Teradata Database Administration Training

Chapter 1 – Introduction and Good Advice

- What is Parallel Processing?
- Start Small and Think Big
- Give your Enterprise the Tools they need
- Model the Business with ERwin
- Educate the Business on the Business by Sharing the Model
- Load Your Models and have the SQL Built Automatically
- Five Brilliant Pieces of Teradata (1 of 5) is MPP
- Five Brilliant Pieces (2 of 5) are Tactical Queries
- Five Brilliant Pieces (3 of 5) Is a Traffic System
- Five Brilliant Pieces (4 of 5) Is Viewpoint
- Five Brilliant Pieces (5 of 5) Are Data Processing Options
- Support Large Queries, but Monitor them closely
- Experiment and Improve Loading Data Strategies
- Compress Your Data with Multi-Value Compression
- Separate your Production System from Your Test System

Chapter 2 - Teradata Architecture Fundamentals the DBA must know

- Parallel Architecture
- The Teradata Architecture
- All Teradata Tables are spread across ALL AMPS
- Teradata Systems can Add AMPs for Linear Scalability
- AMPs and Parsing Engines (PE’s) live inside SMP Nodes
- Each Node is Attached via a Network to a Disk Farm
- Two SMP Nodes Connected Become One MPP System
- There are Many Nodes in a Teradata Cabinet
- This is the Visual You Want to Understand Teradata
- Responsibilities of the DBA

Chapter 3 – The Primary Index is the Axis of all Teradata Systems

- The Primary Index is defined when the table is CREATED
- A Unique Primary Index (UPI)
- Primary Index in the WHERE Clause - Single-AMP Retrieve
- A Non-Unique Primary Index (NUPI)
- Primary Index in the WHERE Clause - Single-AMP Retrieve
- A conceptual example of a Multi-Column Primary Index
- Primary Index in the WHERE Clause - Single-AMP Retrieve
- A conceptual example of a Table with NO PRIMARY INDEX
- A Full Table Scan is likely on a table with NO Primary Index
- Table CREATE Examples with four different Primary Indexes
- What happens when you forget the Primary Index?
- Why create a table with No Primary Index (NoPI)?

Chapter 4 – A DBA's best friend - The Data Dictionary

- The Data Dictionary Resides in User DBC
- The DBC.DBCInfoV View
- Querying the Data Dictionary
- Using the Keyword USER
- Restricted Views have an X at the End of their Name
- The V is New with Teradata V12
- The V and the Restricted X are Now Often Combined
- A Recap of What We Have Learned So Far
- The DBC.DatabasesV View
- The DBC.Users View
- The DBC.Tables View
- Using DBC.Tables to find out about Fallback
- The DBC.Indices View
- The DBC.Columns View

- Clever Queries for the DBC.ColumnsV View
- New V14 - The DBC.PartitioningConstraintsV View
- The DBC.AccountInfo View
- The DBC.AMPUsage View
- Clearing Out the DBC.AMPUsage Data
- The DBC.AllTempTables
- The DBC.Triggers
- The DBC.All_RI_ChildrenV
- DBC.SessionInfoV Information
- DBC.LogonOffV
- AllRoleRights, AllRightsV, UserRightsV and UserGrantedRightsV
- The DBC.Profiles View
- RoleMembers, RoleInfo, UserRoleRights and ProfileInfoVX,
- Understanding that Space is based on a Per-AMP Basis
- Total Space for a Single Database or User
- Using the Data Dictionary to see the Space for Everyone
- Finding the Perm Percent Used
- Finding the Perm Percent Used with a HAVING Clause
- Finding the Perm Percent Left with a HAVING Clause
- Creating a Macro for Perm Percent Used with a Dynamic %
- Orphaned Spool Files That Need to be deleted
- Finding Table Sizes
- Finding Skew in the Tables in a Database
- Finding Skew in a Table
- Display the Distribution of a Column per AMP
- Your Users and Databases
- DBC Tables used in the Collect Statistics Process
- The DBC Table DBC.Next
- DBA Advice - ClearPeakDisk to Reset Peak Space
- DBA Advice – Clean out these Tables Periodically
- The DBC.AssociationV View

- The DBC.JournalsV View
- DBC.Databases2V is for Unresolved Reference Constraints
- The DBC.All_RI_ChildrenV for Inconsistent RI
- The DBC.ShowColChecksV View
- The DBC.ShowTblChecksV View
- The DBC.PartitioningConstraintsV View
- The DBC.AccessLogV View
- The DBC.AccessLogV View for Today's Queries
- The DBC.AccessLogV View Denials for Today
- DBC.DBQLRulesV
- DBC.QryLogV
- DBC.QryLogSummaryV
- ResUsage Macros
- Executing the ResUsage Macro DBC.Resnode
- The DBC.IdCol Table

Chapter 5 - How Teradata Tracks Objects

- Teradata Assigns each Object a Unique Numeric ID
- The Table ID
- The Table ID in Greater Detail
- Looking at the TableID inside the actual Cylinders
- A More Detailed View of TableID inside the actual Cylinders
- The Blocks Below are All Associated with the Same Table
- Bits, Bytes and More
- Cylinder Sizes

Chapter 6 - Creating Users and Databases

- Creating Users and Databases
- Password Security Meanings
- Now we have Two Users in the Teradata System
- A Grant Statement so others Create a Database or User
- And so the Teradata Hierarchy Begins

- Creating a Database
- Users are Given Passwords While Database are Not
- Teradata Administrator Can CREATE Users
- The Modify User Statement
- A Clever Way to Reset a User Password
- Accounts and their Associated Priorities
- Creating a User with Multiple Account Priorities
- Self-Nicing to change Account Priorities
- Account String Expansion (ASE)
- The DBC.AccountInfo View
- The DBC.AMPUsage View
- Account String Expansion (ASE) in Action
- Test – Run queries Under All Accounts for TeraTom
- The DBC.AMPUsage View

Chapter 7 - Profiles

- Profiles
- Getting Started for Profile Creation
- Creating A Profile and a User
- Password Security
- Password Security Meanings
- Creating A Profile and then Modifying a User
- Quiz – What are the Profile Values?
- Answer to Quiz – What are the Profile Values?
- Quiz – What are the Profile Values After Null?
- Answer to Quiz – What Are the Profile Values After Null?
- The DBC.ProfilesVX View
- The DBC.ProfilesV View
- The DBC.AccountInfoVX View
- ProfileInfoVX, RoleMembers, RoleInfo and UserRoleRights
- Teradata Administrator Can CREATE Profiles (1 of 2)

- Teradata Administrator Can CREATE Profiles (2 of 2)
- Dropping a Profile
- The Effects of Dropping a Profile

Chapter 8 – Roles

- Roles
- Getting Started for Role Creation
- Create A Role and then Assign that Role It's Access Rights
- Create a User and Assign them a Default Role
- A Role vs. a Profile
- Granting a Role to a Current User
- Active Roles
- Setting Your Active Role to ALL
- Roles and Valid Objects
- Roles and Invalid Commands
- Nesting of Roles
- Nesting of Roles in Action (1 of 3)
- Nesting of Roles in Action (2 of 3)
- Nesting of Roles in Action (3 of 3)
- Quiz – What Databases Does Mandy Have Access To?
- Answer – What Databases Does Mandy Have Access To?
- GRANT WITH ADMIN OPTION Command
- REVOKE ADMIN OPTION FOR Command
- RoleMembers, RoleInfo, UserRoleRights and ProfileInfoVX,
- DBC Tables for AllRoleRights, AllRightsV, UserRightsV and UserGrantedRightsV

Chapter 9 - Access Rights

- The Objects That Require Access Rights
- Objects and Available Access Rights
- A Few Examples to Get You Started

- There are Three Types of Access Rights
- There are Three Types of Access Rights
- There are Three Types of Access Rights
- A Dinner Invitation of Access Rights
- One of the Problems with Access Rights
- The Rights for SysDBA and TeraTom
- The GRANT Statement
- Create A Role and then Assign that Role It's Access Rights
- GRANT to PUBLIC
- GRANT To ALL DBC
- GRANT Using the ALL Keyword
- GRANT Database Strategy for Users, Views and Tables
- Inheriting Access Rights
- GRANT at the Column Level
- GRANT for the Ability to CREATE Secondary Indexes
- Access Rights to CREATE Triggers
- The REVOKE Command
- DBC Tables for AllRoleRights, AllRightsV, UserRightsV and UserGrantedRightsV
- The GIVE Statement
- A DROP User can be Better than a GIVE Statement
- Removing a Level in the Teradata Hierarchy

Chapter 10 - Collect Statistics

- The Teradata Parsing Engine (Optimizer) is Cost Based
- The Purpose of Collect Statistics
- When Teradata Collects Statistics it creates a Histogram
- The Interval of the Collect Statistics Histogram
- Histogram Quiz
- Answers to Histogram Quiz
- What to COLLECT STATISTICS On?

- Why Collect Statistics?
- How do you know if Statistics were collected on a Table?
- A Huge Hint that No Statistics Have Been Collected
- The Basic Syntax for COLLECT STATISTICS
- COLLECT STATISTICS Examples for a better Understanding
- The New Teradata V14 Way to Collect Statistics
- COLLECT STATISTICS Directly From another Table
- Where Does Teradata Keep the Collected Statistics?
- The Official Syntax for COLLECT STATISTICS
- How to Recollect STATISTICS on a Table
- Teradata Always Does a Random AMP Sample
- Random Sample is Kept in the Table Header in FSG Cache
- Multiple Random AMP Samplings
- How a Random AMP gets a Table Row count
- Random AMP Estimates for NUSI Secondary Indexes
- USI Random AMP Samples are Not Considered
- There's No Random AMP Estimate for Non-Indexed Columns
- A Summary of the PE Plan if No Statistics Were Collected
- Stale Statistics Detection and Extrapolation
- Extrapolation for Future Dates
- How to Copy a Table with Data and the Statistics
- How to Copy a Table with NO Data and the Statistics
- When to COLLECT STATISTICS Using only a SAMPLE
- Examples of COLLECT STATISTICS Using only a SAMPLE
- Examples of COLLECT STATISTICS for V14
- How to Collect Statistics on a PPI Table on the Partition
- Teradata V12 and V13 Statistics Enhancements
- Teradata V14 Statistics Enhancements
- Teradata V14 Summary Statistics
- Teradata V14 MaxValueLength
- Teradata V14 MaxIntervals

- Teradata V14 Sample N Percent
- Teradata V14.10 Statistics Collection Improvements
- Teradata V14.10 Statistics Collection Threshold Examples
- Teradata V14.10 AutoStats feature
- Teradata Statistics Wizard

Chapter 11 – Locking

- The Four Major Locks of Teradata
- The Read Lock
- The Read Lock and Joins
- The Write Lock
- The Exclusive Lock
- The Three Levels of Locking
- Locking at the Row Hash Level
- Locking at the Table Level
- Locking at the Database Level
- The Ongoing Battle between Read and Write Locks
- Compatibility between Read Locks
- Why Read Locks Wait on Write Locks
- Why Write Locks Wait on Read Locks
- The Access Lock is Different from the Other Locks
- What is the Purpose of an Access Lock?
- Locking Modifiers - Locking Row, Table or Database
- All Views should consider the Locking for Access Statement
- What is a Dead Lock or a Deadly Embrace?
- Pseudo Tables are designed to minimize Dead Locks
- Pseudo Tables are referenced in the Explain Plan
- Incompatible Locks Wait on each Other
- The Checksum Lock of Teradata
- The Nowait Option for Locking
- The Automatic Locking for Access Button inside Nexus

- Viewpoint Lock Viewer
- Viewpoint Lock Viewer Lets You Configure Your View
- What is a Host Utility (HUT) Lock?

Chapter 12 – Protection Features

- A List of the Protection Features
- Transient Journal Protects the Transaction Integrity
- The Transient Journal in Action
- A Single Transaction could Involve All AMPs
- The Secret to turning off the Transient Journal
- The Transient Journal’s Write Ahead Logging (WAL)
- A Node with 40 AMPs and 40 Dedicated FSG Caches
- The Transient Journal’s Write Ahead Logging (WAL)
- Working Example of the Write Ahead Log (WAL)
- The First Step in our Example of the Write Ahead Log (WAL)
- The Second Step in our Example of the Write Ahead Log
- The Third Step in our Example of the Write Ahead Log
- The Fourth Step in our Example of the Write Ahead Log
- The Last Step in our Example of the Write Ahead Log
- Fallback to Protect against an AMP Failure
- Fallback Clusters
- AMPs in a Cluster are Physically Separated
- The Reason AMPs in a Cluster are Physically Separated
- The Price you pay for Fallback
- How to Create a Table with Fallback
- How to Create a Table with No Fallback
- How to Alter a Table to Add or Drop Fallback
- What is a Virtual Disk?
- Why do AMPs each have Four Physical Disks?
- Is a Mirror just like Looking into a Mirror?
- RAID 1 Mirroring – Redundant Array of Independent Disks

- What does RAID Protect?
- How Does RAID Fail?
- Do RAID and Fallback have a Connection?
- What is a Clique?
- If a Node goes down the AMPs migrate within the Clique?
- Does Teradata Reset during a Node Failure?
- Four Node Cliques
- Migrating AMPs in Four Node Cliques
- The Hot Spare Node
- The Hot Spare Node in Action
- With a Hot Spare a Second Teradata Reset isn't Needed
- A Node, It's AMPs and their Disks
- How Cliques are Physically Defined
- Cliques are cabled so Migrating AMPs can access their Disks
- A Review of Fallback and Clusters
- An Example of Fallback and Clusters
- Summary of the facts for Fallback, Clusters, and Cliques
- The Permanent Journal
- Difference between the Transient and the Permanent Journal
- Difference Between the Before and After Permanent Journal
- Full System Backup compared to an After Journal
- How Full System Backups work with the After Journal
- The Many Different Permanent Journal Options
- Where is the Permanent Journal Stored?
- Using Common Sense about Journal Locations
- After Journals are Never stored in the Same Node or Clique
- What is a Dual After Journal?
- What is a Dual Before Journal?
- What is a Journal?
- Creating a Table with Fallback and a Before and After Journal
- Does Fallback Affect a Permanent Journal?

- Permanent Journal Rules
- Example 1: Permanent Journal Scenarios to Test the Rules
- Example 2: Permanent Journal Scenarios to Test the Rules
- Example 3: Permanent Journal Scenarios to Test the Rules
- How to Create Database with a Permanent Journal
- Creating Tables under different Journal Circumstances
- Permanent Journal’s Three Main Areas
- The Current Journal consists of the Active and Saved Areas
- Permanent Journal Commands
- Deleting a Permanent Journal
- Some Great Advice for Maintaining the Permanent Journals
- Recovery Using the Permanent Journals
- The Journals View in DBC (DBC.Journals)
- Archive Recovery Console (ARC)
- Reasons You Might Utilize ARC
- ARC raising the BAR (Backup Archive Restore)
- ARC Commands in Alphabetical Order
- An ARC Example of an Archive and then a Restore

Chapter 16 - The Cold, Hard Teradata Facts

- What is Parallel Processing?
- The Basics of a Single Computer
- Teradata Parallel Processes Data
- Parallel Architecture
- The Teradata Architecture
- All Teradata Tables are spread across ALL AMPS
- Teradata Systems can Add AMPs for Linear Scalability
- Understand that Teradata can scale to incredible size
- AMPs and Parsing Engines (PEs) live inside SMP Nodes
- Each Node is attached via a Network to a Disk Farm
- Two SMP Nodes Connected Become One MPP System

- There are Many Nodes in a Teradata Cabinet
- Inside a Teradata Node
- The Boardless BYNET and the Physical BYNET
- The Parsing Engine
- The AMPs Responsibilities
- Teradata Parallel Processing
- Each Table has a Primary Index that is Unique or Non-Unique
- The Hash Map Determines which AMP will own the Row
- A Unique Primary Index Spreads the Data Evenly
- The AMP Adds a Uniqueness Value to Create the Row-ID
- Each AMP Sorts Their Tables by the Row-ID
- A Non-Unique Primary Index Skews the Data
- Comparing the Same Table with Different Primary Indexes
- Unique Primary Index Queries are a Single AMP Retrieve
- A Non-Unique Primary Index is also a Single AMP Retrieve
- Teradata has a No Primary Index Table called a NoPI Table
- There are Normal Tables and then there are Partitioned Tables
- A Visual of One Year of Data with Range_N Per Month
- Partitioning is designed to eliminate the Full Table Scan
- A Partition # and Row-ID = Row Key
- An AMP Stores its Rows Sorted in only Two Different Ways
- AMPs Moves Their Data Blocks into Memory to Read/Write
- The Most Taxing thing for an AMP is Moving Blocks into Memory
- Rows are Stored in Data Blocks which are stored in Cylinders
- Rows for an AMP Stored Inside a Data Block in a Cylinder
- An AMP's Master Index is Used to Find the Right Cylinder
- The Row Reference Array (RRA) Does the Binary Search
- A Block Splits into Two Blocks at Maximum Block Size
- Data Blocks Maximum Block Size has Changed (V14.10)
- The New Block Split with Teradata V14.10
- The Block Split with Even More Detail in Teradata V14.10

- There is One Master Index and Thousands of Cylinder Indexes
- Each Table has a 48-bit TableID

Chapter 17 - How Teradata Tracks Objects

- Teradata Assigns each Object a Unique Numeric ID
- The Table ID
- The Table ID in Greater Detail
- Looking at the TableID inside the actual Cylinders
- A More Detailed View of TableID inside the actual Cylinders
- The Blocks Below are All Associated with the Same Table
- Bits, Bytes, and More
- Cylinder Sizes

Chapter 18 – AMP Worker Tasks

- Teradata is a Message Passing System
- The Parsing Engine Parses the SQL and comes up with a Plan
- What is an AMP Worker Task (AWT)?
- Each AMP has 80 AMP Worker Tasks (AWTs)
- Each Query Takes Up One or More AMP Worker Tasks
- An All-AMP Query Usually Won't Use More Than 4 AWTs
- A Live Example of AWTs in Action
- There are 24 AWTs Reserved for Internal Work
- How Utilities Use AWTs
- Monitoring AMP Worker Tasks with ResAMPCpuByGroup

Chapter 19 - Deep Dive Overhead for each Row

- Why Go Deep inside the Overhead of a Row?
- A Row Layout in Teradata
- Row Length
- Row ID
- How The Row Hash is created for Each Row
- Unique Primary Indexes have Even Distribution

- The AMP adds a Uniqueness Value to Its Rows
- The Row-Hash is 32-bits and so is the Uniqueness Value
- Non-Unique Primary Indexes have Skewed Data
- Flag Byte
- Presence Byte
- Presence Byte is used to show Null Values in each Row
- A Close-up look at the Presence Byte for Nulls
- An Extreme example to look at the Presence Byte for Nulls
- Quiz – How Many Presence Bits used for these Columns?
- Answer – How Many Presence Bits used for these Columns?
- Quiz – How Many Presence Bits used with NOT NULL?
- Answer – How Many Presence Bits used with NOT NULL?
- Quiz – Which bit will be Set to a One?
- Answer – Which bit will be Set to a One?
- Quiz – How Many Presence Bits Needed Now?
- Answer – How Many Presence Bits Needed Now?
- What Happens when we need more than One Presence Byte?
- An Example that must use a 2nd Presence Byte
- An Example that must use a 2nd Presence Byte
- Quiz – Answer the Presence Bit/Byte Questions?
- Answers to the – Answer the Presence Bit/Byte Questions?
- Quiz - How Many Nullable Columns are Possible?
- Answer- How Many Nullable Columns are Possible?

Chapter 20 - Compression

- Important Information about Compression
- Presence Bytes are also used for Compression
- Why One Byte (8 bits) can represent up to 255 Values
- Answers to One Byte (8 bits) can represent up to 255 Values
- Now that you Understand that 8 Bits can Represent 0 – 255
- A Compression Example that Compresses Two Values

- A Compression Example that Compresses Three Values
- Quiz – Name that Compression Value
- The Next Important Concept in Compression
- Quiz – Can you Fill in the Compression Values?
- Answer – Can you Fill in the Compression Values?
- The Last Major Concept in Compression
- Quiz – Using One Presence Byte for Multiple Columns
- Answer – Using One Presence Byte for Multiple Columns
- Quiz – How Many Presence Bytes are Needed?
- Answer – How Many Presence Bytes are Needed?
- Advanced Quiz – Fill in the Presence Bits?
- Answer to Advanced Quiz – Fill in the Presence Bits?
- The Cost Vs. the Savings
- The Cost List of Compression
- A Deeper Dive Into NULL Values
- Quiz - How Much Space Did We Just Save?
- Answer - How Much Space Did We Just Save?
- Advanced Quiz - How Much Space Did We Just Save?
- Advanced Quiz - How Much Space Did We Just Save?
- Using the DBC Tables in a Compression Experiment
- A Compression Test
- A Compression Test
- We then moved all Eight Tables to another Database
- Compression Reports with Nexus and SmartCompress
- We Then Created Two Global Temporary Tables
- We Then Created and Executed our Macro
- Report Comparing Compressed and NonCompressed Tables

Chapter 21 - Data Stored in the Row

- The Varchar Offset
- An Example of a Varchar Offset

- An Example of Two Varchar Columns
- The Fixed Columns
- An Example with Multiple Fixed Columns
- Compressible Columns
- An Example with Fixed Columns and A Compression Column
- An Example with A Fixed Column and a Compressed Varchar
- VARCHAR Columns
- An Example of a Fixed Column and a Varchar
- Teradata's Maximum Row Size

Chapter 22 - How Data Rows are Stored in Blocks

- Why Go Deep inside Data Blocks?
- In The Beginning a Table is created
- Every AMP has the Exact Same Tables
- Rows are Stored in Blocks
- Each Table Header and Data Block have the Same TableID
- AMPs Moves Their Data Blocks into Memory to Read/Write
- AMPs can Read/Write their Rows once they are in FSG Cache
- Every Data Block Starts with a Data Block Header
- Every Data Block Ends with a Data Block Trailer
- Each Block has a Row Reference Array (RRA)
- The Row Reference Array (RRA) is in Descending Order
- A Binary Search is always done through the RRA
- A Binary Search is a quick Search among thousands of Rows
- The Ref Array Pointer in the Row Layout in Teradata
- How Blocks of Data Begin in Teradata
- How Blocks of Data Grow in Teradata
- Did You Notice the Row Reference Array (RRA)?
- A Great Picture of a Single AMP's Data Block with Details
- Data Blocks Grow until they Reach Maximum Block Size
- The Block Split

- The Block Split with Even More Detail
- The Block Split Showing Two Blocks with Greater Detail
- Blocks Continue to Split as Tables Grow Larger
- Reminder - Data Blocks Maximum Block Size has Changed (V14.10)
- Reminder - The New Block Split with Teradata V14.10
- Reminder - The Block Split with Even More Detail in Teradata V14.10
- Reminder - Teradata V14.10 Block Split Defaults
- FYI – Some Advanced Information about Data Block Headers

Chapter 23 - Disk Cylinders and the Master Index

- Disks have Cylinders which hold Data Blocks
- Rows are Stored in Data Blocks which are stored in Cylinders
- A Real World View of Rows inside a Data Block in a Cylinder
- A Top down View of Cylinders
- There are Hot, Warm, and Cold Cylinders
- Cylinders are used for Perm, Spool, Temp, and Journals
- Synchronized Scan (Sync Scan)
- EXPLAIN Using a Synchronized Scan
- Intelligent Memory (Teradata V14.10)
- Teradata V14.10 Intelligent Memory Gives Data a Temperature
- Data deemed VeryHot stays in each AMP's Intelligent Memory
- Intelligent Memory Stays in Memory
- Each AMP has Their Own Master Index
- Each Cylinder on an AMP has a Cylinder Index
- Quiz – What Two Things Does and AMP Read?
- Answer – What Two Things Does and AMP Read?
- Quiz – How Many Row Reference Arrays do you See?
- Answer – How Many Row Reference Arrays do you See?
- Quiz – How Many Row Reference Arrays are there Now?
- Answer – How Many Row Reference Arrays do you See?

- Quiz – How Many Row Reference Arrays in Total?
- Answer – How Many Row Reference Arrays in Total?
- Quiz – How Many Cylinder Indexes are Here?
- Answer – How Many Cylinder Indexes are Here?
- A More Detailed Illustration of the Master Index
- A Real-World View of the Master Index
- An Even More Realistic View of an AMP's Master Index
- The Cylinder Index
- An Even More Realistic View of a Cylinder Index
- How a Query using the Primary Index works
- How the AMPs Do a Full Table Scan
- How An AMP Reads Using a Primary Index
- Teradata Assigns each Object a Unique Numeric ID
- The Table ID
- The Table ID in Greater Detail
- Looking at the TableID inside the actual Cylinders
- A More Detailed View of TableID inside the actual Cylinders
- An Even More Realistic View of a Cylinder Index
- Bits, Bytes, and More
- Cylinder Sizes
- How TVS Monitors and Migrates Tables
- How TVS Monitors and Migrates Partitioned (PPI) Tables
- A Summary of the Master and Cylinder Index

Chapter 24 - Teradata Virtual Storage (TVS)

- Solid State Drives (SSD) Vs. Hard Disk Drives (HDD)
- Teradata Uses Two Types of Disks
- Traditional Teradata Without Teradata Virtual Storage (TVS)
- Teradata With TVS in a Conceptual Diagram
- What TVS is Responsible for Doing
- The Benefits of Teradata Virtual Storage (TVS)
- What is a Clique?

- If a Node goes down the AMPs migrate within the Clique?
- Review of a Clique
- Teradata Virtual Storage (TVS) Manages within a Clique
- Before TVS vs. TVS Today with Teradata V13.10
- TVS Operates in Two Different Modes
- TVS Knows the Disks and Which Cylinders are the Fastest
- A Concept Called Recency
- Data Placement and Migration
- Review - Intelligent Memory (Teradata V14.10)
- Review - Teradata V14.10 Intelligent Memory Gives Data a Temperature
- Review - Data deemed VeryHot stays in each AMP's Intelligent Memory
- Review - Intelligent Memory Stays in Memory

Chapter 25 - Teradata Writes and Blocks

- A Teradata Write
- Teradata Insert (Option 1 of 3) has enough space for the Insert
- Teradata Insert (Option 2 of 3) is a Defragment to make Space
- Teradata Insert (Option 3 of 3) is to Get a Bigger Block
- Checksum Determines if a New Block is Needed
- A Reminder of How Rows are Sorted with Block Utilities
- A Reminder of How Rows are Sorted with SQL Inserts
- When a Block Reaches Maximum Size, it Splits into Two
- A Block Split Always Sorts the Rows Perfectly Once Again
- In Teradata V14.10 the Maximum Block Size is 1 Megabyte
- Cylinder Sizes
- Teradata V14 Large Cylinders
- Quiz on Block Split
- Answer - Quiz on Block Split
- Quiz – How many Items are in the Picture?
- Answer to Quiz – How many Items are in the Picture?
- Quiz on Teradata V14.10 Block Split

- Answer - Quiz on Block Split
- Blocking Terms for Teradata V14 and Below
- Blocking Terms for Teradata V14.10 and Beyond
- Block Sizes and Filling of Cylinders
- Space Fragmentation (1 of 2)
- Space Fragmentation (2 of 2)
- What is a Defrag? (1 of 2)
- What is a Defrag? (2 of 2)
- What Happens When a Cylinder is Full?
- What is a Mini-Cylpack? (1 of 2)
- What is a Mini-Cylpack? (2 of 2)
- What is a Mini-Cylpack Vs. a Pack Disk?
- A Pack Disk Picture
- New Teradata 13.10 Auto Cylinder Pack Feature
- A Pack Disk Honors the Free Space Percent
- Free Space Percent
- Three Examples of the Free Space Percent
- Simpler Terms for our Free Space Percent Examples
- The Free Space Percent can be set in Three Ways
- Two Table Create Examples of Min and Max Block Size
- The Same Table Creates using KILOBYTES (KBYTES)
- Why Would I Want Bigger or Smaller Block Sizes?
- How Does Teradata Manage Space?
- How Can Many Big Blocks Become Many Small Blocks?
- Merge Datablocks (13.10 Feature)
- Merge Datablocks Details
- Setting Merge Datablocks in DBS Control or at Table Level
- How Have Customers Previously Handled Shrinking Blocks?

Chapter 26 – Access Logging

- Access Logging
- Security for the DBA

- The Tables and Views Associated with Access Logging
- Begin Logging Options
- Begin Logging, View Rules, See Log Data, and End Logging
- Begin Logging Examples
- The DBC.AccessLogV View
- The DBC.AccessLogV View for Today’s Queries
- The DBC.AccessLogV View Denials For Today
- Controlling the Log Files

Chapter 27 – DBQL Query Logging

- DBQL Query Logging
- The Tables and Views Associated with DBQL
- There are Seven Major Tables to Store DBQL Entries
- The Views for the Major DBQL Tables
- Begin Query Logging Default Information
- Begin Query Logging WITH Options
- Begin Query Logging Examples
- Begin Query Logging LIMIT Options
- Begin Query Logging LIMIT Examples
- SUMMARY and THRESHOLD have Additional Options
- Begin Query Logging with Additional Options Examples
- Begin and End Query Logging Examples
- Replace Query Logging Statement
- An Inside Look at the View DBC.DBQLRulesV
- The Columns in the View DBC.DBQLRulesV
- Begin Logging, View Rules, See Log Data and End Logging
- DBC.DBQLRulesV
- DBC.QryLogV
- DBC.QryLogSummaryV

Chapter 28 – ResUsage

- ResUsage
- Major Tables to Store ResUsage Entries
- The ResUsage Views
- ResUsage Macro Information
- ResUsage Macros
- Executing the ResUsage Macro DBC.Resnode
- DBC.Resnode Major Column Explanation
- ResAMPCpuByGroup
- ResCpuByAMP
- ResCpuByGroup
- ResCpuByNode
- ResCpuByPE
- ResHostByGroup
- ResHostByLink
- ResHostTotal
- ResHostTotalDay
- ResHostTotalHour
- ResIvprMigrate
- ResIvprMigrateHour
- ResLdvByGroup
- ResLdvByNode
- ResMemByGroup
- ResMemMgmtByNode
- ResNetByGroup
- ResNetByNode
- ResPeCpuByGroup
- ResPeCpuByGroup
- ResScpuDayTotal
- ResScpuSec
- ResSvprDetailReadTotal
- ResSvprPreReadBySec

- ResSvprQLenAvg
- ResSvprQLenAvgByVproc
- ResSvprQLenAvgByVproc
- ResSvprQLenMaxHour
- ResSvprReadByVprocSec
- ResSvprReadByVprocSec
- ResSvprReadTotal
- ResSvprReadTotal
- ResSvprWriteTotal
- ResSvprWriteTotalHour
- ResSyncScan