packdisk is required have it execute during the backup window and not a standalone task during the system outage window. • Delete unnecessary tables in advance of system expansion to free up space on the system	• Trend freespace percentage up until the system merge weekend as much as possible based on prep windows that are available. • Execute final Showspace during final system backup window prior to scheduled system outage. • If needed, execute Packdisk to run in parallel during system backups such that the outage window plan isn't extended. • Delete tables that are no longer necessary in advance.
Spare parts inventory on-site	Depending on maintenance service levels with Teradata, spare parts will be at the data center site. As a contingency for system merge/expansion operations, appropriate inventory of spare parts should be readily available to minimize potential downtime or negative performance for failed components.	• Teradata Customer Services personnel will inventory on-site spare parts and ensure that all required spare parts are at the data center site in proper quantities to minimize risk. • This should be a documented task in the project plan/run book, and be completed as close as possible to the merge outage window.	• Ensure that all spare parts are on-site at customer's data center just prior to merge/expansion outage.

VPN, Service Connect (Axeda), External Network Connectivity Problems	Teradata's Area change control team accesses your systems remotely during the system merge weekend. The typical way is through a VPN connection that is already in place for normal customer support reasons. The failure of VPN connectivity can compromise what can be done, and add significant risk to getting all system expansion work completed.	• Understand from customer if there is scheduled network maintenance or other activities that would potentially impact Teradata VPN system connectivity. If present, determine ahead of time what can be done to pre-test or mitigate any exposure during the expansion weekend. • Use WebEx or other web-based connectivity option that allows for sharing desktops as a backup method for connecting to the Teradata systems from a customer desktop. This should be tested in advance of the expansion weekend. • As a third option, leverage onsite Teradata SSE resources to support the Teradata America's change control team in completing all required template activities. While an option, this is least desirable.	• Assume normal VPN connectivity in the plan. • Plan for and pre-test WebEx or other web-based desktop sharing solution to be used as backup • Consider having local SSE support during the system expansion weekend that can be onsite in the data center if the first two options do not work – this would have to be an agreed upon between Teradata and customer.
Slow Disk LUN	Teradata AMP's are associated with a rank of disk devices that represent a logical unit or LUN as part of the file system. A disk operating a significantly reduced performance can negatively impact reconfig run times based on the parallel nature of Teradata systems.	• Execute the PDECLIO utility that has been developed by Teradata to detect slow disk LUN performance in advance of the reconfig process commencing. Monitor run times of standard Teradata utilities to determine if they are outside of expected completion windows. Examples would include the reconfig estimator that runs prior to the reconfig process starting.	• Execute PDECLIO and respond appropriately by taking slow disks out of the configuration prior to the reconfig process starting.

| Internal Network Settings and Checkout | To realize benefit of new Teradata nodes in an expanded system, the nodes should be accessible from a customer's public or private network post merge. This will include potential assistance in the areas of network setup, DNS, switch management, etc. | • Make sure that formal network setting checkouts are part of the project plan/run book in advance of the system merge operation. This will vary by site. Where feasible, ensure that network settings for new nodes are consistent with existing nodes in the configuration.
 • If DNS or other network changes need to be implemented after merge completion and before system turnover to users, project plan must contain proper local checkout steps with personnel available to respond to last minute changes as required.
 • Care should be given to also check on Hot Standby Node (HSN) setup. Depending on the Teradata release that a customer is on, validating accurate HSN setup is important to ensure that the HSN will correctly enter the configuration given a node failure in new expansion cliques. Intelligent switches such as F5 and leveraging Teradata releases that support the laddered connection protocol (LCC) will mitigate this risk. | • Teradata can assist in certain network checkout steps as it relates to Teradata node IP addresses.
 • Customer should make necessary provisions to ensure correct network setup, including DNS, hostfile, advanced switches (i.e. F5), etc. |

Reconfig Execution time longer than desired based on outage window available	For certain customers, mostly large installations, reconfig time estimates may be longer than can be accommodated. In some cases, other steps can be taken to shorten this time. Each customer must decide based on their own installation whether the mitigation steps make sense.	• Prepare a create/drop script for all JOIN INDEXES and HASH INDEXES. Drop before the reconfig, create after the reconfig and collect stats on them. • Prepare a create/drop script for all NUSIs. Drop before the reconfig, rebuild after the reconfig, collect stats where needed. NUSIs are created in the delete phase of reconfig and not having any may shorten your delete phase time substantially. • Drop any tables/data that doesn't need to be there. Reconfig time is affected by the number of objects and the amount of data. Even empty tables have to have their table header copied to new AMPs. • Drop any noPI tables because they will not be redistributed.	• This should be considered as an approach only if it adds significant time savings. Depending on what needs to be done, the mitigation steps could add much more time to outage window. A detailed analysis is required to understand the benefits to the customers Teradata environment. • It is a good practice to do a table review and drop very large non-critical tables (i.e. PDCR) that can be restored after the reconfig completes when time pressure exists to shorten the outage window.
Permanent Journal Management	Customers that use permanent journaling will have to perform steps to ensure that journals are managed correctly during system expansions	• Ensure all data in journals is checkpointed with save, then dumped and deleted • Prepare scripts to alter tables to remove journaling and also to put it back again afterwards • Prepare scripts to remove the default journal table from your table databases and also to put them back afterwards • Prepare scripts to drop the journal table from your journal databases and also to put them back afterwards • Run in order the remove scripts produced in 2, 3 and 4. • Following the upgrade run the 'put back' scripts in the reverse order	• Perform necessary steps to make sure that permanent journals are successfully migrated to the larger expanded system environment.

| General risks associated with system expansions and merges | Teradata supports investment protection through the addition of nodes to expand systems, including the ability to have nodes of different generations co-exist with each other. This provides the customer with flexibility, but it is important that appropriate planning steps are conducted in advance to ensure a successful expansion. | • Provide ample time an appropriate merge prep windows for Teradata Area Change Control team to complete all activities.
• Meet twice per week 4-6 weeks out from expansion window to develop detailed run book with multiple walk-throughs between Teradata and the customers DBA team.
• Run PDCR data collection reports to be reviewed by the Teradata America's change control team for identification of any issues.
• Execute new Teradata Customer Services Pre-Upgrade Assessment Service. This service is conducted within 7 days from the system expansion event and includes work performed by the CS organization:
    o Accesses the system and performs the data collection process.
    o Data Collection process is non-intrusive and will not impact the system.
    o Analyzes data to pinpoint known performance opportunities.
    o Works with system/site SSE personnel to communicate issues that require additional attention or follow-up.
    o Completes Assessment and delivers to the system/site SSE and SEM for review. | • Ensure that all required prep windows are obtained prior to system expansion.
• Conduct bi-weekly status meetings to develop and review plan, and build out joint detailed run book.
• Execute recommended pre-upgrade activities performed by either/or Teradata Professional Services personnel and Teradata Customer Services personnel. |

5. User Communication

You will want to keep your users well-informed regarding the upgrade timing and content.

Ideas for user communications include:

- Pass on your findings on the use of new reserved words in existing user SQL.
- Publish upgrade information and schedule on your intranet website.
- Consider giving a new features overview class.
- Request users to drop unneeded objects from their personal databases. This is more important for hardware upgrades, but always a good thing to do.
- Verify with users that all their critical data will be backed up, if you don't already regularly back up user personal databases.
- Provide a demo of new tools available in the new release.

- Execute a pilot rollout of TTU to ensure a smooth upgrade.

6. Change Control

The argument for formal change control is that system upgrades are managed in a controlled and successful manner such that there is minimal negative impact on the users of the system.

Main points to consider and potentially include in your change control procedures are:

- Include all software that is being touched. Apart from the obvious new Teradata database release and its client software, you need to include all data exploitation tools being upgraded as well, e.g. SAS, Business Objects and Cognos. Include any data load tools being upgraded as well.
- Be very clear about when changes are to take place and how long they are expected to take.
- Are there pre-requisites that affect the order of changes? An obvious one is that TTU software would normally be upgraded ahead of database software.
- Does your site have software distribution mechanisms like packages for desktops and servers? If not, put in place processes to ensure that the change takes place in the same way for each similar change.
- Will your own staff carry out the changes? Will Teradata employees make any of the changes? Will other 3rd party employees make the changes?
- All changes must have a backout mechanism. Remember to test that mechanism where possible.
- Are you prepared to accept non-supported configurations? For example, if you are not able to upgrade exploitation tools before a database software upgrade, can you obtain approval for running a non-supported configuration? Who from your company will approve the configuration?
- Agree well ahead of time who will work on the project and obtain commitment that they will be available.
- Make sure your change control includes a list of contact names and details for use during the time of change.
- Be sure to know who will test changes and be sure that they have a proper test plan.
- Consider what progress needs to be communicated to whom and at what points. Communication could make or break the project.
  - o A Steering Committee may need a monthly report, whereas others will need the very latest information during the actual upgrade.
  - o During the upgrade give timely (maybe every 4 hours) updates on the upgrade progress.
- Ensure that you take into account the lead time required for the Teradata Change Control process.

The key to successful change is planning and communication. Change control helps ensure all eventualities have been considered and communicated.

7. Upgrade Client Software

Once you have identified all dependent client software in your environment, communicated with your users and opened your change control you can proceed with applying any required client application updates. This is something you want to do long before doing the Teradata server software upgrade.

Work with your third-party software vendors to upgrade their products according to their recommendations. Specifics on that are outside the scope of this document. The remainder of this section will focus on the Teradata-supplied Tools and Utilities (TTU).

The acronym "TTU" refers to the Teradata-supplied Tools and Utilities that enable a client to interface with a Teradata database server. This includes software that runs on Windows as well as various flavors of UNIX and Linux. The Teradata ODBC driver, SQL Assistant, Teradata Administrator (WinDDI) and BTEQ are the most commonly used components, but there are many others. See the *Teradata Tools and Utilities Installation Guide* for a complete list and detailed installation instructions.

8.   Upgrade Teradata Tools and Utilities

Normally the TTU client software will have to be upgraded prior the database being upgraded. Future TTU versions will allow more flexibility, but generally the rule is to upgrade your tools prior to the database being upgraded.

Investigate which version of TTU should be installed. Depending on the timing of when TTU is to be installed, different versions will be available for the database version in question. Ensure this TTU version will work with the database version, OS version and any third-party tools.

Identify servers and workstations that need upgraded versions of the Teradata client tools. Create an implementation plan and checklist to ensure none are missed.

Some companies push TTU clients to workstations. However, it is very common for users with workstation administrative rights to install their own TTU software. In that case you may wish to have each user download the latest install packages from a central source.

Uninstall the old version prior to installing the new one. In the past various issues have surfaced during uninstalls, like leftover DLLs and registry entries. You will want to have a procedure to check that the uninstall completes successfully.

Have database and application teams test the new TTU version. Include tests of TTU along with any third-party tools used. Be sure to do regression testing of unique features of your environment, regardless of prior certifications. You may need to apply software patches as problems are found.

9.   Upgrade Teradata Managed Server Applications

Identify which managed servers need to be upgraded and to what version they need to be upgraded to. It is recommended that the managed servers be upgraded prior to the database upgrade. If this cannot be done they should be the first on the list of task in the DB migration plan.

These key steps should be used when upgrading a managed server.

• 	Create plan for upgrade
• 	Initiate change control process for both Teradata and local
• 	Communicate outage to users if they access managed server
• 	Run full backup of managed server
• 	Turn managed server over to SSE to apply upgrade
• 	Run test cases once upgrade complete
• 	Communicate to users that managed server is available
• 	Close change control

10. Upgrade BAR Software

Your backup and recovery (BAR) software may need to be upgraded prior to the database upgrade. Common BAR software is ARCMAIN on mainframes, and NetBackup (Symantec) and NetVault (BakBone) on the open clients with Data Stream Architecture (DSA) .

First, check the Teradata roadmap to determine which versions of software are compatible with the version of database software you are upgrading to. By understanding your options up front, you will save significant time later.

Should an upgrade to the BAR platform be necessary, you'll need to take inventory of all system software levels in use in the environment (TTU, OS, Patch Level, NBU/NetVault, Tape Silo Software, etc.). In addition to verifying BAR software compatibility, you'll need to verify the OS environment will continue to support this upgraded software. Should your corporation have certain standards of software, make sure the new versions are permitted for use within your company, or begin the process of certifying them as soon as possible. Verify there are no licensing issues with upgrading to the new software version.

Depending on your corporate structure, several groups may need to be involved in this process. Make sure you have representation from the following types of groups:

- NetBackup/NetVault Support (Symantec or BakBone)
- BAR Master and Media server support (OS)
- Tape silo server support (if present)
- Any tape library server support
- Teradata DBA/System admin
- Someone that has or can verify appropriate permissions for the upgrade process on all platforms

You should schedule your BAR upgrade around any critical backups your location may require, in the event additional time is required for the upgrade process.

The BAR master server is typically upgraded first. It is a good idea to save any custom configuration files present on the master server before the upgrade (java, perl, nbu, etc.) as the upgrade process will probably write over them. The master server should also have a full, comprehensive backup prior to the upgrade in case any files need to be restored. Also verify the master catalog is in good shape and has a good backup. Part of the upgrade process is to convert the master catalog for use on the new version. After the software has been upgraded on the master server, run several tests to validate any older versions of media server clients still work with the new master server.

Once the BAR master server software is upgraded, you are ready for the media servers to be upgraded. After those are upgraded and running backups on the local drives successfully, the Teradata software may be upgraded. Record the current software levels of all Teradata software in the event you need to back off the upgrade.

Here is a typical BAR upgrade scenario example using NetBackup:

- Uninstall TDE/TARA.
- Uninstall ARC, PIOM, CLI, ODBC, ICU, GSS (in that order).
- Install the new versions of GSS, ICU, ODBC, CLI, PIOM, ARC (in that order).
- Verify you can log on using ARC to your target Teradata database.
- Install the new version of TDE or TARA.
- Perform your all-system validation again:
- Backup all local media server drives (non-Teradata backup)

- This validates NBU can communicate with the master server and tape drives
- Backup a test Teradata table using all media servers
- This validates NBU can communicate with Teradata
- Restore a test Teradata table
- Test a scheduled backup within NBU to validate the scheduling still works as expected
- If you have a separate test or development Teradata server, consider performing a complete DBC system dump and restore to ensure that that the new version of Arcmain will restore the 'old' version of the system in case of disaster recovery

If all runs well, document your new software levels and make sure you test again once your Teradata database is upgraded. Be aware of new TTU patches that are released in the meantime that may resolve newly discovered performance or connectivity issues.

11. Teradata Pre-Upgrade Preparations

This section concerns those tasks that need to be done shortly before the Teradata software is upgraded. This is a typical list of things to do:

- Open change Controls – Teradata & local.
- Clean reserved words (including TPT).
- It is a good idea to reboot nodes one week prior to upgrade.
- Ensure you have backups of DBS control settings, SysSecDefaults, ResUsage, TDWM, XCTL, PSF settings, etc. in case you need to do a full recovery if something goes wrong.
- Run full backups of important data. All-amp backups are preferred over clustered backups if the amp configuration will be different after the upgrade. Remember that cluster backup datasets must be restored serially to a different amp configuration. That may require an unreasonably long restore window.
- Clean-up unnecessary data and objects
- Run scandisk and checktable on the existing system. Do this enough in advance to allow time to address any problems discovered.
- Clean leftover Spool.
- Generate DDLs of Hash, Join Indexes, Triggers and UDFs. These should be dropped prior to the upgrade.
- If you use journaling, you may need to drop your journal tables prior to the upgrade.
- Disable DBQL and offload to history repository.
- Purge ampusage over 180 days old.
- Resolve checktable issues (tables in a pending load status).
- For any daily archiving of dictionary tables ensure that any dictionary changes implemented by this upgrade are understood. (as an example, PDCR)

12. Teradata Upgrade

Now that the system is fully prepared, the actual upgrade may begin. While the software upgrade itself will typically be done by your Teradata Team, these are last-minute functions the customer DBA may need to perform:

- Quiesce the system

- Abort remaining user sessions
- Backup DBC
- Bring down any TDPs
  Shutdown Teradata Manager server disable the system being upgraded on the Viewpoint servers
- Shutdown open systems backup application (NetBackup, NetVault)
- Disable unnecessary user ID's
- Stop any job schedulers such as AutoSys, cron, etc.
- Run checktable Pending OP and resolve any identified issues
- Ensure that all Hash, Join Indexes, Triggers and UDFs have been dropped
- Turn the system over to the Teradata Team to perform the software upgrade

13. Post-Upgrade Steps

After Teradata has completed and validated the upgrade, Teradata will turn the system back over to the DBA. The DBA would then perform a sequence of functions, including:

- Run checktable DBC only at level 3. This should only take a few minutes.
- Run scandisk and checktable to verify the rest of the file system is clean.
- Backup DBC and any other critical databases. If any new system databases were created as part of the upgrade, remember to grant the appropriate dump/restore privileges on them and add them to your BAR scripts.
- Run collect stats on application tables if recommended for this upgrade version.
- Check/reset TDWM, DBScontrol, XCTL and PSF settings.
- Restart DBQL, Access logging, System Security, Priority scheduler, etc. as before.
- Restart the Teradata Manager server and enable the system on the Viewpoint servers.
- Recreate Hash, Join Indexes, Triggers and UDFs that were dropped prior to the upgrade.
- Recreate journal tables (if any).
- Make sure the default date format is set correctly. This often gets changed.
- The upgrade scripts would have identified and recompiled any Teradata Stored Procedures (TDSP).
- Recompile and test any user stored procedures.
- Restore DDL for system databases that you may have modified (i.e. Sys_Calendar.CALENDAR).
- Validate statistics are collected as should be on DBC and user tables. Recollect if necessary.
- May need to adjust user spool settings if number of amps changed.
- DBA regression testing. Run test program to validate all settings, connectivity, programs, DBC tables, procedures, etc.
- Test applications such as SQL Assistant, BTEQ, BTEQWin, OLELoad, Teradata Administrator, Visual Edge, TASM, AQM, Teradata Manager, backup, restore etc.
- Start/confirm TDP's and enable logons.
- Restart job schedulers such as AutoSys, cron, etc.
- Restart open systems backup application.
- Notify customers of system availability.
- Sign off on user acceptance of upgrade with Teradata.
- Close Change Controls – Teradata & Local.

At this point you could run your post-upgrade benchmarks and compare them to your pre-upgrade benchmarks. Publish results.

14. Project Closeout

Shortly after the upgrade is complete and users are happily using the system you will want to have a project wrap-up meeting. The project manager should lead a "lessons learned" session and document what went well and what could be done better next time. This information will be invaluable for planning the next major upgrade.

If you are maintaining an upgrade plan template document (see the Project Management section above), this would be a very good time to update it with any new learnings.

Hit your management up for a party. You deserve it!

15. Closing

The SFT is a committee of customers that works closely with Teradata on issues related to support services and other areas that fall beyond the scope of product enhancements. Members represent the concerns of Teradata customers by serving as catalysts for service improvements, providing ongoing feedback to Teradata and the other Teradata customer committees.

The members of the SFT sincerely hope that you found this document useful. Please let us know of any comments or suggestions you may have. See you at Teradata Analytic Universe!

You can find our contact information and more about the SFT at our website