

tableBASE

Administration Guide

Release 7.0.3

Copyright © 2026 DataKinetics Ltd.

Document Number: TBM002-R703v1.0

Publication History:

tableBASE Release 7.0.3 Version 1.0 - July 2026

This guide is copyrighted and all rights are reserved. This document may not, in whole or in part, be copied, photocopied, reproduced, translated or reduced to any electronic medium or machine-readable form without the prior written consent of DataKinetics Ltd.

Information in this guide is subject to change without notice and does not represent a commitment on the part of the vendor. The software described in this guide is furnished under a license agreement and may be used or copied only in accordance with the terms of the agreement.

tableBASE, tablesONLINE, VTS Manager and VTS Gate are registered trademarks of DataKinetics Ltd. The names of other products or companies may be trademarks or registered trademarks of their respective companies.

DataKinetics Technical Support: 1-613-523-5588

Email: tableBASE@dkl.com

DataKinetics Ltd.

110 Didsbury Rd.

Kanata, ON

Canada K2T 0C2

<http://www.dkl.com>

Telephone: (613) 523-5500
1-800-267-0730 (toll free in the US and Canada)

Contents

List of Figures.....	xi
List of Tables.....	xiii
Preface.....	xv
Audience for this Guide.....	xv
What is Covered in this Guide.....	xv
Naming protocol.....	xv
What You Should Know to Use this Guide.....	xvi
Glossary.....	xvi
Conventions used in this guide.....	xviii
What This Guide Contains.....	xix
Additional tableBASE references.....	xx
Customer Support.....	xxi
Training and Professional Services.....	xxi
About DataKinetics.....	xxi
 Chapter 1: Installation Modifications.....	 1
 Chapter 2: Master Password Feature.....	 5
 Chapter 3: tableSPACE Reporting.....	 7
Generating the tableSPACE report for batch, IMS and VTS Agent.....	8
Generating the tableSPACE report for CICS.....	8
tableSPACE report samples and description.....	9
 Chapter 4: tableBASE User Exits.....	 15
User Exits.....	15
Working with User Exits.....	16
Step 1—Programming User Exits.....	16
Step 2—Defining Exits using DK1TX072.....	17
Step 3—JCL Requirements for Exit Processing.....	17
Step 4—TBOPT Parameter Requirement.....	17
DK1EXDFN Macro Parameters.....	17
Defining a Header (TYPE=INITIAL).....	18
Defining the Entries (TYPE=ENTRY).....	21
Parameter Data Passed to Exits.....	23
Parameter 1: Communication Area.....	23
Parameter 2: Global Work Area.....	29
Parameter 3: Shared Thread Related Work Area.....	29
Parameter 4: Private Thread Related Work Area.....	30

Exit Related Messages.....	30
Sample User Exits.....	30
Chapter 5: Program Call Server.....	33
Initialization of the Program Call Server.....	33
Server Messages.....	34
Licensing Messages.....	34
Re-initialization of the Program Call Server.....	34
Restarting the tableBASE Program Call Server.....	34
Shutting down the Program Call Server.....	35
Chapter 6: DB2 Stored Procedures.....	37
Using DB2 stored procedures.....	37
Chapter 7: IMS.....	39
JCL.....	39
Pre-loading tableBASE modules.....	39
Application/transaction initialization.....	40
Transaction abends.....	41
VTS-TSR access.....	41
Chapter 8: CICS.....	43
JCL.....	43
Definitions.....	43
Journal.....	43
Programs.....	44
Transactions.....	44
CICS initialization and shutdown.....	44
Transaction TBST.....	44
Restrictions.....	45
Transaction TBDR.....	45
The TBOPT dataset.....	45
Chapter 9: VTS (Virtual Table Share).....	47
Overview.....	47
New Features in Version 7.....	48
Changes from Version 6.....	48
Update commands in a VTS-TSR.....	48
Transaction affinities.....	48
TBPOSTV.....	49
TBOPTV.....	49
SWITCHES parameter.....	49
VTS Agent operations.....	49
Startup.....	49

Initialization of a VTS-TSR.....	50
TBOPT parameters.....	50
VTS-TSR table loading sample JCL.....	51
VTS-TSR refresh process.....	51
VTS Agent shutdown.....	52
VTS Agent recovery.....	52
VTS commands.....	52
VTS operational considerations.....	53
Reducing virtual storage when moving tables into VTS.....	53
Linked tables in VTS.....	53
7/24 VTS.....	54
Capacity.....	54
Loading a VTS-TSR.....	54
VTS messages.....	54
Internal logic errors.....	55
VTS user messages.....	55
Chapter 10: VTSGate (VTS with Enhanced Security).....	57
Overview.....	57
VTS management.....	57
Valid combinations of VTSGate security features.....	57
tableBASE commands by TSR access type.....	58
TBOPT parameters.....	58
VTSDATA.....	58
Access control setup for SAF protected VTS.....	59
Startup and shutdown.....	60
Table loading and updating for read-only VTS.....	61
Strobe reporting.....	61
Licensing Messages.....	62
Chapter 11: VTSManger.....	63
VTSManger Features.....	63
Pre-loaded VTS-TSRs.....	63
Read-only VTS.....	63
Refresh-only VTS.....	63
Update Tables in the Background.....	64
Switching VTS-TSRs.....	64
Alias Names for VTS-TSRs.....	64
Installing VTSManger.....	64
Details of VTSManger Installation.....	64
Administration of VTSManger.....	65
VTSManger Administrative Tasks.....	65
VTS-TSR Access Modes.....	65
Pre-loading VTS-TSR.....	65
Sample JCL to bring up VTS-TSR in RF Mode.....	65
DK1MEXEC Utility.....	66

VTSTManager Command Summary.....	66
DK1MEXEC.....	66
Things to consider while performing VTS Manager SWITCH.....	71
JCL Examples Delivered with VTSTManager.....	72
Chapter 12: tablesONLINE/CICS Architecture.....	73
Overview.....	73
Library architecture.....	73
System library.....	74
Application Control Table library.....	74
Application-Driving Tables library.....	74
Data Dictionary library.....	74
Levels of privilege.....	75
Types of data.....	75
Common corporate data.....	75
Application-specific user data.....	76
Application-driving data.....	76
Chapter 13: tablesONLINE/CICS Security.....	79
The signon security exit.....	79
The Cross Authorization table.....	80
The User/Application Relationship table.....	80
The Application Control table.....	82
Invoking signon security.....	82
Chapter 14: tablesONLINE/CICS Administration.....	83
Set up new users.....	84
Application Control table.....	84
Modifying an existing Application Control table entry.....	90
Editing the User/Application Relationship table.....	91
Cross-Authorization table.....	94
tablesONLINE/CICS and VTS.....	95
Creating new applications.....	97
Copying applications.....	98
Developing applications.....	99
Removing a User-developed application.....	100
Customizing your environment.....	100
User Profile.....	101
Editing system help.....	108
Editing system tutorials.....	112
System constants.....	113
Editing MRO transaction IDs.....	115
Monitoring your system.....	117
Session control.....	118
Session suspension and restoration.....	118

Terminating a session.....	118
Terminating the administrator session.....	120
Transferring to user applications.....	121
Use of temp tables and TS queues.....	122
Converting from Release 4.2.x.....	123
Chapter 15: Best Practices.....	125
Protecting tableBASE libraries.....	125
Concurrent multiple updates.....	125
Read-Only libraries.....	125
Protecting tableBASE tables.....	126
Table read and write passwords.....	126
TSR LOCK-LATCH.....	126
Time out while waiting for a lock.....	126
tablesONLINE temporary tables.....	126
Performance.....	127
Shared tables: Virtual Table Share (VTS).....	127
Linking tableBASE.....	127
Using the TB-PARM Communication Area.....	127
WLM priority level consideration.....	127
Application performance with TBASEV or TBCALLV.....	127
Batch/IMS Performance Degradation with Dynamic Calls to tableBASE.....	128
Table access optimization.....	129
Multitasking.....	129
tableBASE library definition considerations.....	129
Library Blocksize.....	129
tableBASE library space usage.....	130
BDAM/QSAM tableBASE libraries.....	130
VSAM tableBASE libraries.....	130
VSAM tableBASE libraries and LSR pools.....	130
Enhanced Data Integrity option.....	131
Library directory caching.....	131
TSR Space Allocation.....	132
tableBASE modules that can reside in the Link Pack Area (LPA).....	132
Chapter 16: tableBASE Diagnostic Information.....	133
tableBASE use of MVS Enqueues.....	133
tableBASE Library Diagnostics.....	134
Appendix A: tableBASE Runtime Options.....	135
ALLOW_TB_LIB_IO—Allow tableBASE Library I/O.....	135
CICSJRNL—CICS Journal File ID.....	135
DATERTNX—Date-Sensitive Processing Found Code.....	135
FGDELIM—Fetch Generic Delimiter.....	136
HASH_HI_DEN_LIM—High Density Limit for Hash Indexes.....	136

HASH_LOW_DEN_LIM—Low Density Limit for Hash Indexes.....	136
LDS—Linear Data Set Use.....	137
LIBnn, ML—tableBASE Library List.....	137
LISTOPTIONS—List Parameter Options.....	137
LOCKTIMER—Lock Timer Wait Value.....	138
LOCKTIMEWTO—Lock Timer Message Wait Value.....	138
MAXNMTAB—Maximum Number of Tables.....	138
MTRETAIN—Retain Rows and Index Areas.....	139
MULTITASKING—Multitasking.....	139
MULTOPNX—Multiple Alternate Index.....	140
OVERRIDES—Allow Changes to Status Switches.....	140
RACF_LIBACCESS—Perform Checks on Protected Libraries.....	141
RACF_VTS—Perform Checks on SAF Interface Protected VTS-TSRs.....	141
STROBE—Strobe Interval.....	142
STROBEMETHOD—Strobe Method.....	142
SUPPRESS_DUMPS—Suppress tableBASE Messages and Dumps.....	143
SWITCHES—Status Switches.....	144
TABLEWAITRC—Table Open Enqueue Wait Time.....	145
TABLEWAITWTO—Table Open Enqueue Report Time.....	146
TPVM—VTS Group Manager Use.....	146
TSR_ALGORITHM—Optimize TSR Usage.....	146
TSR_KEY_PROTECT—Hardware Key Protection for VTS-TSRs.....	147
TSR_WARNING_FREQ—Frequency of TSR Allocation Warnings.....	147
TSR_WARNING_PCT—Percentage of TSR Allocation for Warning Activation.....	148
TSRACCESS—Read Only (RO), Refreshable (RF), and Read Write (RW) TSR Access.....	148
TSRSIZE—tableSPACE Region Size.....	148
USEREXITS—User Exits.....	149
VTSFIRST, VTSLAST—VTS Search Sequence.....	149
VTSNAME—Specifying the Name of a VTS-TSR.....	150
VTSPREFIX—VTS Prefix Definition.....	151
ZEROROWS—Zero Data Table Rows on De-allocation.....	151
Appendix B: Setting up an organization.....	153
Defining authorities.....	153
Defining the environment.....	154
Defining users and applications.....	155
Appendix C: tableBASE messages.....	161
tableBASE return codes.....	161
tableBASE messages.....	178
tablesONLINE/CICS error messages.....	194
tablesONLINE/ISPF error messages.....	218
DK1TEXEC error messages.....	230
VTSManager Error messages.....	237
Command Interpretation Errors.....	237

Command Processing Errors.....	238
Licensing Warnings and Errors.....	239
Appendix D: PC Server implementation in a CICS environment.....	243
Implementation and Restart/Recovery Considerations.....	243
Steps to Implement the PC Server.....	244
RESTART/RECOVERY Considerations.....	245
NOTES and CAUTIONS.....	246
Appendix E: VTS Gate and VTSManger Compatibility.....	247
Scope.....	247
VTSManger with RACF protection from VTS Gate.....	247
Description of the VV Command.....	247
Compatibility Consideration: VV and TSRACCESS=RF.....	247
VTS Configurations with LDS Backing and VTS Manager Alias Switching.....	248
TSRACCESS=RF.....	248
TSRACCESS=RO.....	248
TSRACCESS=RW.....	248
VTS Configurations without LDS Backing and VTS Manager Alias Switching.....	249
Important Context.....	249
Read-Refresh Access (TSRACCESS=RF).....	249
Read-Only Access (TSRACCESS=RO).....	249
Read-Write Access (TSRACCESS=RW).....	250

List of Figures

Figure 3-1: Sample JCL for batch, IMS, VTS Agent tableSPACE report generation.....	8
Figure 3-2: Sample JCL for CICS tableSPACE report generation.....	9
Figure 3-3: Sample strobe report: local TSR (STROBE=5).....	10
Figure 3-4: Sample strobe report: VTS-TSR (STROBEMETHOD=3, STROBE=5).....	11
Figure 4-1: User exits overview.....	23
Figure 4-2: Extended Field Definition Area Layout.....	28
Figure 4-3: Sample implementation of DX1EXDFN in DK1TX072.....	32
Figure 9-1: Sample VTS Agent Startup JCL.....	49
Figure 9-2: Sample JCL for loading tables.....	51
Figure 9-3: Sample JCL for VTS refresh.....	52
Figure 12-1: Signon and Application Control relationships.....	77
Figure 13-1: Signon identifiers.....	80
Figure 14-1: Administrator menu.....	83
Figure 14-2: Edit Application Control table.....	85
Figure 14-3: Edit Application Control table—Edit Row.....	86
Figure 14-4: Edit User/Application Relationship table.....	92
Figure 14-5: Edit User/Application Relationship table—Edit Row.....	92
Figure 14-6: Edit Cross-Authorization table.....	94
Figure 14-7: Edit Cross-Authorization table—edit row.....	95
Figure 14-8: Edit the TBSYSVTS table.....	96
Figure 14-9: Sample JCL code for initializing the DK1S VTS-TSR.....	97
Figure 14-10: Copy Application Control tables.....	98
Figure 14-11: Application Developer's menu.....	99
Figure 14-12: Edit screen—Identify Table/Row.....	101
Figure 14-13: Default entries for the User Profile table.....	104

Figure 14-14: Help/Tutorial menu.....	109
Figure 14-15: Application's Help/Tutorial sample row.....	110
Figure 14-16: Application's Help/Tutorial—indirect help.....	111
Figure 14-17: Application's Help/Tutorial—Edit Row screen.....	112
Figure 14-18: Editing system constants.....	114
Figure 14-19: System constants—default row.....	115
Figure 14-20: Edit—Edit MRO transaction IDs.....	116
Figure 14-21: MRO Transaction IDs—default row.....	116
Figure 14-22: Browse/Delete Active Users screen.....	119
Figure 14-23: Example of how to delete a user session.....	119
Figure 14-24: Entering the LOCK-LATCH of table that was left open.....	120
Figure 14-25: Place low values.....	120
Figure 14-26: Locate entry for deletion.....	121
Figure 14-27: Delete the row.....	121
Figure 14-28: User Applications transfer to screen.....	122
Figure 14-29: JCL for manually converting VIEWS.....	124

List of Tables

Table 1-1: Default option overrides.....	1
Table 3-1: tableSPACE report field definitions.....	11
Table 4-1: Comparing Old and New User Exits.....	15
Table 4-2: Exit Points.....	22
Table 4-3: Environment Indicator.....	24
Table 4-4: Exit Indicator.....	24
Table 4-5: Status Indicator.....	25
Table 4-6: Access Flag.....	29
Table 4-7: Exit Related Messages.....	30
Table 7-1: tableBASE modules.....	40
Table 10-1: SAF protected VTS Access Levels and Permissions.....	59
Table 11-1: VTSManger Command Summary.....	66
Table 11-2: LICENSE.....	68
Table 11-3: RESET.....	68
Table 11-4: SET.....	69
Table 11-5: SHOW.....	69
Table 11-6: SWITCH.....	69
Table 12-1: Privilege levels.....	75
Table 13-1: Security exit programs.....	79
Table 13-2: User/Application Relationship entries.....	81
Table 13-3: Installation entry.....	81
Table 13-4: Post-installation entries.....	81
Table 14-1: Security settings.....	89
Table 14-2: Command line exit names.....	90
Table 14-3: Field explanation—user/applications.....	93

Table 14-4: All possible entries—User/Application Relationship table.....	93
Table 14-5: Field explanation—Cross-Authorization table.....	95
Table 14-6: Copy application-driving tables.....	98
Table 14-7: Edit Row screen field explanation.....	105
Table 14-8: Application's Help/Tutorial field explanations.....	110
Table 14-9: Application's Help/Tutorial—ID/Function screen values.....	112
Table 14-10: MRO transaction IDs—sample table.....	117
Table A-1: Overrides default values.....	140
Table A-2: Switches default values.....	144
Table B-1: Application IDs and hypothetical organization.....	154
Table B-2: Application Control table.....	155
Table B-3: User/Application Relationship table (TBOLUSAP) for hypothetical organizations.....	157
Table B-4: Cross-Authorization table (TBOLXAUT) for hypothetical organizations.....	158
Table C-1: tableBASE messages and error codes.....	162
Table C-2: Messages and error codes for VTS Agent, CICS, Batch, etc.....	179
Table C-3: tablesONLINE / CICS messages and error codes.....	194
Table C-4: tablesONLINE/ ISPF messages and error codes.....	218
Table C-5: tableBASE batch utility messages.....	230
Table C-6: Command Interpretation messages.....	237
Table C-7: Command Processing Messages.....	238
Table C-8: License-Related Messages.....	240

Preface

This guide provides the information and tools for the tableBASE administrator. It provides general information that relates to most environments, including CICS-specific administration information, the optional DataKinetics Virtual Table Share (VTS) product and the optional VTS Gate product, and the optional VTSManger product that is available with Release 7.0.3.

Note: Keep this guide in a secure location. The information provided could allow an individual to override normal tableBASE software security provisions. The guide should remain in the possession of the tableBASE administrator or the individual who is authorized to perform administrative functions.

Audience for this Guide

This guide is intended for individuals responsible for the administration of the tableBASE software.

The responsibilities of tableBASE administrator may include the following tasks:

- Act as the single point of contact for DataKinetics Technical Support.
- Install and configure tableBASE components.
- Coordinate implementation of tableBASE upgrades.
- Maintain policies and procedures for tableBASE administration and use.
- Set up users and user authorizations.
- Define tableBASE libraries.
- Maintain company-wide table libraries.
- Maintain security rules for tableBASE libraries.
- Define and control naming standards for tableBASE tables.
- Generate and maintain the tableBASE Master Password.
- Assist system support with tableBASE start-up jobs for all online regions.

What is Covered in this Guide

This guide provides the background information needed by the tableBASE administrator to set up and maintain a suitable working environment for the DataKinetics tableBASE product and any additional tableBASE-related DataKinetics software products.

Naming protocol

The new tableBASE naming protocol introduced with Version 6 remains. All tableBASE executables begin with DK1 for easy identification, a prefix that has been reserved for exclusive use with IBM.

Aliases are retained so that no changes are required to your existing applications.

Modules should be used to entirely replace previous release modules.

What You Should Know to Use this Guide

A working knowledge of a programming language (for example, COBOL or Assembler), compiler procedures, a linkage editor, and the MVS JCL systems is required to perform the tasks detailed in this guide. The Administrator should be familiar with the tableBASE concepts and facilities and the MVS environment.

If your installation includes tablesONLINE/CICS, you should also be familiar with CICS and the content of the *tablesONLINE/CICS User's Guide*.

Glossary

The terms defined in this glossary are used frequently in this document.

Alias name	An alternate name for a VTS-TSR that can be created, assigned, and used in lieu of the name assigned at VTS-TSR definition. This is a VTSManger feature. See below for VTSManger and VTS-TSR definition.
Alternate Index	An Alternate Index is an Index that may be defined for a Data Table. The Alternate Index has an Alternate Index definition (ALT-DEFINITION) that defines the key, organization, and search order. Alternate Indexes are optional, and there is no limit to the number of Alternate Indexes a Data Table may have.
Alternate Index View	A tablesONLINE Alternate Index View is identical to a View but applies to a Data Table when access is through an Alternate Index.
Data Table	A Data Table is the actual raw data. Each Data Table has a table definition (DT-BLOCK) that is used to generate the Index for the Data Table.
DB2 SPAS, WLM-managed DB2 SPAS	DB2 Stored Procedure Address Space managed by WLM.
Delivered defaults	The defaults that are delivered with the product. Also known as <i>factory defaults</i> .
ETL	The Extract Transform and Load Batch utility program (DK1ETL) Extracts data from the original DB2 or VSAM source, transforms it according to control card criteria, and loads it into tableBASE libraries and tables.

Index	An Index is defined for each Data Table. A Data Table Index is generated dynamically when a table is opened or defined based on the information in the table definition (DT-BLOCK).
Installation defaults	The defaults set at installation time by an administrator, which may or may not be the same as the delivered defaults. Defined using the TBOPTGEN file. (These defaults may be overridden by an individual application using the TBOPT file.)
LDS	Linear DataSet used for VTSManger.
Linked Table	A linked table (also known as a remote table) is created when a user issues a command to open a table that is already open in a VTS-TSR specified in the LIB-LIST. The table entry in the local TSR is linked to the existing open table in the VTS-TSR. No updates are allowed to a linked table.
Multitasking Batch	An MVS region that implements multitasking by attaching multiple Task Control Blocks (TCBs). This can include a batch job that attaches several subtasks or a transaction processing region like DB2 stored procedures that implements multitasking through multiple TCBs.
PC Server	The PC Server allows tableBASE program calls (PCs) to be available for tableBASE regions. The PC functionality supported by the PC Server provides the functions needed for tableBASE multi-tasking, for CICS thread-safe support, for accessing a VTS-TSR, and for compatibility with DB2 stored procedures, as well as other special functions that are required by tableBASE.
SAF interface	System Authorization Facility interface that controls resource access. Tools like ACF2®, RACF®, and Top Secret® fall into this category.
SPACEMAP	The name of the memory allocator used and manage tableBASE TSRs.
Table Expansion	Dynamic allocation of space for tables in the TSR when the current space allocated becomes insufficient.
Temporary Table	A temporary table exists only within a TSR, and is created by the DT command (or IA). It is never stored in a library. A temporary table can be distinguished from a library table using the GD command output—if found, a temporary table will show no dataset name.
TSR, Local TSR	Table Space Region. A data space of up to 2G is used by tableBASE to house tables. The data space is owned by an application in the associated address space. The application uses tableBASE to access data within the tables.

TSR of record	If there is no VTS name in the TB-SUBSYSTEM field of the TB-PARM when a command is being executed, the TSR of record is the local TSR, otherwise it is the VTS-TSR that has its name in the TB-PARM.
View	A tablesONLINE View provides the field, edit and display attributes for a Data Table with its Index. Previous to Version 5, the View was referred to as a Field Definition Table (FDT).
VTS, DataKinetics VTS	Virtual Table Share. The term "VTS" refers to the DataKinetics VTS product, which permits TSRs to be shared among applications. These shared TSRs are called VTS-TSRs.
VTS-TSR, Shared TSR	A Virtual Table Share (VTS) Table Space Region (TSR) is a shared TSR, and resides in a shared data space. Applications can access tables within a VTS-TSR, and use the information as if it were within their local TSRs.
VTS Agent	The DK1VAGNT program, which initializes VTS-TSRs in tableBASE, and then sits idle until the VTS-TSR is to be terminated.
VTSGate	The product that provides enhanced security for the shared VTS-TSR capability.
VTSTManager	A DataKinetics Ltd. product that extends the functionality of the VTS shared TSR capabilities.
VTS switch	The action of switching an alias name from one VTS-TSR to another. This is a feature of VTSTManager.

Conventions used in this guide

This guide uses conventions to differentiate code and typed commands, to display the names of parameters, and to describe specific version and release levels of the tableBASE product.

Convention	Description
code examples and commands	Code examples and commands appear in this type of font: <code>this is an example of the font.</code>
MAXNMTAB	Names of parameters appear in upper case simply for ease of reading; actual case used is upper or lower or a mixture.

Convention	Description
Version	Following IBM standards, the term <i>version</i> refers to a generation of a software product that has significant new code or new functionality. <i>Version</i> is a more general term than <i>release</i> . For example, <i>Version 7</i> includes <i>Release 7.1</i> and <i>Release 7.2</i> , and is equivalent to <i>Release 7.x</i> .
Release	Following IBM standards, the term <i>release</i> refers to a program or set of programs which represent a specific revision to the base version of a software product. For example, <i>Release 7.0</i> is a term that is used to identify the first release of <i>Version 6</i> . Subsequent releases made available under the Version 6 umbrella, such as <i>Release 7.1</i> , will provide additional revisions to the base product.
Modification Level	Following IBM standards, the term <i>modification level</i> refers to the application of specific program enhancements and error corrections to the release of a software product. For example, <i>Release 7.0.3</i> is at <i>modification level 3</i> , and <i>Release 7.1.0</i> is at <i>modification level 0</i> .
MVS	MVS is a generic term which is used when referring to z/OS and other related IBM operating systems.

What This Guide Contains

Chapter 1 presents the default values for tableBASE parameters and explains how to change them if desired.

Chapter 2 shows how the Master Password for tableBASE tables is generated and used.

Chapter 3 demonstrates how to use the tableSPACE Reporting feature to generate usage statistics for tableBASE.

Chapter 4 explains how tableBASE users can write their own user exits.

Chapter 5 presents the Program Call Server feature of tableBASE, explains what it does, and shows the messages it will generate in a variety of circumstances.

Chapter 6 shows how to access tableBASE tables using DB2 Stored Procedures.

Chapter 7 provides information on how to tailor the IMS environment to accommodate tableBASE Version 7.0.3.

Chapter 8 provides information on how to tailor the CICS environment to accommodate tableBASE Version 7.0.3.

Chapter 9 discusses the administrative issues that arise when using DataKinetics VTS (Virtual Table Share) product with tableBASE.

Chapter 10 provides information on how to set up and maintain VTS-TSRs using the security features provided with the VTS Gate interface.

Chapter 11 provides information on how to set up and maintain VTS-TSRs using a backed VSAM LDS with the VTSManger.

Chapter 12 lays out a detailed overview of the architecture of the DataKinetics tablesONLINE/CICS software product.

Chapter 13 covers security considerations for the tablesONLINE/CICS software product.

Chapter 14 shows how to use tablesONLINE/CICS to assist with the administration of your system when setting up new users, creating new applications, customizing your environment, and monitoring your system.

Chapter 15 contains information on optimizing and organizing the tableBASE environment that should be of interest to tableBASE administrators.

Chapter 16 identifies and describes the tableBASE diagnostics tools for the tableBASE administrator.

Appendix A lists and explains the tableBASE run-time option parameters.

Appendix B describes the initial set up of a tablesONLINE/CICS environment for a hypothetical organization.

Appendix C contains all of the DataKinetics tableBASE error codes and messages that can be encountered during the normal installation, administration and operation of the product.

Appendix D discusses the implementation and restart/recovery and other considerations of using the tableBASE PC Server in a CICS environment.

Appendix E discusses how to implement and use VTSGate and VTSManger together.

Additional tableBASE references

This guide is one of a series that describes tableBASE and tablesONLINE:

- *tableBASE Release Notes*
- *tableBASE Installation Guide*
- *tableBASE Concepts and Facilities Guide*
- *tableBASE Batch Utilities Guide*
- *tableBASE Administration Guide*
- *tableBASE Programming Guide*
- *tableBASE Library Bridge Manual, Release 5.B*
- *tableBASE Quick Reference Guide*
- *tableBASE RestConnector Guide*
- *tablesONLINE/CICS User's Guide*
- *tablesONLINE/ISPF User's Guide*

Customer Support

Customers with maintenance agreements can obtain 24-hour hotline support through a direct line to our Technical Support staff (+1-613-523-5588).

The DataKinetics customer support Service Level Agreement lists the guaranteed response times for each problem severity level. High priority critical and serious problems receive immediate attention from our Technical Support staff.

Before calling the DataKinetics Technical Support staff, be sure to consult the "Support" section of our web site (www.dkl.com) for all the latest information on our technical support protocol. Look in "Support" for our "FAQs", which provide answers to frequently asked questions posed by our users.

The "Customer Login" link supplied on our web site navigates to a special support section which is available to customers who have maintenance agreements with DataKinetics Ltd. A username and password are needed to access this section of the web site—consult your tableBASE administrator, or DataKinetics customer support, for this information.

Training and Professional Services

DataKinetics Ltd. is proud to be able to offer our customers the very best in training and technical consulting services. See the "Services" section of our web site for all the latest information regarding our offerings in this area.

About DataKinetics

DataKinetics®, the leader in enterprise IT optimization, enables the top data centers in the world to overcome today's mainframe throughput, performance, capacity and interconnectivity challenges. In business for over 40 years, DataKinetics products are powering nearly a billion mission-critical transactions per second every day for the world's largest banking, insurance, credit card, brokerage house and retail organizations.

Chapter 1

Installation Modifications

tableBASE has factory defaults; these can be modified for your site at installation using the TBOPTGEN file. Applications can further override options on an individual basis, using the TBOPT file. [Table 1-1](#) on page 1 lists all of the applicable parameters, provides a brief description, and shows default values. For a complete description of all options, see [tableBASE Runtime Options](#) on page 135. For further information see *tableBASE Installation Guide*.

Parameter	Description	Default †
ALLOW_TB_LIB_IO	Allow tableBASE Library I/O - Allows access to tableBASE libraries.	Batch, CICS and IMS interfaces: Y; DB2 SP interface: N
CICSJRN	CICS Journal File ID (CICS only) - Specifies the CICS journal file number if strobe records are written.	99
DATERTNX	Date-Sensitive Processing Found Code - Specifies whether X is to be returned as a Found Code.	N
FGDELIM	Fetch Generic Delimiter - Specifies the wildcard value used for the Fetch Generic command.	*
HASH_HI_DEN_LIM	High Density Limit for Hash Indexes - Upper density limit for hash tables opened starting with Version 6.	900
HASH_LOW_DEN_LIM	Low Density Limit for Hash Indexes - Lower density limit for hash tables opened starting with Version 6.	600
LDS	Linear Data Set Use - Specifies whether the VTS-TSR defined in this jobstep will be mapped to a LDS. VTS Manager only.	N
LIBnn, ML	tableBASE Library List - Default library names for a tableBASE Library List (LIB-LIST).	LIB00=MAINLIB
LISTOPTIONS	List Parameter Options - Specifies whether to list execution time parameters.	N
LOCKTIMERC	Lock Timer Wait Value - Specifies # of seconds that tableBASE should wait for a lock (0= wait indefinitely).	0

Parameter	Description	Default †
LOCKTIMEWTO	Lock Timer Message Wait Value - Number of seconds between messages indicating that the transaction is waiting for an internal lock (0=no message).	30
MAXNMTAB	Maximum Number of Tables - Specifies the maximum number of open tables allowed in the TSR. VTSNMTAB is maintained for backwards compatibility.	Depends on TSR size - see <i>Programming Guide</i> for details
MTRETAIN	Retain Rows and Index Areas - Determines whether the allocated row and index areas are to be retained when an MT command is issued.	N
MULTITASKING	Multitasking Option (Batch only) - Specifies whether this region will allow multiple subtasks to access tableBASE.	N
MULTOPNX	Multiple Alternate Index - This option is maintained for compatibility with previous releases, and is not used starting with tableBASE Version 6.	Y
OVERRIDES	Allow Changes To Status Switches - Specifies whether the user can change specific Status Switches at run time. Positional (see below) Y or N settings.	YYYYY batch, IMS, DB2 YNNYY CICS
RACF_LIBACCESS	Perform Checks on Protected Libraries - Specifies whether tableBASE should check if a tableBASE library is SAF interface protected.	N
RACF_VTS	Perform Checks on SAF Interface Protected VTS-TSRs - Specifies whether tableBASE should check if a VTS-TSR is SAF interface protected. Available only with VTS Gate.	N
STROBE	Strobe Interval - Specifies frequency of strobe records (in # of tableBASE calls or minute intervals).	0
STROBEMETHOD	Strobe Method - Specifies when tableBASE strobe reporting should be triggered for the VTS-TSR. 1. 0, 1, 2 indicates that strobe should be triggered by number of tableBASE calls indicated by the STROBE option 2. 3 indicates that strobe should be triggered by minute intervals indicated by the STROBE option	0
SUPPRESS_DUMPS	Suppress tableBASE messages and dumps - Specifies whether to suppress tableBASE messages and dumps when a transaction abends outside tableBASE code. Only for the CICS environment.	NONE

Parameter	Description	Default †
SWITCHES	Status Switches - Specifies status switch defaults (Y or N): - whether to abend on tableBASE user errors (1-99; 1001-1099) - whether to wait for tables that are opened for write - whether to return empty rows from hash tables - whether to automatically open tables on read accesses - whether to trace last 10 tableBASE commands	YNYYN for batch/CICS NNYYN for DB2/IMS
TABLEWAITRC	Table Open Enqueue Wait Time - Number of seconds tableBASE should wait for exclusive table enqueue (applies only if WAIT switch is on).	0
TABLEWAITWTO	Table Open Enqueue Report Time - Number of seconds between messages indicating that the transaction is waiting for an enqueue.	0
TPVM	VTs Group Manager use - Specifies a VTs Group Manager (TPVM) name to use for all accesses to VTs-TSRs.	compat
TSR_ALGORITHM	Optimize TSR Usage - Determines whether the Performance (P), or Default (D) attributes are predominant when allocating space in a TSR.	D
TSR_KEY_PROTECT	Hardware Key Protection for VTs-TSRs - Specifies whether a VTs-TSR is hardware key protected. Available only with VTs Gate. Any value from 1 to 7 indicates hardware key protection is on.	0
TSR_WARNING_FREQ	Frequency of TSR Allocation Warnings - Number of seconds between messages indicating the TSR is more than TSR_WARNING_PCT allocated.	0000
TSR_WARNING_PCT	Percentage of TSR Allocation for Warning Activation - Indicates percentage of TSR allocation allowed before a message is generated.	85%
TSRACCESS	Read Only or Read Write TSR Access - Specifies whether a VTs-TSR is Read-Only or Read/Write. Available only with VTs Gate.	RW
TSRSIZE	tableSPACE Region Size - Specifies the size of Data Space to be allocated for open tables for a region. TSREGION and TSR are maintained for backwards compatibility.	10M
USEREXITS	User Exits - Specifies whether User Exits are used (Y N).	N
VTsFIRST	VTs Search Sequence - VTs-TSR to search before LIB-LIST (VTs only).	(null strings)
VTsLAST	VTs Search Sequence - VTs-TSR to search after LIB-LIST (VTs only).	(null strings)

Parameter	Description	Default †
VTSTNAME	Specifying the Name of a VTS-TSR 1. Specifies the name of a Shared TSR (VTS-TSR). 2. Specifies the VTS-TSR name used by TBASE/TBCALLV interfaces.	1. DKL1 2. —
VTSPREFIX	VTS Prefix Definition - Specifies a prefix that will be used by tableBASE to denote a VTS name within the ML list. There can be 0-3 characters before the colon; the colon is mandatory.	VTS:
ZEROROWS	Zero Data Table Rows on De-allocation - Determines whether the data rows area should be zeroed when table space in a TSR is deallocated.	Y

Table 1-1: Default option overrides

† The default settings listed here are delivered defaults of tableBASE from DataKinetics. Your site may have set the installation defaults differently.

Chapter

2

Master Password Feature

The tableBASE tables can be protected on the library (read and write passwords) and when loaded in memory (LOCK-LATCH). It is important to note that these passwords offer only limited protection against unauthorized use of tableBASE tables.

In the event that a table password (read, write or LOCK-LATCH) is lost or forgotten, tableBASE provides a safety mechanism that may be used by the tableBASE administrator. The table may be accessed by supplying a master password in place of the normal read or write password or in place of the LOCK-LATCH if the table is open in-memory. Closing a table using the master password as the LOCK-LATCH releases the LOCK-LATCH.

Please contact tableBASE Customer Support for more information on how to obtain the master password at tableBASE@dkl.com

Chapter

3

tableSPACE Reporting

tableBASE can produce a report summarizing your use of tableBASE. This report, known as the tableSPACE Report, includes such information as the space occupied by each table in the tableSPACE Region (TSR), the number of accesses for each table, and much more. These statistics provide information on usage during the life of the region and can be used to optimize performance in all environments.

You control the frequency with which this information is collected. For a local TSR, this is done using the STROBE parameter. For example, if you specify a strobe interval of 10,000, the statistics collection routine is called after every 10,000 calls to tableBASE. The tableSPACE report can be printed on demand or when tableBASE terminates.

For a VTS-TSR, the STROBEMETHOD parameter is also required. A value of 0, 1 or 2 for STROBEMETHOD means that the STROBE parameter still represents the number of calls after which to collect tableBASE usage statistics. A value of 3 for STROBEMETHOD means that STROBE now represents the number of minute intervals after which to collect usage statistics. The default value for STROBEMETHOD is 0.

For STROBEMETHOD of 0, 1 or 2, strobe processing has been improved so that tableBASE processing is prevented from waiting for the strobe report. If the number of calls in the STROBE parameter is set to a small value and tableBASE calls are made more frequently than strobe reports can be produced, the frequency of reports may change due to strobe reports being bypassed.

For tableBASE performance reasons, tableBASE strobe is now handled by a sub-task. Since the sub-task is running concurrently with tableBASE calls, the status of the TSR can change while the strobe data is being gathered. This may lead to apparent inconsistencies in the reports, such as the total number of tables open, and the actual number listed.

Generating the tableSPACE report for batch, IMS and VTS Agent

To generate a strobe report for the batch, IMS, or VTS Agent environments, add the following to your JCL. [Figure 3-1](#) on page 8 is a sample.

```
//* INSERT YOUR JOB CARD HERE
//*
//TBOPT DD DISP=SHR,DSN=*your.prefix*.tbopt_file
//*
//TBTSRPT DD SYSOUT=*
//*
```

Figure 3-1: Sample JCL for batch, IMS, VTS Agent tableSPACE report generation

[Figure 3-3](#) on page 10 and [Figure 3-4](#) on page 11 show examples of a local TSR and VTS-TSR report. [Table 3-1](#) on page 11 describes the definitions for the report fields.

Generating the tableSPACE report for CICS

tableSPACE usage statistics are generated and written to the CICS Journal File after every strobe interval if the run time options, STROBE and CICSJRNL are not set to zero. A tableSPACE report is then generated by executing the tableBASE utility program DK1TSTRB in batch. It reads the journalled data and produces a formatted tableSPACE report.

With this release, we assume that your CICS journal files are written to the MVS system logger. To activate the Strobe feature, you must understand the TBOPT mechanism and, in particular, the STROBE and CICSJRNL TBOPT run-time parameters.

For details on how to set up your CICS Journal File (CICSJRNL) and strobe interval (STROBE), see the *tableBASE Installation Guide*.

Member DK1TSTRB of the TBASE.CNTL library from the installation media contains a sample job to produce the tableSPACE report. Before executing the job, customize the job according to your installation standards. [Figure 3-2](#) on page 9 shows a sample for DK1TSTRB.

```

/* INSERT YOUR JOB CARD HERE
/*
/* NOTE: USE THIS JOB IF YOU USE CICS/TS
/* THIS JOB EXTRACTS INFORMATION FROM THE TABLEBASE/CICS JOURNAL
/* FILE AND PRODUCES THE TABLESPACE REPORT
/*
/* CHANGE THE PARAMETERS BELOW AS REQUIRED
/*
/* EXTRACT STROBE RECORDS AND PRODUCE STROBE REPORTS
/*
//TBSTROBE EXEC PGM=DK1TSTRB
//STEPLIB DD DSN=*YOUR.PREFIX*.TBASE.LOAD,DISP=SHR <=====
//SYSPRINT DD SYSOUT=*,RECFM=FBA
//SYSUDUMP DD SYSOUT=*
//TBDUMP DD SYSOUT=*
//TBLOG DD DSN=*YOUR.JOURNAL.LOGSTREAM.NAME*, <=====
// BLKSIZE=32760,
// SUBSYS=(LOGR,DFHLGCVN)
//TBREPORT DD SYSOUT=*
/*

```

Figure 3-2: Sample JCL for CICS tableSPACE report generation

tableSPACE report samples and description

[Figure 3-3](#) on page 10 is a sample of a tableSPACE report for a local TSR. The report can be generated for any number of strobe cycles; this example shows three strobe cycles with STROBE set to 5). Note that each strobe shows the CALL COUNT value incremented by 5.

The report has a header and then fields that detail each strobe cycle. The information for each strobe cycle has the same format. The last section of the report provides details on each table that is open at the end of the strobe interval, its usage, and status in the TSR.

***** TOP OF DATA *****					
TABLEBASE 7.0.0 STROBE REPORT			PAGE: 1		
TABLE SPACE REGION NAME	LOCAL_DS	TSR REGION START	2017/07/07 15:06:26		
TSR SIZE	209,715,200	STROBE INTERVAL	5		
***** START OF STROBE NUMBER 1 *****					
TIME OF STROBE	2017/07/07 15:06:26	CALL COUNT	5		
TSR USED	552,960	TSR USED HWM	552,960		
LENGTH OF OPEN TABLES	77,824	LENGTH OF TABLES HWM	77,824		
NUMBER OF OPEN TABLES	1	NUMBER OF TABLES HWM	1		
***** O P E N T A B L E S *****					
TABLE NAME	ACCESS	LENGTH	READS	UPDATES	TOTAL ACCESSSES
TBDRTUTR	R	77,824	4	0	4
***** END OF STROBE NUMBER 1 *****					
***** START OF STROBE NUMBER 2 *****					
TIME OF STROBE	2017/07/07 15:06:27	CALL COUNT	10		
TSR USED	552,960	TSR USED HWM	552,960		
LENGTH OF OPEN TABLES	77,824	LENGTH OF TABLES HWM	77,824		
NUMBER OF OPEN TABLES	1	NUMBER OF TABLES HWM	1		
***** O P E N T A B L E S *****					
TABLE NAME	ACCESS	LENGTH	READS	UPDATES	TOTAL ACCESSSES
TBDRTUTR	R	77,824	9	0	9
***** END OF STROBE NUMBER 2 *****					
***** START OF STROBE NUMBER 3 *****					
TIME OF STROBE	2017/07/07 15:06:27	CALL COUNT	15		
TSR USED	552,960	TSR USED HWM	552,960		
LENGTH OF OPEN TABLES	77,824	LENGTH OF TABLES HWM	77,824		
NUMBER OF OPEN TABLES	1	NUMBER OF TABLES HWM	1		
***** O P E N T A B L E S *****					
TABLE NAME	ACCESS	LENGTH	READS	UPDATES	TOTAL ACCESSSES
TBDRTUTR	R	77,824	14	0	14
***** END OF STROBE NUMBER 3 *****					
***** END OF STROBE REPORT *****					

Figure 3-3: Sample strobe report: local TSR (STROBE=5)

[Figure 3-4](#) on page 11 is a sample of a tableSPACE report in minute intervals for a VTS-TSR using STROBEMETHOD=3. This example shows three strobe cycles with STROBE set to 5. Note that each strobe shows the TIME OF STROBE value incremented by 5 minutes.

The rest of the report is similar to that for a local TSR. [Table 3-1](#) on page 11 describes each field of the report.

TABLEBASE 7.0.0 STROBE REPORT				PAGE: 1	
TABLE SPACE REGION NAME	VTSKM3S5		TSR REGION START	2017/10/31 11:25:32	
TSR SIZE	10,485,760		STROBE INTERVAL	5	
***** START OF STROBE			NUMBER	1	*****
TIME OF STROBE	2017/10/31 11:30:32		CALL COUNT	26	
TSR USED	45,056		TSR USED HWM	45,056	
LENGTH OF OPEN TABLES	16,384		LENGTH OF TABLES HWM	16,384	
NUMBER OF OPEN TABLES	2		NUMBER OF TABLES HWM	2	
***** O P E N			T A B L E S	*****	
TABLE NAME	ACCESS	LENGTH	READS	UPDATES	TOTAL ACCESSES
TBL01	W	8,192	4	11	15
TBL02	W	8,192	0	11	11
***** END OF STROBE			NUMBER	1	*****
***** START OF STROBE			NUMBER	2	*****
TIME OF STROBE	2017/10/31 11:35:32		CALL COUNT	26	
TSR USED	45,056		TSR USED HWM	45,056	
LENGTH OF OPEN TABLES	16,384		LENGTH OF TABLES HWM	16,384	
NUMBER OF OPEN TABLES	2		NUMBER OF TABLES HWM	2	
***** O P E N			T A B L E S	*****	
TABLE NAME	ACCESS	LENGTH	READS	UPDATES	TOTAL ACCESSES
TBL01	W	8,192	4	11	15
TBL02	W	8,192	0	11	11
***** END OF STROBE			NUMBER	2	*****
***** START OF STROBE			NUMBER	3	*****
TIME OF STROBE	2017/10/31 11:40:32		CALL COUNT	30	
TSR USED	45,056		TSR USED HWM	45,056	
LENGTH OF OPEN TABLES	16,384		LENGTH OF TABLES HWM	16,384	
NUMBER OF OPEN TABLES	2		NUMBER OF TABLES HWM	2	
***** O P E N			T A B L E S	*****	
TABLE NAME	ACCESS	LENGTH	READS	UPDATES	TOTAL ACCESSES
TBL01	W	8,192	4	11	15
TBL02	W	8,192	4	11	15
***** END OF STROBE			NUMBER	3	*****
***** END OF STROBE			REPORT	*****	

Figure 3-4: Sample strobe report: VTS-TSR (STROBEMETHOD=3, STROBE=5)

Section	Field name	Contains
Header information	tableSPACE Region name	LOCAL_DS if strobe is for a local TSR. VTS-TSR name if the strobe is for a VTS-TSR
	TSR Region start	The date and time that the TSR Region started
	TSR size	The size in bytes of the TSR
	Strobe interval	The strobe interval for the region

Section	Field name	Contains
Start of strobe	Strobe number	The strobe number for which the following information is being written
	Time of strobe	The date and time at which the statistics were captured for this strobe
	Call count	The total number of tableBASE calls that have been made at the end of this strobe interval. Only calls that reference the TSR are counted (calls such as BN, LS, ML, CS, and LL are not counted). The call count can accommodate up to 13 digits (maximum value of 9,999,999,999,999).
	TSR used	The amount of TSR that is in use at the end of the strobe interval
	TSR used HWM	The high water mark is the maximum amount of tableSPACE used up to now
	Length of open tables	The total size of all open tables in the TSR at the end of the strobe interval. The length field can accommodate up to 11 digits (maximum value of 99,999,999,999).
	Length of tables HWM	The highwater mark is the maximum size of all open tables in the TSR at any time since the region has been active. The length field can accommodate up to 11 digits (maximum value of 99,999,999,999).
	Number of open tables	The total number of all open tables in the TSR at the end of the strobe interval (note that this value may not match the number of tables listed, due to the use of sub-tasking in performance data collection)
	Number of tables HWM	The highwater mark is the maximum number of all open tables in the TSR at any time since the region has been active
Open tables	Table name	The name of the table occupying that position in the TSR.
	Access	R for read or W for write

Section	Field name	Contains
	Length	The number of bytes in the table. The length field can accommodate up to 11 digits (maximum value of 99,999,999,999).
	Reads	The number of reads in this TSR. The number of reads can be up to 13 digits (maximum value of 9999,999,999,999).
	Updates	The number of inserts, deletes, or replaces in this TSR. The number of updates can be up to 13 digits (maximum value of 9999,999,999,999).
	Total accesses	The sum of Reads and Updates. The total accesses can be up to 13 digits (maximum value of 9999,999,999,999).

Table 3-1: tableSPACE report field definitions

Note: The page number field at the top of each page of the strobe report can be up to 5 digits long (maximum value of 99999).

For further information on the use of the tableSPACE Report and the STROBE parameter, see the *tableBASE Programming Guide*.

Chapter

4

tableBASE User Exits

tableBASE software allows users to define their own exits. This functionality provides you with the capability to program an exit from tableBASE to call other routines.

User Exits

In previous versions of tableBASE, the exits provided for tableBASE users, along with those reserved for DKL, were referred to as tableBASE system exits. Starting with tableBASE Version 6, a clear distinction is made between user exits and system exits. User exits are those designed for tableBASE users, while system exits are reserved strictly for use by DKL. This chapter deals with building and implementing user exits.

Starting with Version 6, user exits have been significantly enhanced to allow for added flexibility and functionality. There are now six dynamically invoked user exits available in all command environments, as summarized in [Table 4-1](#) on page 15.

Previous Exit Point or Level	New Exit Point or Level	Description
TBEACH	XLBCEUSR	This is the BEFORE COMMAND exit and is executed before each tableBASE command.
New starting with Version 6	XLACEUSR	This is the AFTER COMMAND exit and is executed after each tableBASE command.
TBFIRST	XLARIUSR	This is the AFTER REGION INITIALIZATION exit is executed after region initialization.
TBLAST	XLBRTUSR	This is the BEFORE REGION TERMINATION exit is executed before region termination.

Previous Exit Point or Level	New Exit Point or Level	Description
New in Release 6.0.2	XLATIUSR	This is the AFTER THREAD INITIALIZATION exit is executed after thread initialization.
New in Release 6.0.2	XLBTTUSR	This is the BEFORE THREAD TERMINATION exit is executed before thread termination.

Table 4-1: Comparing Old and New User Exits

Note: The interface to the exits starting with tableBASE Version 6 is NOT compatible with that of prior versions.

Working with User Exits

This section describes the steps required to build and implement user exits.

Step 1—Programming User Exits

Write your exit modules. Once the modules are compiled/assembled, linked, and placed in the appropriate libraries, you are ready to identify them to tableBASE. This is done through macro statements. [Step 2—Defining Exits using DK1TX072](#) on page 17 describes how to identify your exits to tableBASE.

It is important to keep in mind that exit routines that are run often (for example, before and after every command) may affect performance. Moreover, as exit routines may be invoked many times and may be called upon for tasks that a high-level language may not be able to perform efficiently, users may want to consider writing user exits in IBM Assembler.

Programmers should carefully read the sections describing parameter usage. For information on how parameter data is passed when an exit is invoked, see [Parameter Data Passed to Exits](#) on page 23.

R15 is reserved for future use and should be set to zero before returning from a user exit to tableBASE. Upon return from the exit, if R15 is not zero, the following warning message will be issued in the JES log:

```
DK100100W Unexpected non-zero return code from user exit
```

Note: User exit modules must be reentrant, RMODE(ANY), AMODE(31), and must NOT issue tableBASE commands.

Sample JCL for assembling and link editing a user exit program can be found in your.prefix.CNTL(ALTUSREX).

Step 2—Defining Exits using DK1TX072

The exit definition module, DK1TX072, is used to enable each of the user exit points. Sample source is supplied in your.prefix.SRC. Sample JCL to assemble and link edit the module is supplied in your.prefix.CNTL(ALTXX072). The DK1TX072 load module provided by DKL has all exits disabled.

The definition of the user exit points in DK1TX072 is accomplished by using the DK1EXDFN macro. The sample module contains seven iterations of the DK1EXDFN macro, one for the header and one for each of the six user exits. Only 1 header definition is required. The macro iterations must be issued in a specific order, so that they result in a table containing a single header followed by the actual exit definitions.

[Figure 4-3](#) on page 32 shows the sample DK1TX072 source module provided. For details on the use of the parameters, see [DK1EXDFN Macro Parameters](#) on page 17.

To enable your user exit points, the appropriate DK1EXDFN macro must be modified to indicate the user exits you will be using, the name of the user program associated with each exit and the amount of memory allocated for each exit.

Step 3—JCL Requirements for Exit Processing

Modify the STEPLIB of your tableBASE job or region to indicate the location of the DK1TX072 module and the user exit modules identified in DK1TX072.

Modify the SYSLIB, SYSIN, and SYSLMOD statements in ALTUSREX and ALTXX072.

Step 4—TBOPT Parameter Requirement

In order to activate user exits, the tableBASE parameter, USEREXITS, in the TBOPT data set must be set to Y. This will cause the DK1TX072 module to be loaded during initialization of the tableBASE region. For more information on tableBASE parameters, please see the *tableBASE Installation Guide*.

DK1EXDFN Macro Parameters

The parameters of the DK1EXDFN macros in the DK1TX072 module are used to modify the definitions for the user exits.

TYPE parameter:

The TYPE parameter in the DK1EXDFN macro determines whether a header (INITIAL) or an exit (ENTRY) definition will be produced.

- DK1EXDFN TYPE=INITIAL, ... defines a header
- DK1EXDFN TYPE=ENTRY, ... defines an exit

Defining a Header (TYPE=INITIAL)

Example:

```
DK1EXDFN TYPE=INITIAL, GBLWORK=nnn, SHRDWORK=nnn,      X
EXITFLD= (REG_START_TIME,                                X
REG_USER_ID,                                              X
REG_VTS_FIRST,                                           X
REG_VTS_LAST,                                             X
REG_VTS_PREFIX,                                          X
THRD_DD_NAME,                                             X
THRD_LIB_LIST,                                           X
THRD_STATUS_SW,                                          X
THRD_VTS_NAME,                                           X
TSR_CUR_#_OPEN,                                          X
TSR_CUR_SIZE,                                           X
TSR_HWM_#_OPEN,                                          X
TSR_HWM_SIZE,                                           X
TSR_START_TIME,                                          X
TSR_SYSTEM_SIZE,                                         X
TSR_TOTAL_CALLS)                                         X
```

GBLWORK=nnn

This parameter was added with Release 6.1.0 and replaces the SWRKAREA parameter from previous releases of Version 6. The SWRKAREA parameter will still be supported for backward compatibility.

This parameter defines the size of a global non-thread-related work area. The size of the global work area is user defined and can be set to zero, if not used. This area is shared by all of the exits for all of the tasks in the entire address space, that is, all exits for all threads. This means that all user exits will receive the address of the same area in storage.

This parameter is only valid when TYPE=INITIAL.

When an exit is invoked, the address of the global work area will be passed as Parameter 2. For more information, see [Parameter 2: Global Work Area](#) on page 29.

SHRDWORK=nnn

This parameter was added with Release 6.1.0 and replaces the WRKAREA parameter for the TYPE=INITIAL macro from previous releases of Version 6. The WRKAREA parameter will still be supported for backward compatibility.

This parameter defines the size of a work area that will be passed only to the following user exits:

- BEFORE COMMAND and AFTER COMMAND exits (XLBCEUSR, XLACEUSR)
- AFTER THREAD INITIALIZATION and BEFORE THREAD TERMINATION exits (XLATIUSR, XLBTTUSR).

The size of the shared work area is user defined and can be set to zero, if not used. This area will be the same for all exits related to a single task, that is, all exits of the same thread. It can therefore be used for communication between the exits.

When an exit is invoked, the address of the shared, thread-related work area will be passed as Parameter 3. For more information, see [Parameter 3: Shared Thread Related Work Area](#) on page 29.

EXITFLD

This parameter is new with Release 6.1.0 and provides user exits with access to new fields. When an exit is invoked, the address for accessing these fields is passed in Parameter 1. For more information, see [Parameter 1: Communication Area](#) on page 23.

The field names within parentheses are keyword parameters so you only need to specify the field names for fields which are to be accessed by your user exits. The fields with prefix 'TSR' are similar to the fields returned in the LIST-BLOCK parameter for the LT command (see *tableBASE Programming Guide* for more details).

The fields for this parameter are described below.

REG_START_TIME

This is the 16 byte date and time that tableBASE was initialized in the region. The format for this field is *yyyymmddhhmmssst* where *yyyymmdd* is the gregorian date and *hhmmssst* is the time (*th* represents tenths and hundredths of a second).

This field can be accessed by all user exits.

REG_USER_ID

This is the 8 byte userid assigned to the job. This field can be accessed by all user exits.

REG_VTS_FIRST

This is the 8 byte VTSFIRST parameter from the TBOPT setting for the job. This field can be accessed by all user exits.

REG_VTS_LAST

This is the 8 byte VTSLAST parameter from the TBOPT setting for the job. This field can be accessed by all user exits.

REG_VTS_PREFIX

This is the 4 byte VTSPREFIX parameter from the TBOPT setting for the job. This field can be accessed by all user exits.

THRD_DD_NAME

This is the 8 byte DDNAME of the last tableBASE library used to access a table on a library.

This field can only be accessed by the BEFORE COMMAND, AFTER COMMAND, AFTER THREAD INITIALIZATION and BEFORE THREAD TERMINATION user exits (XLBCEUSR, XLACEUSR, XLATIUSR and XLBTTUSR).

THRD_LIB_LIST

This is the 80 byte library list that is currently in use by tableBASE.

This field can only be accessed by the BEFORE COMMAND, AFTER COMMAND, AFTER THREAD INITIALIZATION and BEFORE THREAD TERMINATION user exits (XLBCEUSR, XLACEUSR, XLATIUSR and XLBTTUSR).

THRD_STATUS_SW

This is the 8 bytes of status switches that is currently in use by tableBASE.

This field can only be accessed by the BEFORE COMMAND, AFTER COMMAND, AFTER THREAD INITIALIZATION and BEFORE THREAD TERMINATION user exits (XLBCEUSR, XLACEUSR, XLATIUSR and XLBTTUSR).

THRD_VTS_NAME

This is an 8 byte field that returns the name of the TSR currently being used by this thread. If a local TSR is being used, the field contains the literal "LOCAL_DS". If a VTS-TSR is being used, it contains the name of the VTS-TSR.

This field can only be accessed by the BEFORE COMMAND, AFTER COMMAND, AFTER THREAD INITIALIZATION and BEFORE THREAD TERMINATION user exits (XLBCEUSR, XLACEUSR, XLATIUSR and XLBTTUSR).

TSR_CUR_#_OPEN

This is a 4 binary byte field that returns the number of tables that are currently open in the TSR.

This field can only be accessed by the BEFORE COMMAND, AFTER COMMAND, AFTER THREAD INITIALIZATION and BEFORE THREAD TERMINATION user exits (XLBCEUSR, XLACEUSR, XLATIUSR and XLBTTUSR).

TSR_CUR_SIZE

This is a 4 binary byte field that returns the amount of TSR space, in bytes, that is currently in use.

This field can only be accessed by the BEFORE COMMAND, AFTER COMMAND, AFTER THREAD INITIALIZATION and BEFORE THREAD TERMINATION user exits (XLBCEUSR, XLACEUSR, XLATIUSR and XLBTTUSR).

TSR_HWM_#_OPEN

This is a 4 binary byte field that returns the maximum number of tables that have ever been opened in the TSR.

This field can only be accessed by the BEFORE COMMAND, AFTER COMMAND, AFTER THREAD INITIALIZATION and BEFORE THREAD TERMINATION user exits (XLBCEUSR, XLACEUSR, XLATIUSR and XLBTTUSR).

TSR_HWM_SIZE

This is a 4 binary byte field that returns the maximum amount of TSR space, in bytes, that has ever been used.

This field can only be accessed by the BEFORE COMMAND, AFTER COMMAND, AFTER THREAD INITIALIZATION and BEFORE THREAD TERMINATION user exits (XLBCEUSR, XLACEUSR, XLATIUSR and XLBTTUSR).

TSR_START_TIME

This is a 16 byte field containing the date and time when the current TSR was started. The format for this field is *yyyymmddhhmmssst* where *yyyymmdd* is the gregorian date and *hhmmssst* is the time (*th* represents tenths and hundredths of a second).

This field can only be accessed by the BEFORE COMMAND, AFTER COMMAND, AFTER THREAD INITIALIZATION and BEFORE THREAD TERMINATION user exits (XLBCEUSR, XLACEUSR, XLATIUSR and XLBTTUSR).

TSR_SYSTEM_SIZE

This is a 4 byte binary field that returns the size of the TSR, in bytes.

This field can only be accessed by the BEFORE COMMAND, AFTER COMMAND, AFTER THREAD INITIALIZATION and BEFORE THREAD TERMINATION user exits (XLBCEUSR, XLACEUSR, XLATIUSR and XLBTTUSR).

TSR_TOTAL_CALLS

This is an 8 byte binary field that returns the total number of calls made on tables in the TSR.

This field can only be accessed by the BEFORE COMMAND, AFTER COMMAND, AFTER THREAD INITIALIZATION and BEFORE THREAD TERMINATION user exits (XLBCEUSR, XLACEUSR, XLATIUSR and XLBTTUSR).

Defining the Entries (TYPE=ENTRY)

Example:

```
DK1EXDFN
```

```
TYPE=ENTRY, NAME=cccccccc, LEVEL=xxxxxxxx, PRVTHWORK=nnn, DISABLED=xxx
```

Note: TYPE=ENTRY is the default, and may be omitted.

NAME=cccccccc

This is the name of the module that is invoked for the specific exit indicated by the LEVEL parameter, explained below. This module must be on a library pointed to by the STEPLIB DD statement/concatenation.

LEVEL=xxxxxxxx

This is the exit point at which the named module will be invoked. [Table 4-2](#) on page 22 lists the six exit points and their descriptions.

Exit Point or Level	Description
XLBCEUSR	BEFORE COMMAND exit - invoked before each tableBASE command.
XLACEUSR	AFTER COMMAND exit - invoked after each tableBASE command.
XLARIUSR	AFTER REGION INITIALIZATION exit - invoked after tableBASE region initialization.
XLBRTUSR	BEFORE REGION TERMINATION exit - invoked before tableBASE region termination.
XLATIUSR	AFTER THREAD INITIALIZATION exit - invoked after thread initialization.
XLBTTUSR	BEFORE THREAD TERMINATION exit - invoked before thread termination.

Table 4-2: Exit Points**PRVWORK=nnn**

This parameter was added with Release 6.1.0 and replaces the WRKAREA parameter for the TYPE=ENTRY macro from previous releases of Version 6. The WRKAREA parameter will still be supported for backward compatibility.

This is the size of a PRIVATE area of storage that is to be passed to the exit, that is, it will not be shared, and will be discarded at the completion of the exit. This area can be used as working storage for an exit to avoid having to allocate memory in the exit. The size of the private work area is user defined and can be set to zero, if not used.

The DK1EXDFN macro forces any defined PRVWORK area to be a multiple of eight. If the PRVWORK value is defined to be any other number, it will be rounded up to the next multiple of 8, and a warning message will be generated.

When an exit is invoked, the address of the private, thread-related work area will be passed as Parameter 4. For more information, see [Parameter 4: Private Thread Related Work Area](#) on page 30.

DISABLED=YES | NO

This parameter determines if this exit is to be in use while tableBASE is processing, where

- YES—the exit is disabled, and will not be used
- NO—enable the exit (default)

Note: All of the exit definitions are put into effect when tableBASE is initialized, and will remain in effect until tableBASE is re initialized with new definitions.

Parameter Data Passed to Exits

[Figure 4-1](#) on page 23 shows how parameter data is defined and passed to each of the exits. Note that some parameters are exit specific.

For all exits:

- GPR1 will point to a parameter list.
- The parameter list will contain two entries for the AFTER REGION INITIALIZATION and BEFORE REGION TERMINATION exits (XLARIUSR, XLBRTUSR). There will be four entries for the BEFORE COMMAND, AFTER COMMAND, AFTER THREAD INITIALIZATION and BEFORE THREAD TERMINATION exits (XLBCEUSR, XLACEUSR, XLATIUSR and XLBTTUSR).

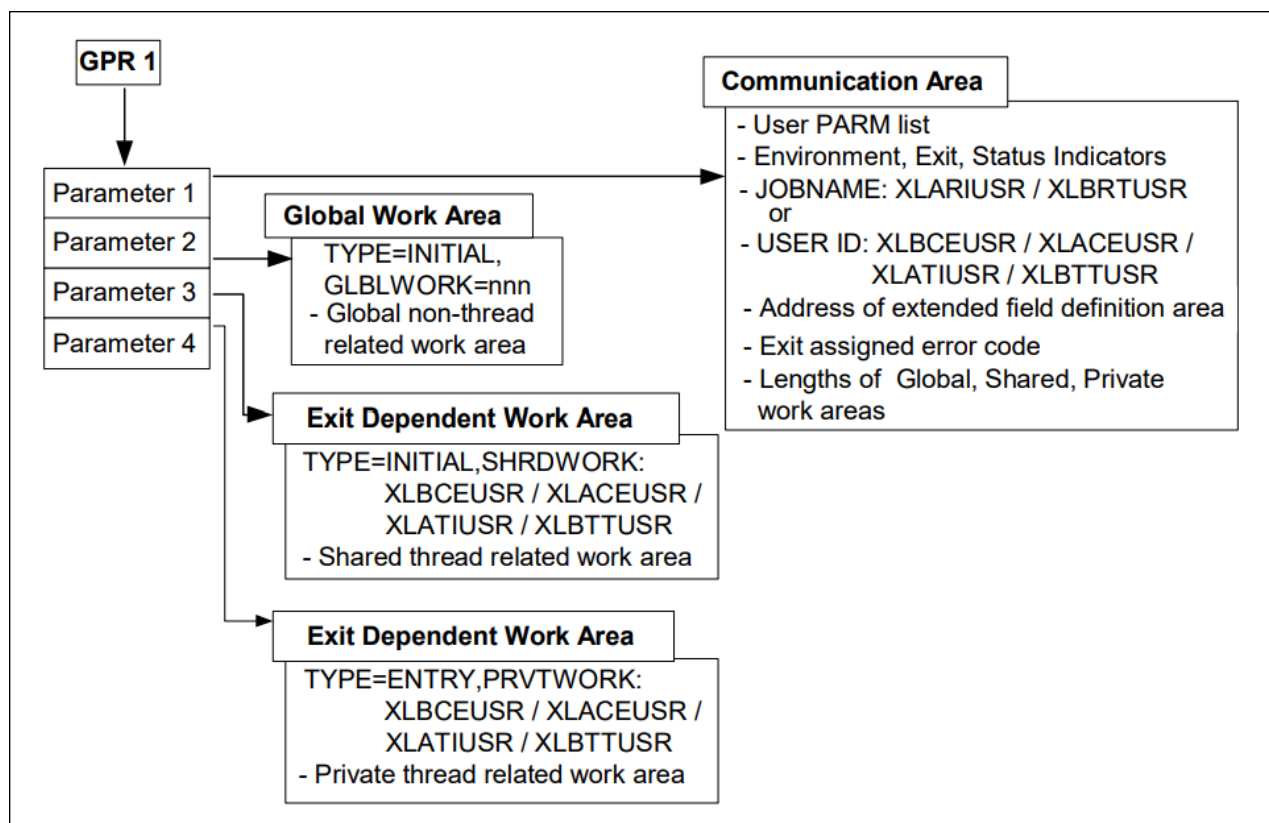


Figure 4-1: User exits overview

Parameter 1: Communication Area

Parameter 1 is used for all exits.

This is the address of a 44 byte Communication Area, containing information describing selected system conditions either at the time the exit is invoked or after exit processing.

The following is a breakdown of the fields in the Communication Area.

Bytes 0–3: Address of User Parameter List

This field is only for the BEFORE COMMAND and AFTER COMMAND exits (XLBCEUSR, XLACEUSR).

It is the address of the user parameter list as passed to tableBASE by the application program. The user parameter list includes TBPARM (if supplied), TB-COMMAND-AREA, and any other tableBASE parameters associated with the current command. Usage of the DKL supplied DSECTs for the various parameter areas is recommended.

Byte 4: Environment Indicator

The single byte area contains a code indicating the environment in which tableBASE is running, as listed in [Table 4-3](#) on page 24.

Environment Indicator	Environment Description
B	Batch
C	CICS
I	IMS Transaction Manager
D	DB2 managed stored procedure address space (SPAS)
W	WLM (WorkLoad manager) managed stored procedure address space (SPAS)

Table 4-3: Environment Indicator

Byte 5: Exit Indicator

This single byte area contains a code that indicates which of the six exits is in effect, as listed in [Table 4-4](#) on page 24.

Value	Exit Level	Exit Description
X'00'	XLBCEUSR	BEFORE COMMAND exit.
X'01'	XLACEUSR	AFTER COMMAND exit.

Value	Exit Level	Exit Description
X'02'	XLARIUSR	AFTER REGION INITIALIZATION exit.
X'03'	XLBRTUSR	BEFORE REGION TERMINATION exit.
X'04'	XLATIUSR	AFTER THREAD INITIALIZATION exit.
X'05'	XLBTTUSR	BEFORE THREAD TERMINATION exit.

Table 4-4: Exit Indicator

Byte 6: Status Indicator

This byte is used to indicate several types of abends that may have occurred in tableBASE. It could also indicate whether the user has selected to issue a user-initiated abend or bypass processing for a tableBASE command. The bit settings and descriptions are listed in [Table 4-5](#) on page 25.

Value	Description
X'80'	This bit indicates that there has been an abend in the main task in the region or the particular thread that is executing. It is set on by the exit manager at the time of calling the BEFORE THREAD TERMINATION and BEFORE REGION TERMINATION exits (XLBTTUSR, XLBRTUSR). Note: This bit does not appear to be set on for all types of abends.
X'40'	This bit indicates that there has been an abend in the mother task. It is set on by the exit manager at the time of calling the BEFORE COMMAND, AFTER COMMAND, BEFORE THREAD TERMINATION and BEFORE REGION TERMINATION exits (XLBCEUSR, XLACEUSR, XLBTTUSR, XLBRTUSR). Note: This bit does not appear to be set in batch even if Multitasking=Y.
X'20'	This bit indicates that there has been an internal tableBASE abend. This bit is set on by the exit manager at the time of calling the BEFORE THREAD TERMINATION and BEFORE REGION TERMINATION exits (XLBRTUSR, XLBTTUSR) if the main task in the region or the particular thread that is executing abends. Note: This bit does not appear to be set on for all types of abends.

Value	Description
X'04'	If this bit is set on by the BEFORE COMMAND or AFTER COMMAND exits (XLBCEUSR, XLACEUSR), the exit manager will issue an abend after return from the exit, unless the current task is the main task running in a multitasking environment, i.e. if bit X'40' of the status indicator is on and MULTITASKING=Y is set, the abend will not be issued. When the abend is issued, the following message will appear in the JES log: <pre>DK100400E: tableBASE User Error Abend nnnn, Command = xx,yyyyyyyy</pre> where <i>nnnn</i> is the user error code in bytes 20-23 of the communication area, <i>xx</i> is the tableBASE command and <i>yyyyyyyy</i> is the table name.
X'02'	If this bit is set on by the BEFORE COMMAND exit (XLBCEUSR) program, execution of the current tableBASE command will be bypassed upon return from the exit. If an AFTER COMMAND exit (XLACEUSR) exists, it will be executed. This bit must be reset to prevent subsequent commands from being bypassed.

Table 4-5: Status Indicator

Byte 7: Reserved

Bytes 8-15: JOBNAME / USERID

The content of the 8-byte JOBNAME/USERID parameter varies depending on the exit:

- for the AFTER REGION INITIALIZATION (XLARIUSR) and BEFORE REGION TERMINATION (XLBRTUSR) exits, it contains the JOBNAME
- for the BEFORE COMMAND, AFTER COMMAND, AFTER THREAD INITIALIZATION and BEFORE THREAD TERMINATION exits (XLBCEUSR, XLACEUSR, XLATIUSR, XLBTTUSR), it contains the USERID.

Bytes 16-19: Address of Extended Field Definition Area

This is new with Release 6.1.0 and contains the address of the extended field definition area allowing access to the fields specified by the EXITFLD parameter in the DK1EXDFN TYPE=INITIAL macro. For more information on the extended field definition area, see [Extended Field Definition Area Layout](#) on page 27.

Bytes 20-23: User Exit Assigned Error Code

This field is new with Release 6.1.0. It contains the user assigned error code for abends forced by setting the Status Indicator (byte 6 of the Communication Area) to X'04' by the BEFORE COMMAND or AFTER COMMAND user exits.

Acceptable values for this field are from 102 to 199. The value set in this field remains until it is reset by another user exit. If the value in this field is zero or outside the range of acceptable values, the default user abend error code of 101 will be used.

This error code is displayed in the DK100400E error message described in [Table 4-5](#) on page 25.

Bytes 24-27: Length of Global Work Area

This is the length of the Global Work Area which was defined in the GLBLWORK parameter of the DK1EXDFN TYPE=INITIAL macro. This field is new with Release 6.1.0.

Bytes 28-31: Length of Shared Work Area

This is the length of the Shared Work Area which was defined in the SHRDWORK parameter of the DK1EXDFN TYPE=INITIAL macro. This field is new with Release 6.1.0.

Bytes 32-35: Length of Private Work Area

This is the length of the Private Work Area which was defined in the PRVWORK parameter of the DK1EXDFN TYPE=ENTRY macro. This field is new with Release 6.1.0.

Bytes 36-43: Reserved

Extended Field Definition Area Layout

This data area is addressed by bytes 16-19 of the Communication Area and its fields are defined in the EXITFLD parameter of the DK1EXDFN TYPE=INITIAL macro. For each field that is specified in the EXITFLD parameter, there will be an associated Extended Field Definition

Area and a corresponding Field Value. [Figure 4-2](#) on page 28 below shows the layout of the Extended Field Definition Area.

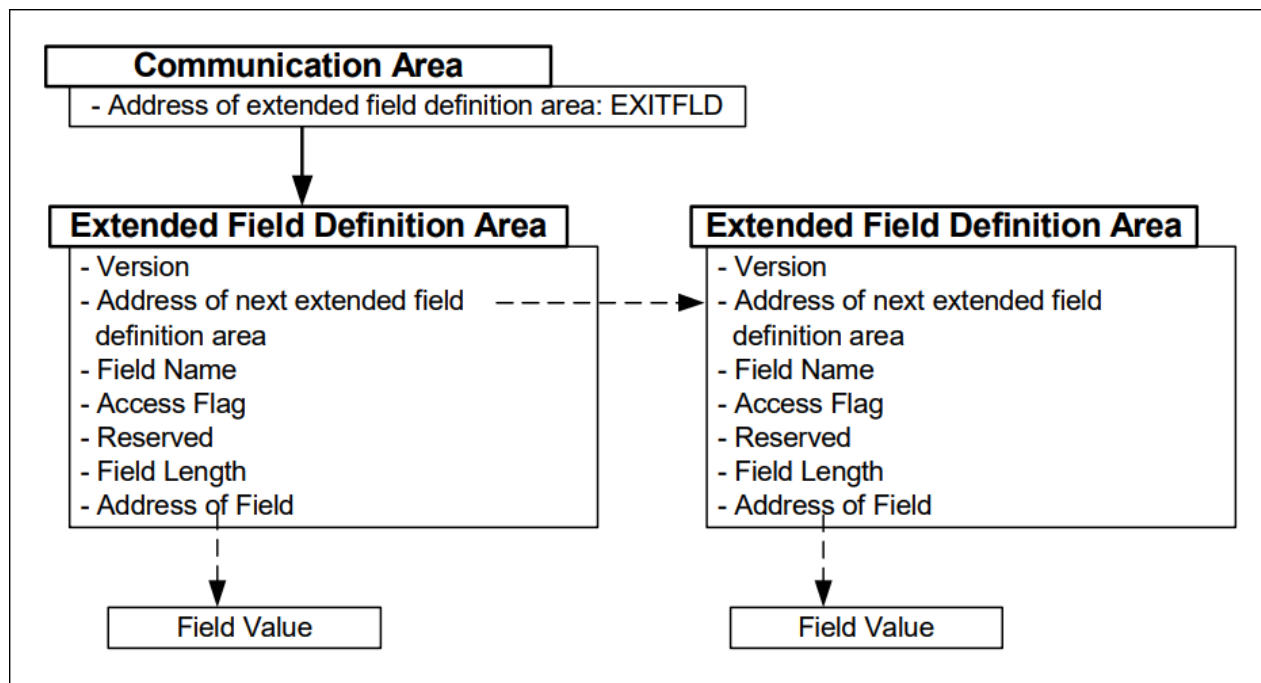


Figure 4-2: Extended Field Definition Area Layout

Bytes 0-3: Version

This field contains the version number for the interface to the extended field definition area. The format is `xvvv` where `vvv` is the version number. For this first release, it is set to 'x603'. This field allows the user exits to verify that the format of the extended field definition area is at the correct level for the exit that is using it.

Bytes 4-7: Address of next extended field definition area

This is the address of the next extended field definition area, if there is one. It is zero if there are no more extended field definition areas.

Bytes 8-23: Field Name

This is the field name for the field that is being passed back in the extended field definition area. The value of the field name is the same as that specified in the EXITFLD parameter of the DK1EXDFN macro.

Byte 24: Access Flag

This byte indicates the type of access that is allowed on the field returned. The descriptions are listed in [Table 4-6](#) on page 29.

Access Flag	Description
R	This field is available for read-only access. This setting should be on for all fields that are accessible by the exit.
N	The field name that was specified is not available for any exit.
U	The field name that was specified is not available for this type of exit.

Table 4-6: Access Flag**Byte 25: Reserved****Bytes 26-27: Field Length**

This is a halfword binary field containing the length of the field being passed.

Bytes 27-30: Address of Field

This is the address of the actual memory area where the value of the field can be found.

Field Value

This is the actual data for the field that was requested in the EXITFLD parameter of the DK1EXDFN TYPE=INITIAL macro. It is the area pointed to by bytes 27-30 (Address of Field) of the Extended Field Definition Area. The maximum length of this area is 80 bytes.

Parameter 2: Global Work Area

Parameter 2 is used for all exits

This is the address of the global data area that is given to all exits in the address space. It is not thread-related. The area length is defined in the GLBLWORK parameter of the DK1EXDFN TYPE= INITIAL macro.

Parameter 3: Shared Thread Related Work Area

For the BEFORE COMMAND, AFTER COMMAND, AFTER THREAD INITIALIZATION and BEFORE THREAD TERMINATION exits (XLBCEUSR, XLACEUSR, XLATIUSR, XLBTTUSR), this is the address of the data area shared by each of these exits when they are part of the same task (thread). This work area may be used for passing information between these exits. It is defined in the SHRDWORK parameter of the DK1EXDFN TYPE=INITIAL macro.

AFTER REGION INITIALIZATION and BEFORE REGION TERMINATION exits (XLARIUSR, XLBRTUSR) do not use parameter 3.

Parameter 4: Private Thread Related Work Area

For the BEFORE COMMAND, AFTER COMMAND, AFTER THREAD INITIALIZATION and BEFORE THREAD TERMINATION exits (XLBCEUSR, XLACEUSR, XLATIUSR, XLBTTUSR), this is the address of a private data area given to each of these exits on a per thread basis. This area is not shared with any other exit or any other thread. It is defined in the PRVWORK parameter of the DK1EXDFN TYPE=ENTRY macro.

AFTER REGION INITIALIZATION and BEFORE REGION TERMINATION exits (XLARIUSR, XLBRTUSR) do not use parameter 3.

Exit Related Messages

[Table 4-7](#) on page 30 below lists exit related messages and their meaning.

Msg Code	Text	Meaning / Instructions
DKI00260I	DK1TX066 Exit Manager Initializing	Information only. The exit manager was initialized during tableBASE initialization.
DK100265W	Exit name <i>exit-field-name</i> not found in current release	The field name supplied in the EXITFLD parameter of the DK1EXDFN TYPE=INITIAL macro is invalid.
DK100400E	tableBASE User Error Abend <i>nnnn</i> , Command = <i>xx,yyyyyyyy</i>	Abend initiated by either BEFORE COMMAND or AFTER COMMAND (XLBCEUSR, XLACEUSR) user exit. <i>nnnn</i> is the user error in bytes 20-23 of the communication area, <i>xx</i> is the command and <i>yyyyyyyy</i> is the table name at the time ofabend.

Table 4-7: Exit Related Messages

Sample User Exits

For the exit definition module (DK1TX072) and each user exit point, DKL supplies sample source code in your.prefix.SRC. Sample JCL to assemble and link edit DK1TX072 is in your.prefix.CNTL(ALTX072). Sample DSECTs for the Communication Area and the Extended Field Definition Area are in your.prefix.SRC(DK1XU1).

The DK1TX072 load module provided by DKL has all exits disabled. Sample source code for DK1TX072 is shown in [Figure 4-3](#) on page 32.

```

        TITLE '***      DK1TX072 V603 USER EXITS TABLE DEFINITION'
DK1TX072 AMODE ANY
DK1TX072 RMODE ANY
DK1TX072 RSECT
*       TABLEBASE STANDARD PREFIX
        DC      CL8'DK1TX072'          MODULE NAME
        DC      CL5'V603'              VERSION 6.0
        DC      CL9'&SYSDATE'          ASSEMBLY DATE
        DC      CL9'&SYSTIME',0H'0'    ASSEMBLY TIME
        ENTRY DK1EXUSR
        PRINT GEN,ON,DATA
        DC      AL4(DK1EXUSR-DK1TX072)      OFFSET FOR CICS
        DC      CL12'USER TABLE  '

*       USER EXIT TABLE DEFINITION:  PLEASE DO NOT CHANGE THE ORDER
*       HEADER DEFINITION:
DK1EXDFN TYPE=INITIAL, GBLWORK=016, SHRDWORK=016,                X
        EXITFLD=(REG_START_TIME,                                X
        REG_USER_ID,                                           X
        REG_VTS_FIRST,                                         X
        REG_VTS_LAST,                                          X
        REG_VTS_PREFIX,                                        X
        THRD_DD_NAME,                                          X
        THRD_LIB_LIST,                                         X
        THRD_STATUS_SW,                                        X
        THRD_VTS_NAME,                                         X
        TSR_CUR_#_OPEN,                                        X
        TSR_CUR_SIZE,                                          X
        TSR_HWM_#_OPEN,                                        X
        TSR_HWM_SIZE,                                          X
        TSR_START_TIME,                                        X
        TSR_SYSTEM_SIZE,                                       X
        TSR_TOTAL_CALLS)
*       USER EXIT DEFINITIONS:
*       0 - BEFORE EACH COMMAND EXECUTION
        DK1EXDFN TYPE=ENTRY, NAME=PT3069, LEVEL=XLBCEUSR,        X
        PRVTWORK=16, DISABLED=NO
*       WRKAREA=16, DISABLED=NO
*       1 - AFTER EACH COMMAND EXECUTION
        DK1EXDFN TYPE=ENTRY, NAME=PT3069, LEVEL=XLACEUSR,        X
        PRVTWORK=16, DISABLED=NO
*       WRKAREA=16, DISABLED=NO
*       2 - AFTER REGION INITIALIZATION
        DK1EXDFN TYPE=ENTRY, NAME=PT3069, LEVEL=XLARIUSR,        X
        DISABLED=NO
*       3 - BEFORE REGION TERMINATION
        DK1EXDFN TYPE=ENTRY, NAME=PT3069, LEVEL=XLBR TUSR,       X
        DISABLED=NO
*       XLATIUSR NOT SUPPORTED FOR HIGH LEVEL LANGUAGES (PT3068) 2010-08-21
*       4 - AFTER THREAD INITIALIZATION
        DK1EXDFN TYPE=ENTRY, NAME=PT3069, LEVEL=XLATIUSR,        X
        PRVTWORK=16, DISABLED=YES
*       WRKAREA=16, DISABLED=YES
*       5 - BEFORE THREAD TERMINATION
        DK1EXDFN TYPE=ENTRY, NAME=PT3069, LEVEL=XLBT TUSR,       X
        PRVTWORK=16, DISABLED=NO
*       WRKAREA=16, DISABLED=NO
        END

```

Figure 4-3: Sample implementation of DX1EXDFN in DK1TX072

Chapter

5

Program Call Server

tableBASE uses Program Call (PC) functionality, provided by IBM, to allow tableBASE program calls (PCs) to be available for tableBASE regions. This functionality is provided by the tableBASE PC Server.

The PC functionality supported by the PC Server provides the functions needed for tableBASE multi-tasking, for CICS thread-safe support, for accessing a VTS-TSR, and for compatibility with DB2 stored procedures, as well as other special functions that are required by tableBASE.

By using program calls, tableBASE can support those tasks that require special authorization. The PC Server also provides authorized address spaces for each LPAR in which tableBASE can run the following special functions:

- End-of-task exit clean-up for each region
- Enqueues, dequeues and cross-memory posts entailed in accessing tableBASE libraries in conjunction with a VTS-TSR
- Monitor the connections and disconnections of each VTS-TSR
- Enqueues and dequeues for accessing tableBASE in a CICS thread-safe application. For more details on PC Server implementation in a CICS environment, see [PC Server implementation in a CICS environment](#) on page 243.

Note: The PC Server is needed, must be installed, and must be running at all times before tableBASE initializes in a region.

Initialization of the Program Call Server

The tableBASE PC Server is required for each LPAR (MVS image) that is running tableBASE. Each PC Server must be initialized before the first use of tableBASE on the LPAR.

Initialization of the tableBASE PC Server is achieved by either submitting a batch job or a started task. It is recommended that it be set up as a started task and initiated at system initialization. Sample JCL is provided in the installation library, **your.prefix.SDK1CNTL**. For more information, see the *tableBASE Installation Guide*. Your systems programmer can assist you with the initialization of the Server.

The modules for the PC Server can only reside in an LPA, STEPLIB or SYSLIB; they cannot be loaded from LINKLIB.

Server Messages

Messages from the tableBASE PC Server will indicate successful initialization. If the PC Server is not available, messages will be returned indicating this condition, with explanations where applicable.

For message details, see [tableBASE messages](#) on page 161.

Licensing Messages

Licensing messages for some optional tableBASE products will also appear in the JESMSG LG of the PC Server.

The following are examples of the licensing messages which will appear for VTS Gate, if you are licensed for the product:

```
DK100833I VTS Gate licensed to <customer name>
DK100828W VTS Gate license will expire <yyyy-mm-dd>
DK100829W VTS Gate license expires today <yyyy-mm-dd>
DK100826W VTS Gate license expired <yyyy-mm-dd>; in grace
period
DK100827W VTS Gate license expired <yyyy-mm-dd>
```

For a detailed explanation of each message, please refer to [tableBASE messages](#) on page 161.

Re-initialization of the Program Call Server

If the Server is brought down for any reason, all regions that have already completed tableBASE initialization will continue to operate using the Program Calls that were established when these jobs initialized. Any regions which attempt to initialize while the Server is down will fail. If the Server is brought back up at a different modification level, all new regions will initialize at the modification level of the new Server. All regions that were running before the previous Server came down and after the current Server was brought up will continue to run at the level of the previous Server.

- Note: 1.** For CICS, the TBST TERM transaction may be used, followed by the TBST TBINIT transaction, to re-attempt initialization. However, do not attempt to do this without first calling DataKinetics Technical Support.
- 2.** See [PC Server implementation in a CICS environment](#) on page 243 for PC Server implementation in a CICS environment.

Restarting the tableBASE Program Call Server

If the tableBASE PC Server was brought down and needs to be restarted at the same level, resubmit the batch job or started task which was used to initialize the PC Server.

If the tableBASE PC Server was brought down and needs to be restarted at a different level, modify the batch job or started task to point to the new load library and submit it.

Refer to the *tableBASE Installation Guide* for details on the job to use to start the tableBASE PC Server.

Shutting down the Program Call Server

In order to shutdown the tableBASE PC Server before system shutdown, all VTS regions that are running under the Server must first be shutdown. The MVS STOP (P) command can be used to shutdown the Server.

tableBASE PC Server shutdown

The tableBASE PC Server may be shutdown normally by issuing, from an MVS console, a stop command to the PC Server job or started task. The form of the command is:

P jobname

Chapter

6

DB2 Stored Procedures

DB2 Stored Procedures in a WLM-Managed DB2 SPAS (Stored Procedure Address Space) have been supported by tableBASE since Version 6. Each WLM-Managed DB2 SPAS environment is similar to using tableBASE in a multi-threaded batch environment.

The first Stored Procedure that runs in a DB2 SPAS causes tableBASE to initialize and a local TSR is created for that DB2 SPAS. Therefore, tables that are in a local TSR are not shared between DB2 SPAS. A VTS-TSR must be used for Stored Procedure usage.

Using DB2 stored procedures

All DB2 Stored Procedures must be linked AMODE(31), RMODE(ANY). With the use of DB2, strobe is forced off by the operating environment. As a result strobe is not supported for use with DB2 Stored Procedures, so it should be set to 0.

Note: Stored Procedures must be linked with the Version 6 DK1TCALL stub.

In JCL for WLM-Managed DB2 SPAS, include:

- your.prefix.LOAD in the STEPLIB (not needed if tableBASE load modules are available in LPA or system link-libraries)
- DD for TBOPT file (if desired),
- //TBDUMP DD SYSOUT=* statement to capture diagnostic information in case of an abend.

- Note: 1.** Stored Procedures must *not* create or access tables using a local TSR. If the first Stored Procedure that uses tableBASE in the WLM-Managed DB2 SPAS abends, the local TSR in that DB2 SPAS vanishes. This can have unpredictable effects on any other concurrently running Stored Procedures in the same DB2 SPAS that are also using tableBASE, if tables from the local TSR are being used. This also means that you must not use linked VTS tables in a Stored Procedure (these are tables with a VTS:xxxx entry in the ML-List or tables accessed using the VTSFIRST and VTSLAST TBOPT options). When accessing linked VTS tables, tableBASE creates an entry in the local TSR which re-directs accesses of the table to the VTS-TSR.
- 2.** Stored Procedures cannot access tableBASE libraries. The normal configuration for a Stored Procedure is to use a shared VTS-TSR that would be pre-loaded via a batch DK1TDRV job. The tableBASE run-time option ALLOW_TB_LIB_IO will allow you to set up access to a tableBASE library, but it is not recommended as this could cause tableBASE to abend if more than one Stored Procedure in a DB2 SPAS were to access a tableBASE library at the same time.

Chapter

7

IMS

This chapter provides information on how to tailor the IMS environment to accommodate tableBASE.

JCL

Add the following DD statements to the JCL of your BMPs or MPRs that will be using tableBASE:

1. To the STEPLIB concatenation, add

```
//      DD  DISP=SHR,DSN=your.prefix.SDK1LOAD
```

2. Add, for each tableBASE library to be referenced

```
//libname      DD  DISP=SHR,DSN=your.prefix.table_lib
```

Here, libname is the name that will be used in LIB-LIST of applications calling tableBASE to reference the library specified by the DSN value.

3. Add, if desired, a DD for the TBOPT file (a sequential file or PDS member, of format F or FB, record length 80).

```
//TBOPT      DD  DISP=SHR,DSN=your.prefix.tbopt_file
```

This file is read whenever tableBASE initializes. See [tableBASE Runtime Options](#) on page 135 for more information on TBOPT parameters.

4. If you use the Strobe feature, add a DD for the strobe report, for example,

```
//TBTSRPT    DD  SYSOUT=*
```

For IMS programs that are not BMPs or MPRs, see the *tableBASE Programming Guide* on how to set up your JCL.

Pre-loading tableBASE modules

DataKinetics recommends adding all of the module names to your MPR pre-load members. Not all of the modules included in [Table 7-1](#) on page 40 are needed in certain IMS MPR configurations. However, determining which modules are required depends on many factors, including whether your applications call tableBASE dynamically or statically and whether or not your application load modules have been re-linked with the tableBASE Version 6 stub. The larger

modules reside above the line, many others are aliases of DK1TCALL. It is simplest to load all of the names in your MPR pre-load.

Pre-Load Module	Alias of	Size (approx) Bytes	Above line
DK1TROT ^{1,2}		70000	Yes
DK1TNUCL ²		100000	Yes
DK1IBASE		2000	Yes
DK1TVROT		10	Yes
DK1TNTDP		16	Yes
DK1TNAME		1300	Yes
TBCALL ^{3,4}	DK1TCALL	6000	No
TBCALLI	DK1TCALL		No
TBROOTI	DK1TCALL		No
TBASEV	DK1TCALL		No
TBCALLV	DK1TCALL		No

Table 7-1: tableBASE modules

Application/transaction initialization

At the first call to tableBASE from an MPR application, tableBASE initializes itself in the region. This consists of reading any TBOPT file, processing the DK1IBASE module, obtaining storage space for control blocks, and acquiring a Data Space for the local TSR. This initialization occurs only once if no transactions abend. If a transaction abends, all resources (memory, TSR) acquired by tableBASE are freed; any tables in the TSR vanish without being stored. If the MPR control

¹ All modules listed are reentrant.

² If DK1TNUCL or DK1TROT is in LPA, DK1TVROT and/or DK1TNTDP should also be in the pre-load list.

³ Only one copy of aliased module pre-loaded.

⁴ Could be re-linked to be above the line if you are absolutely certain you have no application modules that run RMODE(24).

task re-attaches the application control task, tableBASE initialization will re-occur at the next call to tableBASE.

This effectively means the contents of the local TSR can be volatile since any updates made to local tables will be lost in the event any transaction abends.

tableBASE treats the entire sequence of transactions, from first call until normal shutdown of the MPR (or until a transaction abends) as a single 'thread' of execution. A tableBASE thread has some attributes that persist for the life of the thread, for example, its active ML list, its status switch settings. Those two statements imply that one transaction will inherit the attributes of any earlier transaction that may have set them differently from the default values. Thus writers of MPR transaction modules that call tableBASE must explicitly set a suitable ML list and status switch values to meet the application's needs. If a TBPARM parameter is used (that is, the application adheres to the tableBASE R5 API definition), then the TB-TURBO field should be set to low values (binary zeros) at the start of each transaction. It should not be changed after that. If a specific VTS-TSR is to be accessed by the application, its name should be put into the TB-SUBSYS field. If no VTS is required, set TB-SUBSYS to low values (binary zeros).

Essentially this states that each transaction module should start off by initializing as if it were its first invocation, no matter how many times it may be used during the MPR's lifetime.

Transaction abends

When a transaction abend occurs in an MPR, the current state of tableBASE is lost (TSR, tables in it, memory acquired for control information, including thread control information). If it is subsequently called by another transaction, it will re-initialize as if it were starting up for the first time.

VTS-TSR access

If a VTS region is stopped and restarted while an IMS region has access to the VTS-TSR, transactions attempting to access the VTS-TSR will receive error code 1072 until one of the following occurs:

- the transaction is stopped and restarted
- the transaction abends

Chapter

8

CICS

This chapter provides information on how to tailor the CICS environment to accommodate tableBASE.

JCL

Add the following DD statements to the JCL of your CICS regions that will be using tableBASE:

1. To the STEPLIB concatenation, add

```
//      DD    DISP=SHR,DSN=your.prefix.LOAD
```

2. Add, for each tableBASE library to be referenced

```
//libname      DD    DISP=SHR,DSN=your.prefix.table_lib
```

Here, libname is the name that will be used in LIB-LIST of applications calling tableBASE to reference the library specified by the DSNNAME value.

3. If you are using a sequential file or PDS member for TBOPT, instead of a VSAM file, add a DD for the TBOPT file.

```
//TBOPT      DD    DISP=SHR,DSN=your.prefix.tbopt_file
```

A TBOPT sequential file or PDS member must be defined with format F or FB with record length 80. This file is read whenever tableBASE initializes. See [tableBASE Runtime Options](#) on page 135 for more details on the TBOPT parameters.

Note: Starting with Version 6, the TBOPT file can either be a sequential file or a VSAM file. In previous releases, the TBOPT file always had to be a VSAM file. Omit this DD statement if you continue to use a VSAM TBOPT file.

Definitions

Definitions for CICS are covered in the *tableBASE Installation Guide*, including:

Journal

Definitions for journal files are covered in the *tableBASE Installation Guide*, in the "Create CICS definitions for tableBASE journal files" section of the "Installing the CICS Interface option" procedure.

Programs

Definitions for programs are covered in the *tableBASE Installation Guide*, in the "Create CICS definitions for tableBASE programs" section of the "Installing the CICS Interface option" procedure.

Transactions

Definitions for transactions are covered in the *tableBASE Installation Guide*, throughout the "Installing the CICS Interface option" procedure.

CICS initialization and shutdown

Under normal circumstances, tableBASE is started automatically on CICS initialization, via the PLT (PLTSI)—see the *tableBASE Installation Guide*, in the "Installing the CICS Interface option" procedure, for further information.

When a CICS region is terminated normally, tableBASE termination within the region is handled by the program DK1TSHUT, via the PLT (PLTSD). If a user issues an emergency shutdown of a CICS region, PLTSD is not executed, and tableBASE termination is handled by DK1TCRM. Normal region termination, handled by DK1TSHUT via the PLTSD, will ensure that all transactions are completed before proceeding with tableBASE termination for the region.

Transaction TBST

Under CICS, tableBASE is now initialized when DK1TCIN is invoked through transaction TBST (or the PLT). In earlier releases, it was initialized by the first application call to tableBASE. TBST can also assist in resolving tableBASE software problems in a CICS region.

The TBST INIT function initializes the tableBASE software in the CICS region, essentially in the same manner as during CICS initialization. The TBST INIT function reloads all modules and allocates the TSR. The TBST INIT function does not preload any tables.

Normal completion is indicated by the following messages:

- tableBASE 7 - Resource manager initializing
- tableBASE Resource manager initialized tableBASE

Note: Under CICS/TS 1.3, message "Phase-in of DK1TCRM not allowed by CICS, current in-storage copy used." may be displayed. This message may be safely ignored.

The function TBST TERM terminates tableBASE in the region. (Before using TBST TERM, contact DataKinetics Technical Support.) Ensure that no transactions are using the tableBASE software before starting this transaction. The TBST TERM function closes all tables in the local TSR (it does not store them). The TSR and all tableBASE control blocks and modules are deleted from the region.

Normal completion is indicated by the following messages:

- tableBASE Deactivating

- tableBASE Deactivated

Restrictions

The TBST transaction is supported for stopping and restarting tableBASE within a CICS region with restrictions. Stopping tableBASE results in the TSR being released. Restarting tableBASE results in a new, empty TSR being created. This transaction is only supported if all transactions accessing tableBASE have completed or have been purged. If any transactions are still accessing tableBASE when this transaction executes, unpredictable results, including abends and blocked transactions, can result. Recovery could entail recycling CICS and any affected VTS-TSRs.

The tableBASE CICS interface does not support the loading of DK1TROTC from LPA.

tableBASE CICS modules must be loaded from DFHRPL.

Transaction TBDR

Transaction TBDR is covered in detail in the *tableBASE Programming Guide*.

The TBOPT dataset

If a VSAM TBOPT file is to be used, it has to be defined in CICS. With full support of QSAM TBOPT files since Version 6, tableBASE has to determine how to access the TBOPT dataset. If it is defined under CICS, then CICS I/O is used to read it; if not, and the DD is present, MVS I/O will be used. Note that a CICS TBOPT definition is not sent out.

Starting with Version 6, tableBASE does not initialize if there are apparent user errors in defining TBOPT. If the CICS definition is found, it is assumed that TBOPT is desired, and an error is generated if the VSAM TBOPT file cannot be found.

Chapter

9

VTs (Virtual Table Share)

This chapter discusses the administrative issues that arise when using the VTS (Virtual Table Share) product with tableBASE. The VTS product can also be used in conjunction with the optional VTS Gate product. This chapter provides background information needed by the tableBASE administrator to set up and maintain VTS's using VTS Agent.

If you are licensed for the VTS Gate product, please refer to [VTSGate \(VTS with Enhanced Security\)](#) on page 57 for additional information on how to set up and maintain VTS's with added security.

Note: VTS is an optional DataKinetics product which is used with tableBASE in order to allow many regions/applications to share in-memory tables.

Overview

A VTS-TSR is a shared tableSPACE Region (TSR), and resides in a shared dataspace. A VTS-TSR allows shared access to tableBASE tables across multiple batch, TSO, CICS TS, DB2 SPAS and IMS TM regions in a single LPAR or z/OS image. Each VTS-TSR has a VTS Agent program which manages the Data Space.

With Version 5 tableBASE VTS-TSRs were read-only. With Version 6 this access was improved to allow full read and write access to tables in a VTS-TSR. With Version 7, VTS Gate features provide added security to manage access to the VTS-TSR; the VTS dataspace can also be initialized as read-only or read-write.

By loading one copy of data into a VTS-TSR, the I/O, system paging, and overall memory requirements can be reduced. A VTS-TSR can be referenced by any programming language that uses the standard IBM calling protocols, including C and C++.

You may have several differently named VTS Agents active simultaneously, allowing you to access multiple shared VTS-TSRs, each of which has a maximum size of approximately 2 GB. (Your installation may impose limits on the number or maximum size of Data Spaces.)

Applications accessing a VTS-TSR may use any mix of batch, TSO, CICS TS, DB2 SPAS or IMS TM interfaces. If a table is opened for write in a VTS-TSR by any user application, it is available for update by any other users of the VTS-TSR unless this is restricted using VTS Gate. The table cannot, however, be opened for write in any other TSR. It is the users' responsibility to coordinate updates to the table in the VTS-TSR. The LOCK-LATCH capability can also be used as a means to control update access to specific tables in a VTS-TSR.

New Features in Version 7

The optional VTS Gate interface, new with Version 7, provides enhanced security features like SAF interface and hardware key protection for VTS-TSRs. VTS-TSRs can also be initialized in read-only or read-write mode. These new features provide the ability to better manage VTS-TSR access.

Also with Version 7, a VTS-TSR now contains a generation number. This generation number increases each time a VTS-TSR is restarted or switched to read-only or read-write mode. Switching of a VTS to read-only or read-write mode is relevant only if you have licensed VTS Gate.

The generation number for a VTS is contained in VTS-DATA and can be obtained using the LV command. For more information on VTS-DATA and how to use the VTS generation number, please refer to the *tableBASE Programming Guide*.

For more information on VTS Gate product features and administration, please refer to [VTSGate \(VTS with Enhanced Security\)](#) on page 57.

Changes from Version 6

The significant VTS product feature changes from Version 6 are listed below. Please review the sections below carefully for a complete understanding of the product features and its differences from Version 5 VTS.

Note: VTS Agent from the current version cannot be used with tableBASE from prior versions.

Update commands in a VTS-TSR

tableBASE Version 6 introduced the read-write VTS-TSR which provides the ability to read and write to a VTS-TSR using a direct connection i.e. where the VTS name resides in the TB-SUBSYSTEM field of the TB-PARM. Write passwords on tables are enforced when tables are opened from a tableBASE library. However, passwords only protect tables in the library. Once a table is opened into a TSR (local or VTS), LOCK-LATCH or some other means of security process is needed to protect the table in the TSR.

Read-only VTS-TSRs are supported by the optional VTS Gate product. Read-only access to a VTS-TSR on a table-by-table basis is also supported through links between a table in the local TSR and the corresponding table within a VTS-TSR (see [Linked tables in VTS](#) on page 53).

Version 5 tableBASE VTS-TSRs were read-only.

Transaction affinities

The use of a read-write VTS-TSR, which was unavailable with prior versions of tableBASE, allows the removal of CICS transaction affinities, imposed by using a local TSR.

TBPOSTV

With the introduction of a read-write VTS-TSR, TBPOSTV (from Version 5) is no longer provided. DK1TDRV (DK1TDRV) has been enhanced and now has the capability of executing commands previously initiated by TBPOSTV. DK1TDRV eliminates the complexities inherent in TBPOSTV.

TBOPTV

The TBOPT file is the recommended method for providing parameters formerly set by TBOPTV (from Version 5). However, TBOPTV is maintained for backward compatibility (TBOPTV, if present, is logically considered as if it were concatenated after TBOPT).

SWITCHES parameter

In Version 5, the VTS product had its own settings for the SWITCHES run time option. Starting with Version 6 the VTS product does not have its own SWITCHES settings; it inherits the settings from the interface corresponding to the application being run—Batch, IMS, CICS, DB2 SPAS, etc.

The default settings for the SWITCHES option in Version 5 VTS was NNNNN, while the default settings for the Batch, IMS, CICS, and DB2 SPAS interfaces differ from these values (refer to the installation chapters applicable to the interface). Therefore an application from Version 5 that may have not abended when the VTS-TSR was down (due to a 1072 error) may abend in later versions, unless the SWITCHES setting is changed from the default value to N. Other differing settings may cause other behavioral differences unless changed from their Version 5 default values.

VTS Agent operations

This section describes the operation of the VTS Agent. [Figure 9-1](#) on page 49 is a sample VTS Agent start-up JCL. In the sample, the VTS Agent to be started will be known as DKLA.

```
//VTS      EXEC PGM=DK1VAGNT,TIME=1440
//STEPLIB  DD DISP=SHR,DSN=*YOUR.PREFIX*.TBASE.AUTHLIB      <=====
//SYSUDUMP DD SYSOUT=*
//TBOPT    DD *
LISTOPTIONS=Y
VTSNAME=DKLA
TSRSIZE=50M
/*
```

Figure 9-1: Sample VTS Agent Startup JCL

Note: No initializing command file is executed; this is different from Version 5.

Startup

The VTS Agent may be initiated either as a batch job, by submission of the start-up JCL, or as an MVS Started Task (STC). The STC could be included in the IPL or daily start-up procedures

for your installation. The VTS Agent must be run from an authorized library. In the startup JCL shown in [Figure 9-1](#) on page 49, the TBOPT data set is of special importance.

Note: The tableBASE Program Call Server must be initialized before starting a VTS-TSR. See [Initialization of the Program Call Server](#) on page 33.

Initialization of a VTS-TSR

Starting with Version 6, the VTS-TSR no longer processes the command file, CMD, during initialization. To process commands, check for the following message that indicates that the VTS is ready:

```
DK100600I tableBASE Vvrm VTS vtsname initialized
```

Once this message is received, you can use DK1TDRV to execute the necessary commands. [Figure 9-2](#) on page 51 shows an example of the JCL.

TBOPT parameters

The TBOPT DDNAME is used to set execution-time parameters that override the defaults defined in the VTS install process. For detailed descriptions of these parameters, see [tableBASE Runtime Options](#) on page 135.

The parameters that you may wish to change are:

1. VTSNAME=xxxxxxx where xxxxxxx is the one to eight character name used to identify the VTS-TSR created by this VTS Agent. VTS users use this name to make a logical connection to the VTS Agent. It must be unique for each VTS-TSR created within a single MVS image. The default value is DKL1.
2. TSRSIZE=nnnnnnn (this replaces the Version 5 VTSSIZE parameter). nnnnnnn specifies the amount of virtual storage (Data Space size) to be allocated for this VTS-TSR. It needs to be large enough to contain all open tables and control information for this VTS-TSR. The delivered default is 10M. For format details, see [TSRSIZE—tableSPACE Region Size](#) on page 148.
3. MAXNMTAB=nnnnn (VTSNMTAB is supported for compatibility with Release 5.2.). This number allows you to limit the number of tables opened in this VTS-TSR in order to reduce the overhead of the open table index constructed in the TSR. Attempts to open more tables will receive an error. The default is calculated to be the maximum number of tables that could possibly be opened in the TSR.
4. STROBE=nnnnnnnnnn which specifies the interval after which to collect VTS-TSR usage data. This information will be collected and reported if STROBE is set to a non-zero value. If the value is set too low this may result in a large amount of strobe output which may exceed the installation maximum for job output. The default is 0, which results in no usage data being collected. Starting with Version 7, this option is used in conjunction with STROBEMETHOD (see [STROBEMETHOD](#)).
5. STROBEMETHOD=n which is new with Version 7. This option is used in conjunction with STROBE (see [STROBE](#)). If n is 0, 1 or 2, STROBE specifies the call count after which to collect VTS-TSR usage data. If n is 3, STROBE specifies the number of minutes after which to collect VTS-TSR usage data. The default is 0.

6. SWITCHES=xxxxxxx which are one-byte codes that are employed to alter the operation of the tableBASE interface used by your programs. Applications using the TBCALLV or TBASEV interfaces may wish to override existing Status Switches at run-time. For more detail, see the *tableBASE Programming Guide* section on "VTS User access" under "Environmental interfaces".

VTS-TSR table loading sample JCL

Since tables can be opened into the VTS-TSR by any region that accesses it, there may be no need for preloading of tables into the VTS. However, if the VTS-TSR must be preloaded with specific tables, as it was in Version 5, a separate job executing DK1TDRV will need to be run. The job executing DK1TDRV can be cued off the message that is returned indicating that the VTS-TSR is now available:

```
DK100600I tableBASE Vvrm VTS vtsname initialized
```

A similar CMD file that was used as input (to TBPOSTV) in Version 5 can be used as input (to DK1TDRV) in later versions. Sample JCL is shown in [Figure 9-2](#) on page 51.

```
//VTSLOAD EXEC PGM=DK1TDRV
//STEPLIB DD DISP=SHR,DSN=*YOUR.PREFIX*.TBASE.LOAD <=====
//MAINLIB DD DISP=SHR,DSN=*YOUR.PREFIX*.MAINLIB <=====
//SYSUDUMP DD SYSOUT=*
//TBDUMP DD SYSOUT=*
//SYSOUT DD SYSOUT=*
//CMD DD *
SS,DKLA
OR,ABCD* Open for read all tables starting with letters ABCD
OA,TBL* Open any tables,
```

Figure 9-2: Sample JCL for loading tables

VTS-TSR refresh process

Since tables can be updated, closed, and reopened in the VTS-TSR by any region that accesses it, there may be no need for a separate refresh process in post Version 5 releases. However, if the VTS-TSR must be refreshed external to the applications which access it, DK1TDRV must now be used to implement the same functionality.

The VTS-TSR refresh process is, as it was in Version 5, the execution of a series of commands which may or may not include the RF command (refresh table). For more information on the RF command, see [VTS commands](#) on page 52 and the chapter on "tableBASE commands" in the *tableBASE Programming Guide*.

[Figure 9-3](#) on page 52 shows example JCL for a job that refreshes a VTS-TSR—in this case, closing some tables and refreshing others. Note that all other tables in the VTS-TSR are not affected.

```
//VTSRFRSH EXEC PGM=DK1TDRV
//STEPLIB DD DISP=SHR,DSN=*YOUR.PREFIX*.TBASE.LOAD <=====
//MAINLIB DD DISP=SHR,DSN=*YOUR.PREFIX*.MAINLIB <=====
//SYSUDUMP DD SYSOUT=*
//TBDUMP DD SYSOUT=*
//TBDUMP DD SYSOUT=*
//SYSOUT DD SYSOUT=*
//CMD DD *
SS,DKLA
CL,XX*
RF,TBL* Refresh any tables starting with letters TBL
/*
//
```

Figure 9-3: Sample JCL for VTS refresh

VTS Agent shutdown

The VTS Agent may be shutdown normally by issuing, from an MVS console, a stop command to the VTS Agent job/STC. The form of the command is:

```
P jobname
```

Any user attempting to access a terminating VTS Agent receives error code 1072.

VTS Agent recovery

If the VTS Agent is terminated or abends during execution for any reason, the VTS Agent job/STC may be resubmitted to MVS. However, the VTS-TSR will be reinitialized and must be repopulated. If the VTS-TSR had been updated by users, these updates will have been lost. While the VTS Agent is reinitializing, applications will not be able to access tables in the VTS-TSR, and will receive error code 1072 indicating that the VTS-TSR is unavailable.

VTS commands

Starting with Version 6, a VTS-TSR can be accessed using the same commands, and operates in a similar manner, as the local TSR. The chapter entitled "tableBASE commands" in the *tableBASE Programming Guide* describes these commands in detail.

The chapter entitled "DK1TDRV command processor for Batch/TSO" in the *tableBASE Programming Guide* describes the commands implemented in the DK1TDRV utility. In addition, there are several DK1TDRV commands that are unique to the VTS Agent. They may be accessed by anyone using DK1TDRV. As the tableBASE Administrator, it is your responsibility to control access to these commands, if desired, and, if so, to take measures to restrict their use.

Some commands for DK1TDRV accept the asterisk (*) as a wildcard in the table name to allow operating on a set of tables. Please refer to the chapter entitled "TBRIVER command processor for Batch/TSO" in the *tableBASE Programming Guide* for a complete list of these commands.

Note: To minimize the time during which access to the table is delayed, the affected Data Table and any alternates are reloaded while accesses are continuing with the original open copy in the VTS-TSR. Only after the refreshed copy and any alternates are completely loaded will internal pointers be switched to the refreshed copy. Access is delayed for the instant the internal pointers are switched.

VTS operational considerations

This section provides VTS operational considerations.

Reducing virtual storage when moving tables into VTS

Significant reduction of virtual storage utilization is possible when accessing tables from a VTS-TSR. If your installation uses multiple IMS regions or multiple CICS regions that each require access to the same tables, without VTS the tables are replicated in the TSR local to each region. Since each IMS or CICS region can access tables from a common VTS-TSR, the storage required for local TSRs can be greatly reduced. If a local TSR is not required after moving all the tables into a VTS-TSR, the local TSR size can be reduced to the absolute minimum by setting TSRSIZE to 0. See [TSRSIZE—tableSPACE Region Size](#) on page 148.

Linked tables in VTS

A linked table (also known as a remote table) is created when a user issues a command to open a table that is already open in a VTS-TSR by specifying the VTS name in the VTSFIRST or VTSLAST run-time options or by specifying it in the LIB-LIST parameter. When the table is opened from the VTS name specified using any of these methods, the table entry in the local TSR is linked to the existing open table in the VTS-TSR. The table data is not duplicated in the local TSR.

Use of the VTSFIRST or VTSLAST options or the LIB-LIST parameter to access a table in a VTS-TSR (linked table) does not allow for update commands against the table. Updates to the table can only be achieved by using the VTS name in the TB-SUBSYSTEM field of the TB-PARM parameter to access the table.

A linked table is always treated as opened for read, even if the table in the VTS-TSR is opened for write. An attempt to open for write a linked table will fail with error code 13-5. A linked table can be identified by the use of an LT or GD command.

If all tableBASE calls in your application are to a VTS-TSR, then using the VTS name in the TB-SUBSYSTEM field of the TB-PARM parameter to access the VTS-TSR is more efficient than using VTSFIRST, VTSLAST or LIB-LIST. Use of the VTS name in the TBPARM completely bypasses the use of the local TSR.

Limitations

The use of linked tables has limitations. Most significantly, if you open a linked table within your local TSR, there will be no indication to you when the actual table is changed, deleted or replaced.

7/24 VTS

If the VTS-TSR is to remain operational for an extended period of time (for example, seven days a week, 24 hours per day) and tableBASE usage reporting (see [tableSPACE Reporting](#) on page 7) is enabled for the VTS Agent (STROBE is not = 0), ensure that the output file is large enough to capture all the strobe activity that you wish to see. If the file is not large enough, any strobe overflow records will be suppressed to ensure that the VTS can remain running. When this occurs, the following error messages will be displayed in the JESMSG LG of the VTS Agent job:

```
DK100280E QSAM ERROR: Function=PUT QSAM,File=TBTSRPT
DK100281E Strobe output suppressed
```

Capacity

Each VTS-TSR is a shared Data Space that may be defined to be as large as 2 GB. If additional capacity is needed, an application may refer to any number of VTS-TSRs consecutively, by changing the VTS name in the TB-PARM, or by using a suitable tableBASE Library List (LIBLIST) with VTS name entries and/or using VTSFIRST and/or VTSLAST run time options.

Loading a VTS-TSR

Library caching can reduce the time to load a VTS-TSR.

Library caching may be valuable when:

- the VTS-TSR is large
- the VTS-TSR is pre-loaded with a large number of relatively static tables (not frequently updated in the VTS-TSR)
- no other regions are updating tables in the same library during the VTS load operation (this could require a cache refresh, possibly increasing load time)

For more information about library caching, see [Library directory caching](#) on page 131.

VTS messages

In a VTS environment, specific VTS messages are written to the MVS console and to the JESMSG LG for the VTS Agent. These messages are in addition to the tableBASE messages displayed when processing commands. For a complete list of tableBASE and VTS-specific messages, see [tableBASE messages](#) on page 161.

Messages ending with an I are informational, and messages ending with an E or S will not allow the VTS Agent to continue.

Internal logic errors

The following message is displayed in the event of an internal logic error in the VTS Agent:

```
'DK100301S LOGIC ERROR; DUMP TAKEN TO TBDUMP'
```

Programs that are accessing the VTS agent will not abend. The programs will continue to run in their environment. The system will return an error code of 1072 on every attempt to access the VTS-TSR.

VTS user messages

tableBASE/VTS user error messages are the same as those issued by tableBASE when accessing a local TSR. User error codes are also the same, with the addition of error code 1072 (VTS access failed because the specified VTS is not available). Error codes are set in the ERROR and ERROR-SUBCODE fields of the tableBASE COMMAND-AREA parameter.

Chapter 10

VTSGate (VTS with Enhanced Security)

This chapter provides additional information for setting up and maintaining VTS-TSRs using the security features provided by VTSGate. For general administration information on VTS-TSRs, please refer to [VTS \(Virtual Table Share\)](#) on page 47.

Note: VTS Gate is an optional DataKinetics product which provides added security features for tableBASE/VTS.

Overview

VTSGate provides enhanced security for VTS-TSRs, specifically, Read-only, SAF interface protected (using security tools like ACF2, RACF, and Top Secret) and hardware key protected VTS-TSRs.

With a Read-only VTS, user applications are only allowed to read from the VTS. Updates to the VTS can only be performed by switching the VTS to read-write mode using an MVS-authorized program.

With an SAF interface protected VTS, access to the VTS is controlled by authorization levels granted using an SAF security tool. The authorization granted to the user controls their ability to read from, update, start up or shut down the VTS. An SAF protected VTS can be started up in read-only or read-write mode.

With a hardware key protected VTS, the VTS is built in a hardware key protected dataspace which can only be updated by MVS-authorized programs running in supervisor state and key 0. A hardware key protected VTS can only be started up in read-only mode.

A hardware key protected VTS can also be SAF interface protected, in which case, the VTS will be a Read-only VTS.

VTSGate management

Valid combinations of VTSGate security features

The valid combinations of VTSGate features that can be used to start up a VTS-TSR are:

- read-only
- read-only with SAF protection
- read-write with SAF protection

- read-only with hardware key protection
- read-only with SAF and hardware key protection

tableBASE commands by TSR access type

For the purpose of implementing VTSGate, tableBASE commands have been classified into three groups:

- a. those that read from a TSR
- b. those that write to a TSR
- c. those that do not access a TSR.

Detailed information on what these commands are and how they are classified can be found under Chapter 2 of the *tableBASE Programming Guide*. These groups of commands will be referenced in the sections that follow.

TBOPT parameters

In order to use any of the VTSGate security features, some VTS Agent run time option changes are required. The following is a summary of the run time options used for implementing the security features of VTSGate. For detailed information on these options, refer to [tableBASE Runtime Options](#) on page 135.

1. TSRACCESS specifies whether a VTS is read-only or read-write. RO indicates that the VTS is read-only. The default is RW.
2. RACF_VTS specifies whether a VTS is SAF interface protected. A value of Y indicates that the VTS is SAF protected. The default is N.
3. TSR_KEY_PROTECT specifies whether a VTS is hardware key protected. Any value from 1 to 7 can be used to specify a hardware key protected VTS. The default value of 0 specifies a VTS with no protection. If any value from 1 to 7 is used, then the option for TSRACCESS ([1](#) on page 58) must be set to RO and the option for STROBEMETHOD ([4](#) on page 58) must be set to 3.
4. STROBEMETHOD specifies when usage statistics will be collected for a VTS. If TSR_KEY_PROTECT is set to any value from 1 to 7, then this option must be set to 3 (for more details, see [Strobe reporting](#) on page 61). This option is used in combination with the STROBE run time option which specifies the interval for collecting the statistics.

Except for STROBEMETHOD, the above options cannot be modified using the TBOPT DD statement in the VTS Agent job or task. It can only be modified using the DK1V1134 source provided and relinking DK1VBASE.

For information on how to override the default for these parameters, please see the chapter on *"Installing the VTSGate product"* in the *tableBASE Installation Guide*.

VTSDATA

Information on the security status of a VTS-TSR, like whether it is SAF interface protected, hardware key protected, in read-write or read-only mode as well as its generation number can

be found in the parameter, VTS-DATA. This parameter is retrieved using the LV command by providing the VTS name in the TB-SUBSYSTEM field of the TB-PARM parameter.

For more information on VTS-DATA and the LV command, please refer to the *tableBASE Programming Guide*.

Access control setup for SAF protected VTS

The access control for an SAF protected VTS is set up using the security tool used by your installation.

When RACF_VTS is set to Y for a VTS, tableBASE will perform checks on the user's profile before allowing access to the VTS. This access is authorized by tableBASE based on a resource profile in the Facility Class.

The format of the resource profile used for authorization by tableBASE is:

DKI.VTS.lparname.vtsname where *lparname* is the name of the LPAR and *vtsname* is the name of the VTS-TSR.

The resource profile for the VTS can also fall under a generic profile in the Facility Class. If there is no resource profile in the Facility Class that the VTS falls under, the default for the class will be used; this is normally NONE which means no access to the VTS will be allowed by any user.

Access levels

When a user is granted access to a SAF protected VTS, it is the userid used to submit a batch job or to start up a region (CICS, IMS or DB2 SPAS) that is verified for sufficient authorization to access the VTS-TSR. For CICS, IMS and DB2 SPAS regions, any userid permitted to access the region, and while executing in the region, inherits the same access levels to the VTS as the user that started up the region.

[Table 10-1](#) on page 59 shows the type of access allowed for each access level granted on a SAF protected VTS-TSR.

Access Level	Access Allowed on a SAF Protected VTS
ALTER, CONTROL	Start up, shut down VTS; all tableBASE command groups (a, b and c) - update, read and none.
UPDATE	All tableBASE command groups (a, b and c) - update, read and none.
READ	tableBASE command groups b and c - read and none.

Access Level	Access Allowed on a SAF Protected VTS
NONE	tableBASE command group c - none.

Table 10-1: SAF protected VTS Access Levels and Permissions

Startup and shutdown

SAF interface protected VTS

As specified in [Table 10-1](#) on page 59, the VTS Agent for an SAF protected VTS, i.e. with RACF_VTS set to Y, can only be started up or shut down by a userid with ALTER or CONTROL access. However, this does not prevent any user with sufficient authorization from canceling the VTS Agent job or task as this action bypasses all tableBASE security checks.

If the startup or shutdown of a VTS with SAF protection is attempted by a user without sufficient authorization, the following SAF error message will be displayed in the JESMSG LG of the VTS Agent task:

```

ICH408I USER(userid) GROUP(groupname) NAME(username)
DK1.VTS.lparname.vtsname CL(FACILITY)
INSUFFICIENT ACCESS AUTHORITY
[FROM resource profile (G)]
ACCESS INTENT(CONTROL) ACCESS ALLOWED(access-allowed)

```

Note: Parts of the message in square parentheses ([]) will only appear if the resource profile for the VTS falls under a generic profile.

The following tableBASE messages will also be displayed for unauthorized VTS startup and shutdown respectively:

```

DK100608E Not authorized to start VTS resource:
DK1.VTS.lparname.vtsname

DK100639E Not authorized to stop VTS resource:
DK1.VTS.lparname.vtsname

```

Hardware key protected VTS

If the TSRACCESS option for a hardware key protected VTS is not set to RO during VTS Agent startup, it will be automatically forced to RO and the following message will be displayed:

```
DK100710W TSRACCESS forced to RO: TSR_KEY_PROTECT
```

If STROBEMETHOD for a hardware key protected VTS is not set to 3 (see [Strobe reporting](#) on page 61) during VTS Agent startup, the VTS Agent task will fail with the following message:

```

DK100709E StrobeMethod 3 required for VTS with nonzero
TSR_KEY_PROTECT

```

Table loading and updating for read-only VTS

With a Read-only VTS, only tableBASE read commands are allowed to operate on the VTS (group [a](#)). Hardware key protected VTS-TSRs as well as SAF protected VTS-TSRs started up in read-only mode fall into this category. In order to perform tableBASE update commands on the VTS (group [b](#)), which include commands for opening tables into the VTS, the VTS must be switched to read-write mode.

A Read-only VTS is switched to read-write mode using the VV (Vary VTS) command. This command can only be issued by batch programs linked with the linkage editor input, SETCODE AC(1) and run from an authorized library. It cannot be run from a CICS, IMS or WLM-managed DB2 SPAS region. If the VTS is also SAF protected, the user-id executing the command must have a minimum authorization of CONTROL to the VTS.

Executing the VV command from a batch program that is not authorized will cause an abend of S047. If an authorized program bypasses tableBASE controls in order to update a VTS, it will be successful i.e. it is the responsibility of the client to ensure that programs cannot be easily authorized.

When a VTS is switched to read-write mode, the job that performs the switch holds a read-write lock on the VTS and no other job or region can access the VTS until it is switched back to read-only mode. A VTS can only be switched back to read-only by the same job step that holds the read-write lock.

If a VTS is locked in read-write mode after the job that holds the lock terminates abnormally, it can be forced back to read-only mode by supplying a special request id for the VV command.

Detailed information on the VV command can be found in Chapter 3 of the *tableBASE Programming Guide*.

Each time a VTS is successfully switched, its generation number increases by 1. To continue to access a VTS that has its generation number changed, the START_TBUOW field in the COMMAND-AREA must contain an S. For more information on the START_TBUOW parameter, please refer to the *tableBASE Programming Guide*.

Note: If a library is used to open tables into a key-protected VTS before it is used to open tables into a non key-protected TSR, the DE command should be used to close the library, otherwise a S0C4-4 may occur.

Strobe reporting

For hardware key protected VTS-TSRs, since application programs will only have read access to the VTS, a special dataspace is acquired for the purpose of updating tableBASE statistics. Both tableBASE strobe and LT command processing will access this special area for these statistics. The run time option, STROBEMETHOD=3, specifies that this special area will be used for updating tableBASE statistics.

tableBASE strobe reports for any VTS using the setting of STROBEMETHOD=3 will be displayed by minute intervals instead of by call count. The number of minutes for this interval is specified using the STROBE run time option.

Licensing Messages

PC Server Messages

The following licensing messages for VTSGate will appear in the JESMSG LG of the tableBASE PC Server at different times.

1. At PC Server initialization:

```
DK100833I VTSGate licensed to <customer name>
```

2. In the warning period specifying when the license will expire:

```
DK100828W VTSGate license will expire <yyyy-mm-dd>
```

3. On the day when the license will expire after midnight:

```
DK100829W VTSGate license expires today <yyyy-mm-dd>
```

4. If there is a grace period for the license and the product is running in its grace period:

```
DK100826W VTSGate license expired <yyyy-mm-dd>; in grace period
```

5. When the license has expired and the product has stopped functioning:

```
DK100827W VTSGate license expired <yyyy-mm-dd>
```

VTSGate Messages

If there is no VTSGate license or the product has stopped functioning because the license has expired, any attempts to start up a VTS Agent with the following TBOPT options will fail:

- TSRACCESS=RO
- RACF_VTS=Y
- TSR_KEY_PROTECT=1,2,3,4,5,6 or 7

The following message will be displayed in the VTS Agent job that attempts to use any of the above options without a valid VTSGate license:

```
DK100901S Not licensed for VTSGate options
```

Chapter 11

VTSTManager

VTSTManager is a performance-enhancement product which augments the DataKinetics VTS and tableBASE core products. tableBASE and VTS provide the increased speed of mainframe applications and simplicity of maintenance that come with using table-driven applications. VTSTManager provides further improvements in performance, flexibility, security, and manageability.

For more information, see the *tableBASE Concepts and Facilities Guide*.

VTSTManager Features

The following is a brief description of the features provided with the VTSTManager software product.

Pre-loaded VTS-TSRs

Using VTSTManager, VTS-TSR tables can be mapped to Linear Data Sets (LDSs). On VTS-TSR initialization, the link to the LDS is set up, and data is available immediately to calling applications. The use of LDSs in this way improves performance by eliminating the time previously required at VTS-TSR startup to load table data row-by-row from the DASD-based library, and to rebuild all indexes.

Tip: LDS-backed Read/Write VTS-TSRs are **not** recommended.

Read-only VTS

With VTSTManager, it is now possible to create read-only VTS-TSRs. Read/write VTS-TSRs, useful when updating tables is a requirement, are less desirable when that data is intended for table-driven decision processing. Access time can be burdened by needless queuing and table locking—read-only VTS-TSRs eliminate the need for this type of overhead processing. Using read-only VTS access, such costly processing can be eliminated completely.

Refresh-only VTS

With VTSTManager, it is now possible to create refresh-only VTS-TSRs. Features of this type of VTS, fall between pure read-only and read/write VTS-TSRs.

If the tables in VTS are updated frequently, and a SWITCH process for every update is not feasible, refresh only VTS will be the right choice. The only update command allowed on this VTS is REFRESH.

Update Tables in the Background

Using VTSManger, VTS-TSRs can be updated in the background without affecting the performance of calling applications, and without the delays associated with VTS startup. In effect, you can synchronize live updates to your active tables.

Switching VTS-TSRs

Using VTSManger, VTS-TSRs can be switched in real time. Initiated by a single command, the switch effectively switches old data (in the original VTS-TSR) for updated data (in a new VTS-TSR). Calling applications (batch programs, transactions, etc.) still accessing the original VTS-TSR will be allowed to complete their tasks. New tasks will access the updated data in the new VTS-TSR.

Alias Names for VTS-TSRs

Calling applications can access VTS-TSRs using an alias name; the actual name of a VTS-TSR is unknown to the calling application. This feature supports the switching and background updating features.

Installing VTSManger

This chapter provides information to help the VTSManger administrator understand and confirm the correct installation of VTSManger for tableBASE 7.0.3.

Details of VTSManger Installation

To install and begin using VTSManger requires two steps.

- Install the necessary modules into an APF-authorized library.
- Activate the VTSManger license.

Install the Necessary Modules into an APF-authorized Library

If VTSManger is installed after installing tableBASE, the following modules will be delivered:

- DK1MEXEC
- DK1MBASE

These modules must be placed into an APF-authorized load library. Typically, this would be the same library where existing tableBASE modules requiring APF authorization reside.

Activate the VTSManger License

Among the modules delivered for VTSManger is a licensing module containing information about the expiration date of the VTSManger license. To notify the system with this information, the VTSManger LICENSE command must be run, or a restart of the PC Server must be executed. The LICENSE command is explained in more detail here: [LICENSE Command](#) on page 68.

The LICENSE command does not need to be specified every time a VTSManger command is issued. It only needs to be run for these special cases.

- You have received and installed the VTSManger product for the first time and do not want to recycle the PC Server.
- You have received an updated license module or a zap from DataKinetics and your license is expired.
- You wish to set a new or different warning period for the license. (The PC Server uses a 30-day default warning period.)

Administration of VTSManger

This topic discusses the administrative tasks and responsibilities that are associated with VTSManger for tableBASE 7.0.3 when it is implemented with DataKinetics tableBASE and VTS products. Background information is provided wherever it might be needed by the VTSManger administrator.

VTSManger Administrative Tasks

Depending on your specific role and responsibilities as a VTSManger administrator, little may be required for the day-to-day administration of the product. This information is covered in the [VTSManger Command Summary](#) on page 66.

VTS-TSR Access Modes

VTS-TSRs can be brought up in following modes.

1. Read-only TSR (TSRACCESS=RO)
2. Refresh-only TSR (TSRACCESS=RF)
3. Read-write TSR (TSRACCESS=RW)

Pre-loading VTS-TSR

One additional step that is required in the VTS Manager administration is the process of pre-loading the VTS-TSR to a VSAM LDS. Please refer DK1MLDSB member in the CNTL library supplied with VTS manager for the sample job.

We recommend using a local TSR with a VSAM LDS in read-write mode to pre-populate the VSAM LDS before starting the VTS-TSR in read only or refresh mode.

Sample JCL to bring up VTS-TSR in RF Mode

```
//VTS          EXEC PGM=DK1VAGNT
//STEPLIB      DD DISP=SHR,DSN=*your-auth-load-library*
//LDSTSR       DD DISP=OLD,DSN=*your-LDS*
//SYSOUT       DD SYSOUT=*
//TBDUMP       DD SYSOUT=*
//TBTSRPT      DD SYSOUT=*
//TBOPT        DD *
               VTSNAME=*your-vts-name*
```

```

TSRSIZE=16M
TSRACCESS=RF
LDS=Y
/*

```

Note: The DDNAME LDSTSR must have DISP=OLD for both TSRACCESS=RF and TSRACCESS=RW. DISP=SHR should be used for TSRACCESS=RO.

Note: Every time REFRESH is executed, only the table in VTS-TSR gets updated or refreshed, the VSAM LDS is not.

The only time the LDS gets updated is, when the VTS Agent is brought down normally with the MVS STOP command. This is to ensure that the VTS-TSR and LDS are in SYNC, so that when the VTS-TSR is brought up again it is with the latest data when it is recycled.

Following messages in the VTS agent console indicate a normal termination of the VTS agent job which synchronize the data from VTS-TSR to LDS.

```

DK100634I STOP command received, V703 VTS XXXXX terminating
DK100417I TSR successfully saved on LDS

```

If the VTS agent abnormally terminates, the LDS will not be updated.

The LDS must be pre-loaded again from the latest updated library before bringing up the VTS-TSR again.

DK1MEXEC Utility

This topic provides a list of VTSManger commands, describes their use and syntax, and demonstrates how to use the DK1MEXEC utility to submit those commands.

VTSManger Command Summary

VTSManger features five commands, which are summarized in this table:

Command	Description	Parameters
LICENSE	Set VTSManger License	PRODUCT, VERSION, WARNING
RESET	Reset Alias	ALIAS, VTSNAME
SET	Set Default	VERSION
SHOW	Display Status	Product Version Details
SWITCH	Switch Alias	ALIAS, FROMVTS, TOVTS

Table 11-1: VTSManger Command Summary

DK1MEXEC

DK1MEXEC is a utility program provided with VTSManger. VTSManger commands are submitted using DK1MEXEC.

Commands are read from an input file identified by the DDNAME CNTLCARD. CNTLCARD must contain 80 byte, fixed length records. DK1MEXEC terminates when end of file is reached on the CNTLCARD file.

Sample JCL to run DK1MEXEC

```
//STEP1   EXEC   PGM=DK1MEXEC,REGION=0M
//STEPLIB DD   DISP=SHR,DSN=*your-load-library*
//SYSPRINT DD   SYSOUT=*
//SYSOUT   DD   SYSOUT=*
//TBMSG    DD   SYSOUT=*
//CNTLCARD DD *
* A SAMPLE DK1MEXEC STEP
command keyword = value keyword = value ;
command keyword = value keyword = value ;
/*
```

Command Structure and Syntax

Every VTSMManager command sequence submitted through the DK1MEXEC utility program consists of a command and a list of one or more parameters. Each parameter consists of a keyword set to a corresponding value with an equal sign (=), and the entire command sequence is terminated by a semicolon (;).

An example of a properly constructed command sequence looks like this:

```
COMMAND KEYWORD1=VALUE KEYWORD2=VALUE;
```

The number of KEYWORD=VALUE parameters you can specify depends on the command. All required and optional parameters for every command are described later in this chapter.

The following rules apply to DK1MEXEC command sequences:

- Each command sequence must begin on a new line.
- Valid keyword=value parameters may be specified in any order.
- Commands and keyword=value parameters must be separated from each other by at least one blank.
- Command sequences may span multiple lines, and any number of blanks may be used, provided that commands, keywords, and value fields are not split up.
- The character set for command sequences is limited to the uppercase letters A-Z, the numbers 0-9, and the national characters @, #, and \$, except for comments.
- A command sequence must be terminated by a semicolon (;). Anything following a semicolon on the same line is treated as a comment.
- Any input statement starting with an asterisk (*) in column one is treated as a comment.
- Columns 73-80 of all input statements are ignored.

The following rules apply to VTS names and alias names in VTSMManager parameters:

- Lower case letters are not allowed.

- Names must be one to eight characters in length. Names shorter than eight characters will be automatically padded on the right with blanks.
- The first character of a name must be an upper-case letter (A-Z) or a national character (@, #, or \$).
- Characters of a name after the first must be an upper-case letter (A-Z), a number (0-9), or a national character (@, #, or \$).

VTSManger Commands

VTSManger's commands are described below in detail.

Note: In order to use the LICENSE, RESET, and SWITCH commands, DK1MEXEC must be run from an APF-authorized library as recommended in the *tableBASE Installation Guide*.

LICENSE Command

The LICENSE command sets the VTSManger License, and optionally sets a number of days prior to the expiration of the license for which warning messages will be issued. The LICENSE command must be issued whenever a new emergency licensing module is installed. It may be run any time a different expiration warning period is desired. The LICENSE command cannot be used to change the date of the license expiration.

The PC Server checks the license daily at midnight. This command is either optional or for updating the license without waiting for the PC Server to recheck.

Parameter	Description
PRODUCT=(product name)	Required parameter. The product name for VTSManger is VTSMGR
VERSION=(product version)	Optional Parameter. The Default is V7.0.3
WARNING=(number of days)	Optional parameter. Number of days before license expiration to start receiving warnings. Default is 30. Values not in the valid range of 1-60 will be set to the default.

Table 11-2: LICENSE

RESET Command

The RESET command switches an ALIAS back to its original VTS. Optionally, the ALIAS can be validated against a VTS, and if the ALIAS is not currently associated with the named VTS, the resetting of the ALIAS will not occur.

Parameter	Description
ALIAS=(alias name)	Required parameter

Parameter	Description
VTSTNAME=(VTS Name)	Optional Parameter. If specified, RESET won't occur unless the alias is currently associated with the VTS name. If this parameter is omitted, the RESET proceeds unconditionally.

Table 11-3: RESET**SET Command**

The SET command sets default values which are used by subsequent commands.

Parameter	Description
VERSION=(product version)	Required parameter.

Table 11-4: SET**SHOW Command**

The SHOW command displays the status of a product license for a product version, and optionally, the status of VTSES.

Parameter	Description
PRODUCT=(product name)	Required Parameter. The product name for VTSManger is VTSMGR.
VERSION=(product version)	Optional parameter. Default is V7.0.3
DETAILS=(YES NO)	Optional parameter. Default is NO. If specified with a value of YES, details of all known VTSES will be written to the job's output file. Otherwise, only the status of the product license will be externalized.

Table 11-5: SHOW**SWITCH Command**

The SWITCH command is used to assign an ALIAS to VTS for the first time.

The SWITCH command can also be used to switch an ALIAS from one VTS to another VTS. In order for SWITCH to succeed, any VTSES which are specified must be running.

Parameter	Description
ALIAS=(alias name)	Required parameter

Parameter	Description
FROMVTS=(old VTS name)	Optional parameter. If the value specified is not currently associated with the alias specified, the switch will not occur.
TOVTS=(new VTS name)	Optional parameter. If omitted, the SWITCH command behaves identically to RESET, with FROMVTS serving as an optional validation check if specified.

Table 11-6: SWITCH

Command Sequence Examples

The following command sets the license for version V703 of the VTSManger product, with a warning period of 45 days before expiration:

```
LICENSE PRODUCT = VTSMGR VERSION=V703 WARNING = 45 ;
```

The following command sets the default version to V703.

```
SET VERSION = V703 ;
```

The following command initially assigns an alias name PROD to a VTS named VTSDEV:

```
SWITCH ALIAS = PROD TOVTS = VTSDEV ;
```

The following command switches an alias named PROD from a VTS named VTSDEV to a VTS named VTSQA:

```
SWITCH ALIAS = PROD FROMVTS = VTSDEV TOVTS = VTSQA ;
```

The following command resets the alias named CURMONTH to its original VTS, regardless of the VTS with which it is now associated:

```
RESET ALIAS = CURMONTH ;
```

The following command displays the status of the VTSManger product:

```
SHOW PRODUCT = VTSMGR VERSION = V703 ;
```

The following command displays the status of the VTSManger product and the status of all VTSes defined to VTSManger for Version V703:

```
SHOW PRODUCT = VTSMGR VERSION = V611 DETAILS = YES ;
```

TBMSG Report

DK1MEXEC writes one report to an output file named TBMSG.

The TBMSG report displays the control statements and any associated messages.

Example of TBMSG Report for DK1MEXEC

```
DATE/TIME: 2026/06/09 10:08                      DK1MEXEC AUDIT REPORT
VTS Mgr V703 Licensed to: <Customer name shown here>
VTS Mgr V703 Expires: 2029/12/31
*** COMMENT:* SAMPLE DK1MEXEC COMMANDS
*** COMMENT:* COMMENTS START WITH * IN COLUMN ONE
*** REQUEST:
*** REQUEST:      SHOW          PRODUCT=VTSMGR ;
Licensing for product VTS Mgr
Product VTS Mgr V703 licensed
*** REQUEST:
*** REQUEST:      SWITCH  ALIAS=ALIASTWO TOVTS = VTSDEV;
*** ERROR   : Alias not found                      Return Code: 2311
                                   End of data- DK1MEXEC Utility ended
```

Things to consider while performing VTS Manager SWITCH

Batch

If a Batch job is in progress while a SWITCH is performed, the job will continue to use the original VTS-TSR until the end of the job.

You should not stop the old VTS-TSR until all the batch jobs have completed.

CICS

In Conversational Mode, SWITCH will not affect the current transaction in progress.

In Pseudo-Conversational Mode, SWITCH will take effect when the current transaction RETURNS control back to CICS and resumes.

If a transaction accesses a VTS-TSR using a linked table (using VTSFIRST, VTSLAST or the VTSname in a Library list (ML command with VTS:xxxx)) and a VTS SWITCH occurs, the SWITCH will not be recognized until the linked table is closed and reopened by tableBASE or the VTS-TSR is taken down, returning a 1072 error code.

IMS

If a transaction program accesses a VTS-TSR with the VTSNAME name in the TB-PARM and a VTS SWITCH occurs while a conversational transaction is in progress, and if a different transaction accessing tableBASE, of any type, ends in the same MPR region, the conversational transaction(s) will access the newer VTS when the next segment in the conversational transaction occurs.

If a transaction program accesses a VTS-TSR using a linked table (using VTSFIRST, VTSLAST or the VTSname in a Library list (ML command with VTS:xxxx)) and a VTS SWITCH occurs, the SWITCH will not be recognized until the linked table is closed and reopened by tableBASE or the VTS-TSR is taken down, returning a 1072 error code.

If an abend occurs in the IMS MPR after a SWITCH, the next access to the linked table in that region will be from the newer VTS-TSR.

JCL Examples Delivered with VTSManger

Below is a list of the JCL Examples provided with VTSManger.

DK1MALIA sets an alias to a VTS-TSR for the first time.

DK1MLDSB defines a VSAM Linear DataSet (LDS) and populates it

DK1MLDSV runs the VTS agent, using a Linear DataSet (LDS)

DK1MLIC initializes or reinitializes the license

DK1MRSET resets the ALIAS of a VTS-TSR

DK1MSHOW shows the current status of VTSManger (license) and optionally all VTS-TSRs known to the system

DK1MSWTC switches the ALIAS of one running VTS-TSR to a different running VTS-TSR

DK1MUSER runs DK1TDRV to test a VTS-TSR under VTSManger

DK1MVCHK detects if an old VTS-TSR is still in use. (Please make sure you test this script thoroughly before using it in production.)

Chapter

12

tablesONLINE/CICS Architecture

The tablesONLINE/CICS software package is an optional DataKinetics product that provides a flexible, interactive front end for the tableBASE system.

Several special administrative considerations exist when using tablesONLINE (TBOL) in a CICS environment. This chapter, along with the next two, discusses the administration of a tablesONLINE/CICS environment, and provide background information needed by the tableBASE administrator to set up and maintain tablesONLINE/CICS.

Overview

tablesONLINE/CICS is optimized to reduce the need for Working Storage by utilizing a Constants module that is shared by all tablesONLINE users in this region. This module can be updated using the TBOLCNST table that is accessible from the **Administrator** menu.

Note: tablesONLINE/CICS requires that working storage be initialized to binary zeroes. Under LE, this is accomplished by ensuring that the run-time storage option for the CICS region is STORAGE=(00,...). Refer to IBM's LE Customization manual for guidance in creating a CEEUOPT or CEEROPT module.

tablesONLINE/CICS is also fully reentrant and can take advantage of the features offered by the read/write VTS Agent, if the optional VTS product is installed.

Administrative features include:

- Controlling access to the tables
- Maintaining several generations of tables so errors can be fixed by returning to a previous generation
- Maintaining copies of the distribution versions of tables available for use if all modified generations become corrupt
- Set up of read/write VTS

Library architecture

The System tables are maintained in four separate libraries. The System tables will be needed to access and perform many of the administrative tasks. Each of these libraries contains a related group of System tables. The organization of the System tables into the various libraries has been designed to permit the enforcement of RACF security rules.

You may create your own library structure to store your organization's tables and views, in addition to the libraries described here.

System library

The System library (TBSYSLB) contains all of the system Views that are used by tablesONLINE/CICS for table definition and utility functions. This library also contains tables that are used internally by tablesONLINE/CICS, like the Cursor Positioning table and Session ID table. The System library is read-only and can be shared by all CICS regions using tablesONLINE/CICS.

These Views and tables have write passwords and are not intended for public use. The exceptions are the Session ID table that is used to delete sessions, the TBSYSVTS table which is used to control which VTS-TSRs or local TSR to load/access tables from, the MRO table and the Constants table, all of which are described in [tablesONLINE/CICS Administration](#) on page 83.

Note: Contact tableBASE customer support prior to modifying other tables in the System library.

Application Control Table library

The Application Control Table library (TBACTLB) contains the Application Control table (TBOLACT), the User/Application Relationship table (TBOLUSAP) and the Cross-Authorization table (TBOLXAUT). These tables are used for system administration, security control and user set up. See [Figure 12-1](#) on page 77 for a block diagram view.

This library is also intended to contain any tables that you may require for your own internal security procedures.

Application-Driving Tables library

The Application-Driving Tables library (TBAPPLB) contains all of the application-driving tables for tablesONLINE/CICS. These include the initial tablesONLINE/CICS menus, message text, PF key definitions, description text, help text, command alias list, and library alias list.

tablesONLINE/CICS can be used to create custom applications by modifying existing application-driving tables and creating new ones. Although it is possible to store these customized tables in any library you choose, the library must be referenced as TBAPPLB for tablesONLINE/CICS to function properly.

Views are provided for both the Application Control Table library and the Application-Driving Tables library tables so that these tables can be edited using tablesONLINE/CICS.

Data Dictionary library

The Data Dictionary library (TBDICLB) is intended to contain a corporate data dictionary, if you use a dictionary. tablesONLINE/CICS can be configured so that the Views you create, as well as all tables containing corporate edit rules or codes, are stored in this library.

The Data Dictionary library initially contains only the TBOLM2M table. This table is used to assign Table Object names to View/Table combinations. This facility is described in detail in the *tablesONLINE/CICS User's Guide*.

Levels of privilege

The library table-access assignments are listed in [Table 12-1](#) on page 75:

Technical role	Typical responsibilities
System administrator	Editing tables in the: • System library • Application Control Table library • Application-Driving Tables library • Data Dictionary library
Application developers	Editing tables in the: • Application-Driving Tables library • Data Dictionary library
Application users	Automatic read-only application-driving tables

Table 12-1: Privilege levels

The process for setting user privileges, which involves editing the Application Control table, is discussed in detail in [tablesONLINE/CICS Administration](#) on page 83.

Types of data

The tables in the System library, Application Control Table library, and Application-Driving Tables library contain all of the information that tablesONLINE/CICS requires to configure itself and to find the libraries that contain the Data Tables it will operate on.

Data Tables are categorized into three groups: common corporate data, application-specific user data, and application-driving data.

Common corporate data

This group includes data that is used by multiple applications and users but may only be altered by authorized users or specialized applications. Often, all data of this kind is kept in one library and all applications are designed to look for data in that library.

Depending on your organization, you may wish to further divide this group into subcategories, such as corporate data, divisional data, and departmental data and set up a separate library for each subcategory.

Application-specific user data

This group includes data that relates to either one stand-alone application or a group of related applications. You may wish to keep tables of this type in a separate library that is then searched only by that one application or group of applications.

Application-driving data

This group includes data that is seen by the developer but not by the user. The user simply sees the data stored here as part of the system and does not manipulate it.

When setting up your environment, you may wish to collect all application-driving data in one central Application-Driving Tables library or you may wish to place it in specific Application-Driving Tables libraries. User data should not be placed in the Application-Driving Tables library

as this library is often the subject of special RACF security. If this library becomes corrupt, tablesONLINE/CICS will not launch.

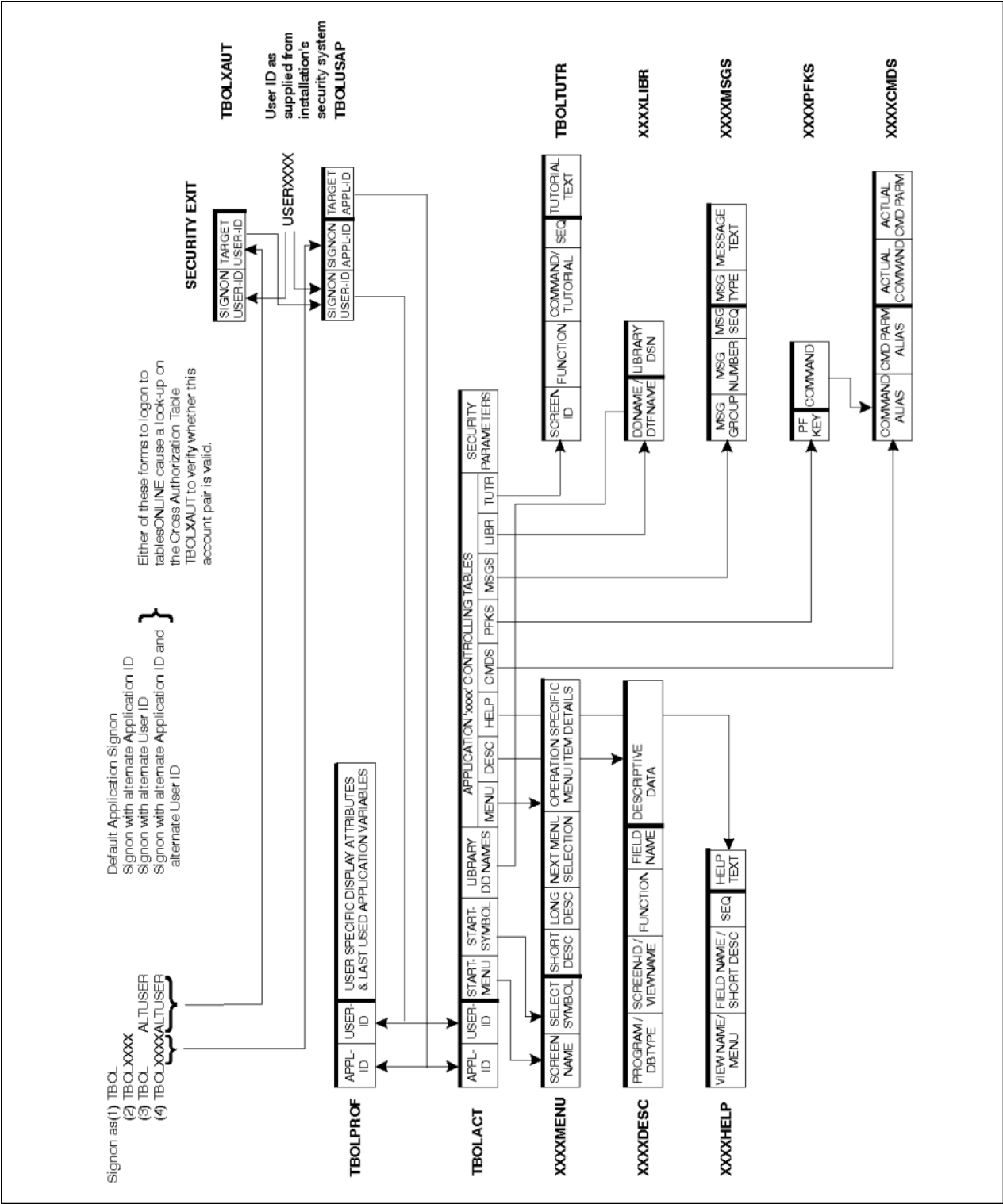


Figure 12-1: Signon and Application Control relationships

Chapter

13

tablesONLINE/CICS Security

This chapter covers security considerations for the tablesONLINE/CICS software product. Dataset security for tableBASE libraries is provided by other products such as RACF, ACF2 and TSS. The chapter describes the system exits used for signon security. Security below signon—at the command, table, row, or field level—is applied using exits controlled through the View. Creation and use of exits at the lower levels is described in the *tableBASE Programming Guide*.

The tablesONLINE/CICS software system is designed to allow comprehensive security using exits intended for this and other purposes. The security exit programs listed in [Table 13-1](#) on page 79 can be called for various tablesONLINE/CICS operations:

Security exit program	When called
SYSTEM	At signon/signoff, for tablesONLINE/CICS access control.
COMMAND	At the time of command line processing.
TABLE	At table open, refresh, store and close, for table-level access control
ROW	During data input and output for row-specific processing
FIELD	During data input/output for setting attributes such as DARK or PROTECT based on data values.

Table 13-1: Security exit programs

The signon security exit

Signon security is controlled by an exit routine and three tables. A security module, TBDKUSID, is included on the installation tape. This module is also supplied in COBOL source code so that you can modify it to suit your installation and replace the supplied module with your own customized version. The three security tables, TBOLACT TBOLXAUT and TBOLUSAP are stored in TBACTLB, the Application Control Table library, separate from other tables for security purposes.

Update access to this library is restricted to the tableBASE administrator. Therefore, the library may be used as a repository for any other tables requiring the same access authority. Each of the security tables is described in more detail in this chapter.

The security exit routine receives control immediately when tablesONLINE/CICS is invoked. Each user is assigned an application identifier, (up to four characters in length) and a user identifier, (up to eight characters in length), as shown in [Figure 13-1](#) on page 80. At signon, a user types these optional identifiers after the transaction code on the first line of the screen. TBDKUSID will determine if the information entered is valid and, if so, will allow signon to continue. If the information is invalid, signon will be aborted.

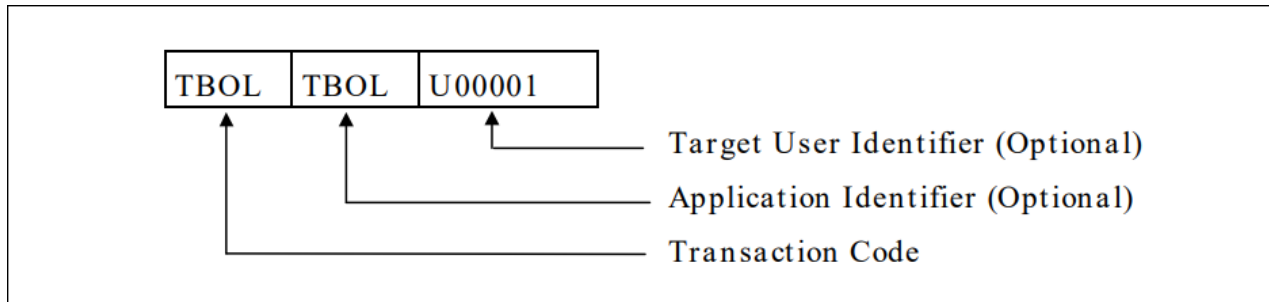


Figure 13-1: Signon identifiers

The Cross Authorization table

If the target user identifier field is non-blank, the first thing the supplied security exit TBDKUSID does is access the Cross Authorization table, TBOLXAUT. This table indicates whether a user is authorized to sign on to tablesONLINE/CICS using another user ID. This allows an application developer to sign on as an end user for testing purposes, or a group administrator to access the menus of users under his or her supervision.

When a value is entered, the exit assumes that this user ID is different from the identifier used to signon to CICS. TBOLXAUT is accessed to verify that the user who is signed on to CICS under one user ID has the authority to access tablesONLINE/CICS using another ID. If there is an entry in the Cross Authorization table indicating that the user does in fact have that authority the signon procedure continues.

If your CICS system does not use signon security, the table TBOLXAUT can be set up to translate whatever EXEC CICS ASSIGN USERID returns into an acceptable target user ID as entered when signing on to tablesONLINE. For example, if CICS returns the value 'CICSUSER', then an entry of **CICSUSER** for the signon user ID and **TBUS0001** for the target user ID will allow TBUS0001 to sign on to tablesONLINE.

The User/Application Relationship table

The security exit will then access the User/Application Relationship table, TBOLUSAP, to verify that the user who is signed on is allowed to access the specified application. The user-application pairs defined in this table allow you to control access to user applications. [Table 13-2](#) on page 81 lists the type of entries that will be contained in the User/Application Relationship table:

Signon user ID	Signon appl ID	Target appl ID	Description
		TBOL	The default application if no user/application pair is found.
U00001		ADMN	The default application for user U00001.
U00001	TBOL	TBOL	The default application is optionally available to user U00001 if TBOLT BOL is entered.
U00002		PAYR	PAYR is the only application available to user U00002.

Table 13-2: User/Application Relationship entries

At installation, the User/Application Relationship table is set up to allow the administrator to perform the initial user setup for tablesONLINE/CICS. After installation TBOLUSAP contains the entry listed in [Table 13-3](#) on page 81.

Signon user ID	Signon appl ID	Target appl ID	Description
		ADMN	The Administration Application (Intended only for initial administrative access).

Table 13-3: Installation entry

To restrict access to the administrative functions of tablesONLINE/CICS, change the entries in the TBOLUSAP after installation to the entries listed in [Table 13-4](#) on page 81.

Signon user ID	Signon appl ID	Target appl ID	Description
		TBOL	The default application (generally for instructional use)
XXX1	ADMN	ADMN	The Administrators application for USER XXX1
XXX2	ADMN	ADMN	The Administrators application for USER XXX2

Table 13-4: Post-installation entries

The first entry is set up to allow first-time users to access and become familiar with tablesONLINE. The additional two users XXX1 and XXX2 are intended to be used as user IDs by

the tableBASE administrator(s). In order to sign on to the **Administrator** menu, you must enter TBOLADMN.

The Application Control table

The goal of the security exit is to return both an application ID and a user ID to tablesONLINE. Immediately after the security exit processing, the application ID and user ID are used to retrieve controlling information from the Application Control table or TBOLACT. This table is used in system administration, security control, and environment setup for users. Detailed information on the setup of the Application Control table is provided in [tablesONLINE/CICS Administration](#) on page 83.

Note: If you have never set up a tablesONLINE/CICS environment before, please see [Setting up an organization](#) on page 153. You will find an overview of the process and logic required to set up an environment.

Invoking signon security

Signon security is automatically invoked by tablesONLINE. The default signon security program name, TBDKUSID, may be changed by modifying the table TBOLCNST, which can be accessed from the **Administrator** menu.

If the program name is changed from TBDKUSID to SECURITY, the security exit program is not called and complete control must be managed by TBOLACT.

Chapter

14

tablesONLINE/CICS Administration

The tablesONLINE/CICS software provides a facility to assist with the administration of your system. It provides options for setting up new users, creating new applications, customizing your environment, and monitoring your system.

When you sign on to tablesONLINE/CICS as the administrator, the first screen that is displayed is the **Administrator** menu, as shown in [Figure 14-1](#) on page 83. Since all the screens in tablesONLINE/CICS are modifiable, your screen may not appear exactly as shown.

```
tablesONLINE Administrator ----- Administrator -----  
COMMAND ==>  
  
To select, enter number/symbol on command line:  
  
D      DEVELOP APPLICATION      - Create and Develop Table Applications  
M      EDIT MRO TRAN IDS        - Add/Change Sets of tablesONLINE Transaction IDs  
VTS    EDIT VTS ASSIGNMENT      - Add/Change/Delete DDname - VTS Assignments  
1      EDIT APPL. CONTROL       - Add/Change/Delete Applications and/or Users  
2      DELETE SESSIONS          - Delete Active tablesONLINE Sessions  
3      COPY APPLICATION         - Copy an Application's Controlling Tables  
4      EDIT USER PROFILES       - Add/Change/Delete Items on User Profile Table  
5      EDIT HELP TABLES        - Edit tablesONLINE Help Tables  
6      EDIT TUTORIAL TABLE     - Add/Change/Delete tablesONLINE Tutorial Table  
7      EDIT X-AUTHORIZATION      - Add/Change/Delete Cross Authorizations for Users  
8      EDIT USER APPL TABLE    - Edit the User/Application Relationship Table  
9      EDIT CONSTANTS           - Add/Change Sets of tablesONLINE System Constants  
  
Enter HELP at any stage for help within tablesONLINE.  
Enter PF for program function key assignments.  
Enter X to suspend tablesONLINE and return to CICS.
```

Figure 14-1: Administrator menu

The options in this menu allow you to perform the following administrative functions:

To set up new users:

- (1) Edit Application Control table
- (8) Edit User/Application Relationship table
- (7) Edit Cross-Authorization table

To create new applications:

- (3) Copy application
- (D) Develop application

To customize your environment:

- (5) Edit Help tables
- (6) Edit Tutorial table
- (M) Edit MRO Transaction IDs
- (9) Edit constants
- (4) Edit user profiles

To monitor your system:

- (2) Delete sessions

To modify the TBSYSVTS table:

- (VTS) Edit VTS assignment

Note: All of these options are discussed in detail in this chapter, except the Develop application (D) option. Although we will discuss this option briefly, it is the main focus of the *tablesONLINE/CICS User's Guide*.

Set up new users

There are three administrative options that relate to setting up new users.

- To define a new user to your tablesONLINE/CICS system, you must place an entry in the Application Control table (TBOLACT). This table allows you to define the applications and the portions of the system to which a user will have access.
- You must place an entry in the User/Application Relationship table (TBOLUSAP). This table allows you to map signon IDs to actual target applications within tablesONLINE.
- The Cross-Authorization table (TBOLXAUT) allows you to define IDs that will allow one user to sign on to the system as another user. Placing entries in this table is optional although it may be useful for administrative purposes.

Application Control table

The Application Control table (TBOLACT) contains a list of all users, tablesONLINE/CICS applications, and the information required to start up a custom application at signon time.

To add an entry for a new user to the TBOLACT:

1. Select 1 from the **Administrator** menu (see [Figure 14-1](#) on page 83) and press <Enter>. The system displays the **Edit Application Control table** screen, as show in [Figure 14-2](#) on page 85.

```
tablesONLINE Administrator ----- Edit the Application Control Table -----  
COMMAND ==>  
  
Please indicate the table required by entering the parameters below.  
Data Table Library          ==> SECURITY.CONTROL.LIBRARY  
View Library (If different) ==> SYSTEM.TABLES  
Table Object==> TBOLACT  
  
Generation ==> 1  
  
Enter row key for direct search or positioning on the edit table screen.  
  
APPLICATION ID      ==>  
USER ID             ==>
```

Figure 14-2: Edit Application Control table

The information on this screen that identifies the name and location of the Application Control table is protected and may not be modified. It is possible to modify the generation of the table to be edited in the event that you wish to return to a previous generation of the table. Two generations of this table are maintained. To display the **Edit Table** screen and the list of rows currently in the Application Control table, press <Enter>.

2. To create a new entry in this table, copy an existing row to use as a template.
3. Use the tab or new-line keys to move to a row that is similar to the new row you wish to create.
4. Type "n" in the left column and press <Enter>.

The selected item now appears in the **Edit Row** screen for editing. On your terminal, the fields displayed will be spread over multiple screens, as shown in [Figure 14-3](#) on page 86.

Use the scroll commands <PF7> to scroll up, and <PF8> to scroll down to view all the items on the screen.

```
tablesONLINE Administrator ----- Edit the Application Control Table -----
COMMAND ==>

                                New Row Mode                                Row Location : 1

APPLICATION ID      : ADMN
USER ID            :

START SCREEN NAME   : TB ADMIN
START SELECT SYMBOL :

TABLEBASE LIBRARY 1 : MAINLIB
TABLEBASE LIBRARY 2 : TBAPPLB
TABLEBASE LIBRARY 3 : TBSYSLB
TABLEBASE LIBRARY 4 : TBACTLB
TABLEBASE LIBRARY 5 : TESTLIB
TABLEBASE LIBRARY 6 :
TABLEBASE LIBRARY 7 :

DESC TABLE NAME    : TBOLDESC
MENU TABLE NAME    : TBOLMENU
PFKS TABLE NAME    : TBOLPFKS
CMDS TABLE NAME    : TBOLCMDS
MSGs TABLE NAME    : TBOLMSGs
HELP TABLE NAME    : TBOLHELP
LIBR TABLE NAME    : TBOLLIBR
TUTR TABLE NAME    : TBOLTUTR

SEARCH DICTIONARY   : N
UPDATE DICTIONARY   : N
UPDATE TABLE IND   : Y
ENABLE M2M COMMAND   : Y
ENABLE DIRECTORY LST :
DISPLAY LIBRARY IND  : Y
DISPLAY VIEWLIB IND  : Y
DISPLAY TBL NAME IND : Y

ACT EXITPGMS VERSION :
SECURITY EXIT NAME    :
SECURITY IN/OUT IND   :
MENU CMDL EXIT NAME   :
MENU CMDL EXIT IND    :
IDEN CMDL EXIT NAME   :
IDEN CMDL EXIT IND    :
TABL CMDL EXIT NAME   :
TABL CMDL EXIT IND    :
EDIT CMDL EXIT NAME   :
```

Figure 14-3: Edit Application Control table—Edit Row

Application Control table field descriptions

This section describes each of the fields in the **Application Control table** screen (see [Figure 14-3](#) on page 86), together with information necessary to complete each field. Default values are shown.

Application ID and user ID

The application ID is a four character name that identifies a particular application to tablesONLINE. The user ID is an eight character name that identifies a particular user to tablesONLINE. The application ID together with the user ID make up a user's signon ID. You may wish to create an application ID called ADMN that will be used by all users authorized to use the administrative functions, or one called DEVL that will be used by all users authorized to use the application development functions.

If within the ADMN application, two users require special authorities, the user ID field is used to further qualify the application ID. In this case, we would have three entries for the ADMN application; one with the user ID blank and two more with the appropriate signon user IDs.

One application ID may include entries for many user IDs, and each may be set up with a different starting position and authorities.

The start screen name and start select symbol

The start screen name field contains the name of the tablesONLINE/CICS menu screen that is to be displayed to users when they sign on using the previously specified application ID and user ID.

This start screen name may be one of the menus delivered with tablesONLINE/CICS or it may be a new, custom menu that has been created by your organization. The following is a list of the menus that are delivered with tablesONLINE/CICS.

- APPLDEVL
- TBADHELP
- DEFINE
- USERCNTL
- TB ADMIN
- UTILITY

It is possible to define a user's starting position more narrowly by indicating that the user is to begin with a particular option in the specified menu. To do this, you must insert the select symbol which identifies the desired menu option. This field is optional.

tableBASE Library List

The next seven fields contain the names of the tableBASE libraries that may be accessed by the identified user. These are the libraries that will be searched by tablesONLINE/CICS for required tables. Although it is not necessary to use all seven fields, the library list may not contain gaps. The TBDICLB library should not be included in this list. Two special flags, which are described below, are used to control the use of this library.

Application-driving tables

There are eight application-driving tables supplied with tablesONLINE/CICS. These tables contain the screen descriptions, menus, PF keys, commands, messages, help, library names and tutorial information. The application-driving tables may be customized to create your own applications using the instructions supplied in the *Building Applications* chapter of the *tablesONLINE/CICS User's Guide*.

You may use these eight fields to specify the default tables: TBOLDESC, TBOLMENU, TBOLPFKS, TBOLCMDS, TBOLMSGs, TBOLHELP, TBOLLIBR and TBOLTUTR, or you may specify customized tables that have been developed for the specified application.

Search dictionary

This field indicates whether the application uses the data dictionary. The possible values for this field are F, Y, L or N. If you wish to keep all Views and related edit values tables in a data dictionary, this flag should be set to F, Y or L. Doing so ensures that any newly created Views are automatically placed in the data dictionary.

Entering Y or F indicates that the dictionary library TBDICLIB will be searched before any other tableBASE library. This will prevent the substitution and use of another View by a user when editing or browsing a table.

Entering L indicates that the dictionary library will be searched after any other tableBASE libraries. This will allow the substitution and use of another View by a user when editing or browsing a table.

If you do not wish to use the data dictionary facility, this field should be set to N.

Update dictionary

This field indicates whether or not the user will be permitted to update the tables in the data dictionary library. This indicator may be set to either Y or N.

Update table indicator

This indicator may be set to N to prevent a user from changing table data, thereby creating a read-only version of any application. When this is done, access to all tablesONLINE/CICS utilities is denied. This provides a fully protected read-only version of an application without requiring any alteration to the application menus. If this indicator is set to Y, the default, users can make changes to the table data.

Enable M2M command

This field controls whether the user is permitted to turn the M2M (many-to-many) facility on and off. When turned on, this facility allows a user to refer to a table/View combination by a single Table Object name. This indicator may be set to either Y or N. For additional information on the M2M facility and its use, see the *tablesONLINE/CICS User's Guide*.

Enable directory list

This field controls whether F1 produces a list of library tables, or just the standard help panel display. This indicator may be set to either Y or N. If this indicator is set to N, the help panel will be displayed; if set to Y, all tables in the library will be listed.

Display library indicator

This field controls whether the library name is presented to the user on the table identification screens for Data Tables. This indicator may be set to either Y or N. If this indicator is set to N the user will not be able to search other libraries for a table by changing the library name.

Display view library indicator

This field controls whether the library name is presented to the user on the table identification screens for Views. This indicator may be set to either Y or N. If this indicator is set to N the user will not be able to search other libraries for a View by changing the library name.

If you are using a corporate data dictionary (TBDICLIB) to store all Views, this indicator should be set to N since all Views will be found in the dictionary.

Display table name indicator

This field controls whether the table name and generation number are presented to the user on table identification screens. This indicator may be set to either Y or N.

ACT exit programs version

Indicates the version of the EXITPARM copybook the exit program was compiled with. This field must be set to 4, indicating a version of tableBASE prior to Version 5, or 5 indicating Version 5, or 6 or blank indicating the current version of the EXITPARM for Release 5.1 or later.

Security exit name

This field indicates the name of the customer-written exit program that is to be called as indicated by the security indicators (see below). This security exit program will be invoked by tablesONLINE/CICS after its own security procedure has been completed.

Security in/out indicator

This field controls whether a security exit program is invoked before, after, or before and after, executing the application. In this context, before implies a point prior to application-specific initialization, while after implies a point after application clean-up, prior to signing off tablesONLINE. Set the indicator to a value listed in [Table 14-1](#) on page 89.

Indicator setting	Description
Blank	No security exit program is invoked.

Indicator setting	Description
I	Security exit program is invoked before application.
O	Security exit program is invoked after application.
Y	Security exit program is invoked before and after application.

Table 14-1: Security settings

Command line exit names and indicators

The command line exit indicators indicate when the respective command line exit program is called. Set the menu and table exits indicator to a value listed in [Table 14-2](#) on page 90.

Indicator setting	Description
Blank	No command line exit program is invoked.
B	Command line exit program is invoked after parsing the command line but before parsing menu selection or table edit line commands.
A	Command line exit program is invoked after parsing a menu selection or table edit line command.
Y	Command line exit program is invoked after parsing the command line and after parsing a menu selection or table edit line command.

Table 14-2: Command line exit names

For the identify (IDEN) and edit (EDIT) exits, the valid values are blank, B and Y.

The command line exit names indicate the name of the exit program to be called from various screens in tablesONLINE/CICS. Exit programs may be invoked from the **Menu** screen, the **Identify Table/View** screen, the **Edit Table** screen and the **Edit Row** screen. For detailed information on the creation of exit programs see the chapter "tablesONLINE/CICS exit programming" in the *tableBASE Programming Guide*.

Once you have made all the necessary modifications to these fields, press <End> to add the new entry to the Application Control table, then use <End> to exit to the **Administrator** menu.

Modifying an existing Application Control table entry

To modify an existing entry, select option 1, Edit Application Control table, from the **Administrator** menu (see [Figure 14-1](#) on page 83) and press <Enter>. The system displays the **Identify** screen.

The key fields for the ACT are the application ID and the user ID. To modify a specific item in the ACT, enter the application ID and, optionally, the user ID if one exists for that item, in the key fields and press <Enter>. This will display the row in the **Edit Row** screen where you may make the required modifications.

Editing the User/Application Relationship table

The User/Application Relationship table (TBOLUSAP) contains three fields, as shown in [Figure 14-4](#) on page 92:

- Signon user ID
- Signon application ID
- Target application ID

These fields map the user and application IDs listed in the Application Control table to actual applications. When a user signs on to the system using a user ID and application ID combination, tablesONLINE/CICS searches the User/Application Relationship table to determine which target application is to be started up for that user.

In most cases, the signon application ID and the target application ID will be the same. If the name of the target application changes and you do not want to force users to change their signon procedure, you may simply change the target application ID in the User/Application Relationship table for those users to the new target application. Users may continue to sign on using their regular signon ID but they will access the new target application.

To add an entry for a new user to the User/Application Relationship table, select option 8, Edit User/Application Relationship table, from the **Administrator** menu (see [Figure 14-1](#) on page 83) and press <Enter>. The system displays the **Identify** screen. The information on this screen that identifies the name and location of the User/Application Relationship table is protected and may not be modified. It is possible to modify the generation of the table to be edited in the event that you wish to return to a previous generation of the table. Two generations of this table are maintained.

Press <Enter> to display the **Edit Table** screen and the list of items currently in the User/ Application Relationship table, as shown in [Figure 14-4](#) on page 92.

tablesONLINE ----- Administrator ----- Edit Table -----
COMMAND ===>

Table Name : TBOLUSAPRow Location : 1

	SIGNON USER ID	SIGNON APPL ID	TARGET APPL ID
		ADMN	ADMN
		APP1	APP1
n	USER1	ADMN	ADMN
	USER1	TBOL	TBOL
	USER1	TREK	TREK
	USER1DCT	ADMN	ADMN
	USER2	ADMN	ADMN
	USER2	TBOL	TBOL
	USER2	TREK	TREK
	USER3	ADMN	ADMN
	USER3	TBOL	TBOL
	USER3DCT	ADMN	ADMN
	DEBUGS	ADMN	ADMN
	USER4	ADMN	ADMN
	USER4	TBOL	TBOL
++++	TESTER	ADMN	ADMN

Figure 14-4: Edit User/Application Relationship table

To create a new entry in this table, you must copy an existing row to use as a template. Use the tab or new-line keys to move to a row that is similar to the new row you wish to create. Use the New line command by inserting **n** to the left of the row and selecting <Enter>. The selected item now appears in the **Edit Row** screen for editing, as shown in [Figure 14-5](#) on page 92.

tablesONLINE ----- Administrator ----- Edit Row -----
COMMAND ===>

Table Name : TBOLUSAPNew Row ModeRow Location : 3

SIGNON USER ID	: USER1
SIGNON APPL ID	: ADMN
TARGET APPL ID	: ADMN

Figure 14-5: Edit User/Application Relationship table—Edit Row

[Table 14-3](#) on page 93 is a description of each of the fields in this screen together with information necessary to complete each field.

Field	Description
Signon user ID	This field contains the user ID for the new user that you are defining. This ID may be up to eight characters in length.
Signon application ID	This field contains the application ID for the application that the new user is to access. This ID may be up to four characters in length.
Target application ID	This field contains the actual application ID for the target application that is to be accessed by the user who specifies the signon user and application IDs. This ID may be up to four characters in length.

Table 14-3: Field explanation—user/applications

In addition to the entries in this table, there must be matching entries in the Application Control table for the application ID.

It is possible to create an entry in this table leaving the signon user ID blank and inserting only a signon application ID and target application ID. If such an entry exists, anyone signing on to tablesONLINE/CICS with the appropriate application ID, and either no user ID or one that is unknown to the system, will access the application identified by the target application ID.

It is also possible to create an entry leaving both the signon user ID and the signon application ID blank. Anyone then signing on to tablesONLINE/CICS and specifying no user ID or application ID or specifying IDs that are unknown to the system will access the application identified by the target application ID.

[Table 14-4](#) on page 93 list the possible entries for the TBOLUSAP table. Entries that contain bbbb indicate that the field is blank.

Signon user ID	Signon application ID	Target application ID
bbbb	bbbb	PLTB
AD01	bbbb	ADMN
AD01	APPL	APPL
AD01	FINE	FINE
AD01	PLTE	PLTE

Table 14-4: All possible entries—User/Application Relationship table

Based on the information in [Table 14-4](#) on page 93, the following situations will occur:

- Any user may access application PLTB.

- User AD01 will access application ADMN unless the user enters a signon application ID. The user may enter APPL, FINE and PLTE as signon application IDs to access the respective applications.

Cross-Authorization table

The Cross-Authorization table (TBOLXAUT) is used to allow one user to sign on to the system as another user. This is intended to allow an administrator to sign on as an application developer or end user and see the system as it would be presented to that user. This facility may also be used by an application developer during development to see the system as it will be presented to the end user. It is not necessary to make a new entry in this table when creating new users.

In order to sign on to tablesONLINE/CICS as another user, an entry must exist in the cross authorization table indicating that you are authorized to do so. The signon user ID in the table is the ID used to sign on to CICS. Once signed on to CICS, you may provide the alternate ID to sign on to the tablesONLINE/CICS system as the desired user. tablesONLINE/CICS will take the alternate ID you have provided and check the Cross-Authorization table to ensure that you are authorized to sign on to the system using that ID.

If your CICS system does not use signon security, the table TBOLXAUT may be set up to translate whatever EXEC CICS ASSIGN USERID returns into an acceptable target user ID as entered when signing on to tablesONLINE. For example, if CICS returns the value 'CICSUSER', an entry of **CICSUSER** in the signon user ID field and **TBUS0001** in the target user ID field will allow user TBUS0001 to sign on to tablesONLINE.

When a user signs on to a user ID/application ID that is in use, the session is terminated at the first terminal and continued at the second terminal. If the user ID/application ID is not in use, then a new session is started.

To add an entry to the Cross-Authorization table, select option 7, Edit X-Authorisation table, from the **Administrator** menu (see [Figure 14-1](#) on page 83) and press <Enter>. The system displays the **Identify** screen.

The information on this screen that identifies the name and location of the Cross-Authorization table is protected and may not be modified. It is possible to modify the generation of the table to be edited in the event that you wish to return to a previous generation of the table. Two generations of this table are maintained. press <Enter> to display the **Edit Table** screen and the list of items currently in the Cross-Authorization table, as shown in [Figure 14-6](#) on page 94.

tablesONLINE ----- Administrator ----- Edit Table -----	
COMMAND ==>	
Table Name : TBOLXAUT	Row Location : 1
SIGNON	TARGET
USER	USER
__ID__	__ID__
USER1	USER3
USER3	USER1

Figure 14-6: Edit Cross-Authorization table

To create a new entry in this table, you may copy an existing row to use as a template. Use the tab or new-line keys to move to a row that is similar to the new row you wish to create. Use the line command New by inserting N to the left of the row and pressing <Enter>. The selected item now appears in the **Edit Row** screen for editing, as shown in [Figure 14-7](#) on page 95.

```
tablesONLINE ----- Administrator ----- Edit Row -----
COMMAND ==>

Table Name : TBOLXAUT          New Row Mode          Row Location : 3

SIGNON USER ID      :
TARGET USER ID      :
```

Figure 14-7: Edit Cross-Authorization table—edit row

[Table 14-5](#) on page 95 is a description of each of the fields in this screen together with information necessary to complete each field.

Field	Description
Signon user ID	This field contains the CICS signon ID of the user, usually the system administrator or application developer, who will be authorized to sign on to tablesONLINE/CICS as the target user.
Target user ID	This field contains the name of the user who the administrator or application developer will be signed on to tablesONLINE/CICS.

Table 14-5: Field explanation—Cross-Authorization table

tablesONLINE/CICS and VTS

With Release 6 tablesONLINE/CICS takes advantage of the read/write capabilities of VTS to remove tableBASE transaction affinities associated with using CICS MRO within a single LPAR. When first installed tablesONLINE/CICS will use the local TSR for these tables. In order to take advantage of VTS, the tables that control the operation of the tablesONLINE/CICS must be in a VTS-TSR named DK1S. (See [Use of temp tables and TS queues](#) on page 122.) The following modifications are required.

1. Rename the TBSOSVTS table to TBSYSVTS.

During initialization tablesONLINE/CICS 6.0 will look for a table named TBSYSVTS. The presence of this table in the DK1S VTS-TSR indicates that tablesONLINE/CICS is being used with VTS.

The TBSOSVTS table is installed with tablesONLINE/CICS and can be found in the TBSYSLB library. The renaming of this table to TBSYSVTS will enforce the sharing of

the tablesONLINE control tables (TBSYSLB, TBAPPLB, TBACTLB, and TBDICLB) by opening them into the DK1S VTS-TSR rather than the local TSR.

2. Display the option to Edit the TBSYSVTS table in the **Administrator** menu.

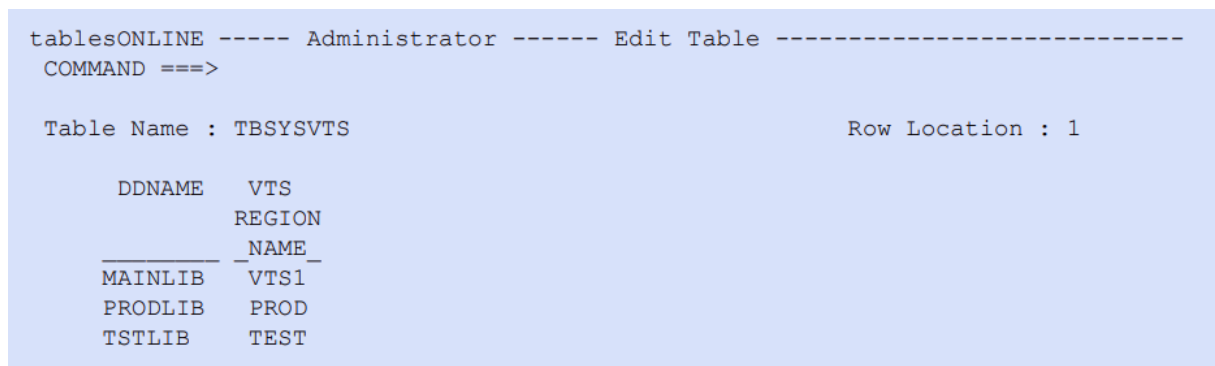
The menu option to edit the TBSYSVTS table is hidden from display. Follow the instructions provided in the chapter "Building Applications" of the *tablesONLINE/CICS User's Guide*, for editing the TBOLMENU or XXXXMENU table. The entries that make up this menu are identified in the MENU table by the TB ADMIN Screen name. The Select Symbol—VTS row is displayed; change the Display Indicator to Y.

3. Modify TBSYSVTS.

The TBSYSVTS table must be modified to indicate into which VTS-TSRs a library's tables are to be opened. The modification of this table provides a means of using other VTS-TSRs in addition to the default DK1S.

The delivered defaults are set so that PRODLIB tables are opened into a VTS-TSR names PROD, TESTLIB tables are opened into a VTS-TSR name TEST, and the MAINLIB tables are opened into a VTS-TSR named VTS1, as shown in [Figure 14-8](#) on page 96. Obviously, you will need to modify this table to reflect your installation.

Note: If this table does not indicate which VTS-TSR a library's tables should be opened into, then the tables will be opened in the DK1S VTS-TSR.



```

tablesONLINE ----- Administrator ----- Edit Table -----
COMMAND ===>

Table Name : TBSYSVTS                                Row Location : 1

      DDNAME      VTS
              REGION
              NAME
-----
MAINLIB      VTS1
PRODLIB      PROD
TSTLIB       TEST
  
```

Figure 14-8: Edit the TBSYSVTS table

4. Ensure the VTS-TSR DK1S is initialized.

During tablesONLINE/CICS initialization, if the TBSYSVTS table is found and the DK1S VTS-TSR is not up, you will receive an error message indicating that the DK1S VTS-TSR needs to be started before launching tablesONLINE/CICS. For more information on

initializing a VTS-TSR, please see the *tableBASE Installation Guide* and the sample JCL shown in [Figure 14-9](#) on page 97.

```
//VTS DK1S1 JOB 0, 'TBOL TEST VTS',MSGCLASS=Q,CLASS=V,NOTIFY=&SYSUID
//*
//* LAUNCH A 20 M VTS-TSR
//*
//RUN      EXEC PGM=DK1VAGNT,TIME=1440
//STEPLIB DD DISP=SHR,DSN=*YOUR.PREFIX*.TBASE.LOAD
//TBOPT    DD *
LISTOPTIONS=Y
VTSNAME=DK1S
TSRSIZE=20M
//*
```

Figure 14-9: Sample JCL code for initializing the DK1S VTS-TSR

Although other VTS-TSRs may be used with tablesONLINE CICS (as indicated in the TBSYSVTS table), DK1S must always be present during tablesONLINE/CICS initialization.

5. The region must be terminated and reinitiated.

This will cause the tablesONLINE control tables (TBSYSLB, TBAPPLB, TBACTLB, and TBDICLB) to be loaded into DK1S where they can then be shared among regions in the LPAR. The TBSYSVTS table will indicate which VTS-TSR other tables should be opened in, as shown in [Figure 14-8](#) on page 96.

Creating new applications

tablesONLINE/CICS allows you to create new table-driven applications simply by copying existing application-driving tables and customizing them. The seven application-driving tables control:

- menus
- PF key definitions
- command aliases
- help text
- description text
- message text
- library list for the application

In order to begin creating a new application, you must copy an existing set of application-driving tables. Select option 3, Copy Application, from the **Administrator** menu (see [Figure 14-1](#) on page 83).

In the delivered version of tablesONLINE/CICS the Copy Application option is provided as an administrative function. Depending on your installation, you may wish to grant the authority to perform this function to your application developers. For the procedure to do this, see the section on editing menu tables in the chapter "Building applications" of the *tablesONLINE/CICS User's Guide*.

Copying applications

To create a new application by copying the application-driving tables, select option 3, Copy Application, from the **Administrator** menu (see [Figure 14-1](#) on page 83). The system displays the **Copy Application Control Tables** screen, as shown in [Figure 14-10](#) on page 98.

```
tablesONLINE ----- Administrator ----- Copy Application Control Tables -----
COMMAND ==>
The application identifier is the prefix for the application tables.

                                Copy all the Applications Control Tables
                                ***** Press the 'EXECUTE' key *****

                                TO INVOKE

APPLICATION NAME      : TBOL

NEW APPLICATION NAME :
```

Figure 14-10: Copy Application Control tables

There are only two fields to be completed on this screen.

- application name, is the name of the application that is being copied
- new application name that is assigned here is the name that will be used as the prefix for the new application's driving tables

When you have completed both of these fields, press <Execute>. This copies the seven application-driving tables to create driving tables for the new application. For example, if XXXX and YYYY are entered, tablesONLINE/CICS will copy the tables listed in [Table 14-6](#) on page 98.

From	To
XXXXMENU	YYYYMENU
XXXXPFKS	YYYYPFKS
XXXXCMDS	YYYYCMDS
XXXXDESC	YYYYDESC
XXXXMSGs	YYYYMSGs
XXXXHELP	YYYYHELP

From	To
XXXXLIBR	YYYYLIBR

Table 14-6: Copy application-driving tables

Once the new tables have been created, the next step is to edit the copies for the new application.

The copy application function will not copy application-driving tables from one library to another. If you wish to store the application-driving tables for your new application to a different Application-Driving Tables library, you must use the **Copy** utility in the **Utilities** menu to do this.

For the first new application you create, you must use the application-driving tables for tablesONLINE/CICS as a prototype. To do so, insert an application name of TBOL. If you leave the application name field blank, it will default to TBOL. Once you have created a custom application you may wish to use that application as a starting point for future applications or you can continue to use TBOL, making all necessary changes once the tables have been copied.

Developing applications

Once the application-driving tables have been copied, they may be modified as required by an application developer. Select option D, Develop Application, from the **Administrator** menu (see [Figure 14-1](#) on page 83). The system will display the **Application Developer's** menu as shown in [Figure 14-11](#) on page 99.

```
tablesONLINE ----- Administrator ----- --- Application Developer's menu -----
COMMAND ==>
```

To select, enter number/symbol on command line:

A	EDIT TABLE	- Add/Change/Delete Rows in a Table
B	BROWSE TABLE	- Display Contents of a Table
C	TBDRIVC	- Execute TBLBASE Commands
D	DEFINE TABLE	- Define Table, View and Data Descriptions
U	UTILITIES	- Copy/Rename/Delete a Table
1	EDIT MENU TABLE	- Add/Change/Delete Application menu Items
2	EDIT PFKS TABLE	- Add/Change/Delete Application PF Keys
3	EDIT CMDS TABLE	- Add/Change/Delete Application Alias Commands
4	EDIT HELP TABLE	- Add/Change/Delete Application Help Items
5	EDIT MSGS TABLE	- Add/Change/Delete Application Messages
6	EDIT DESC TABLE	- Add/Change/Delete Application Screen Descriptions
7	EDIT LIBR TABLE	- Add/Change/Delete Application Library Names

Enter HELP at any stage for help within tablesONLINE.

Enter PF for program function key assignments.

Enter X to suspend tablesONLINE and return to CICS.

Figure 14-11: Application Developer's menu

Removing a User-developed application

The following procedure will remove a tablesONLINE/CICS User-developed application:

1. Delete the following application tables from TBAPPLB.

- xxxxMENU
- xxxxDESC
- xxxxMSGs
- xxxxHELP
- xxxxCMDS
- xxxxPFKS
- xxxxLIBR

- Note:**
- Delete tables where xxxx is a 4 byte table prefix, used when the application was copied through TBOL admin.
 - NEVER delete tables that have a prefix of TBOL. We assume that tables with prefix TBOL were never modified when tablesONLINE/CICS was delivered, and that the tables were copied over instead.
 - To delete a particular application you can use TBDR to delete each table by:
 - doing an ML to TBAPPLB.
 - doing a CL of the specified table to close it from the TSR
 - doing an XT to eliminate the table from the library.
 - Before deleting any of the tables, please ensure that no other application ID, in the TBOLACT table (in TBACTLB), except the application you are trying to delete, uses any of the same table names in any of its rows.
 - If the table name is used in any other application ID, then DO NOT delete the table.
2. Delete the application rows from TBOLACT.
Delete the respective application rows from the TBOLACT table in TBACTLB.
3. Delete the application rows from TBOLM2M.
Delete the respective application rows from the TBOLM2M table in TBDICLB.

Customizing your environment

It is possible to customize your tablesONLINE/CICS environment to suit your own organization.

The User Profile table contains an entry for each user that stores information about the user's environment. When setting up your initial tablesONLINE/CICS environment, you may create a customized default entry in this table, which will be used as a starting point by each new user.

A facility is provided that allows you to customize both the Help and Tutorial tables for your system. Changes made to these tables at the administrative level affect the help and tutorial information for your entire tablesONLINE/CICS system. It is also possible for application developers to create application specific help, as described in the *tablesONLINE/CICS User's Guide*.

The tablesONLINE/CICS software includes a Constants table. This table contains the system names for commands, system wildcard, applications and libraries. The names assigned to all of these items may be customized for your organization.

You may edit the MRO table, to identify applications that exist in other CICS sessions.

User Profile

tablesONLINE/CICS maintains a record of a user's environment so that it may be used from one session to the next. When a user exits tablesONLINE, environmental information such as the setting of program indicators and extended attributes is stored by tablesONLINE. The next time the user accesses the system their environment will be set up with the same environmental attributes.

The User Profile table maintains an entry for each user ID and application ID combination. These profiles are updated automatically when changes are made by the user to the environment.

When tablesONLINE/CICS is initially installed, there is one entry in the User Profile table. This entry is used as a default by tablesONLINE/CICS the first time each user accesses the system. At this point, there is no entry in the User Profile table for the user and tablesONLINE/CICS will use the default entry to set up the user's environment. When a user exits tablesONLINE, a new entry is made for that user in the User Profile table, recording any changes to the environment that the user has made during the session.

Customizing the default entry

The default entry that exists in the User Profile table, following installation, should be reviewed and modified to meet your own requirements. This section describes each of the fields in the User Profile table and the possible values for each.

To edit the default entry in the User Profile table, select option 4, Edit User Profiles, on the **Administrator** menu (see [Figure 14-1](#) on page 83). The system displays the **Edit** screen, as shown in [Figure 14-12](#) on page 101.

```
tablesONLINE ----- Administrator ----- Edit ----- Identify Table/Row
COMMAND ==>

Please indicate the table required by entering the parameters below.
Data Table Library      ==> APPLICATIONS.DRIVING.TABLES
View Library (If different) ==> SYSTEM.TABLES
Table Object==> TBOLPROF

Generation ==> 165

Enter row key for direct search or positioning on the edit table screen.

APPLICATION ID      ==>
USER ID             ==>
```

Figure 14-12: Edit screen—Identify Table/Row

The Data Table library, View library and Table Object are protected fields and may not be modified. If you wish to return to a previous copy of the User Profile table (two copies are

maintained by tablesONLINE), you can set the generation field to -1. If you know the application ID and user ID that you wish to modify you can insert them in the appropriate fields. In the case of the default entry, the application ID is PROF and the user ID is DEFAULT. Select <Enter> to display the **Edit Row** screen (see [Figure 14-13](#) on page 104). This screen contains all of the fields in this table with the delivered default settings.

These fields do not all fit on one screen of your terminal and will be displayed on several screens. Use <PF7> to scroll up, and <PF8> to scroll down. Each of the fields in this screen are described below. Once you have completed all of the necessary information on this screen, press <Execute> or leave the row to update the User Profile default entry.

Many of the fields in this table (see [Table 14-7](#) on page 105) are set automatically by tablesONLINE/CICS based on the actions of the user. Changing these fields is not required. In the following descriptions, the fields that are automatically set are indicated by an asterisk (*).

```

tablesONLINE ----- Administrator ----- Edit Row -----
COMMAND ==>

Table Name : TBOLPROF          Update Mode          Row Location : 1

APPLICATION ID      : PROF
USER ID            : DEFAULT
TABLE CONFIRM SAVE  : Y
ROW CONFIRM DELETE  : Y
TRANSFER TBLNAME IND : Y
SUPPRESS INFOMSG IND : N
TABLE COPY REPL IND : N
SUPPRESS MSG ID IND : N
FREEZE KEYS COUNT   : 3
EDIT /BROWSE DBTYPE : TABLE
EDIT /BROWSE LIBRARY : MAINLIB
EDIT /BROWSE OBJECT :
EDIT /BROWSE TABLE :
EDIT /BRWS VIEW LIB : MAINLIB
EDIT /BROWSE VIEW   : TBOLMENU
UTILITY TO LIBRARY   :
UTILITY TO TABLE    :
UTILITY TO VIEW LIB  :
UTILITY TO VIEW      :
RESTRUCTURE TABLE   :
RSTRCT ALL GENS IND  : Y
JOB NAME             :
ACCOUNTING INFO       :
NOTIFY               :
COPYLIB DATASET NAME :
MEMBER NAME          :
FIELD NAME PREFIX     :
INCLUDE COMMAND AREA :
C STRUCTURE NAME      :
JCL TO MAKE COPYBOOK : TBOLJCL1
JCL TO PRINT TABLE   : TBOLJCL2
JCL TO PRINT VIEW     : TBOLJCL3
JCL TO MAKE C STRUCT  : TBOLJCL4
COMMAND ==> EXTATT    : F100F200
COMMAND AREA EXTATT   : 0000F400
MENU/PF DASH EXTATT   : F2000000
PFXX SYMBOL EXTATT    : 00000000
PF COMMAND EXTATT     : F2000000
LIBRARY NAME EXTATT   : 00000000
TABLE NAME EXTATT     : 00000000
GENERATION NO EXTATT  : 00000000
IDEN ARROW EXTATT     : 00000000
DUPL KEYS EXTATT      : 00000000
TITLE EXTATT          : 00000000
LINE CMD EXTATT       : 00000000
LINE CMD-HL EXTATT    : F600F200
ROW DATA EXTATT      : F5000000
ROW DATA-HL EXTATT   : F6000000

```

Figure 14-13: Default entries for the User Profile table

Field name	Description
Table confirm save	The first field controls whether tablesONLINE/CICS asks for confirmation before saving a table. If the switch is set to Y, users will be asked for confirmation before the save operation is performed. If the switch is set to N, users will not be asked for confirmation before the table is saved.
Row confirm delete	The second field controls whether tablesONLINE/CICS asks for confirmation before deleting a row. If the switch is set to Y, users will be asked for confirmation before the delete operation is performed. If the switch is set to N, users will not be asked for confirmation before the row is deleted.
Transfer TBLNAME indicator	This field controls whether tablesONLINE/CICS changes its internal record of the current table name when a rename or copy operation is done. For example, if you are editing table AAA, move to the utilities to rename table AAA to BBB and then return to the editor, the initial table name which appears on the Select Table/Row screen will be BBB if this flag is Y and AAA if this flag is N.
Suppress INFOMSG indicator	This field is used to suppress the display, by tablesONLINE, of messages whose message type field in the message table is I, indicating an information message. If this field is set to Y, informational messages are suppressed. Messages whose message type is error, warning or abend are not affected by this indicator. These messages are displayed regardless of the setting of this indicator.
Table copy/replace indicator	This field indicates whether or not existing tables are replaced during a copy operation that specifies an existing table. If this indicator is set to Y, it is possible to copy a table to a location where a table with the same name exists and overwrite the existing table with the copied table. If this indicator is set to N, an operation that copies a table to a location where a table with the same name exists will result in an error and the table will not be overwritten.
Suppress message ID indicator	This field is used to suppress all message identification. If the indicator is set to Y, only the text of messages will appear. If the indicator is set to N, the text as well as an identifying code for the message will appear.
Freeze keys count*	This field indicates the number of fields that are currently frozen on the screen. Frozen fields will always be displayed on the screen when fields are being scrolled left and right.

Field name	Description
Edit/browse DBTYPE	This field indicates the type of database that is being accessed. This field will be set to TABLE.
Edit/browse library*	This field indicates the current tableBASE library for the Data Table that is being browsed or edited.
Edit/browse object*	This field indicates the current object that is selected.
Edit/browse table*	This field indicates the current table that is being browsed or edited.
Edit/browse View library*	This field indicates the current tableBASE library for the View table that is being browsed or edited.
Edit/browse View*	This field indicates the current View that is being browsed or edited.
Utility to library*	This field indicates the library used as the target library for a table when using a utility.
Utility to table*	This field indicates the table used as the target table for a utility.
Utility to View library*	This field indicates the library used as the target library for a View when using a utility.
Utility to View*	This field indicates the View used as the target View for a utility.
Restructure table*	This field will contain the name of a restructuring table if it has been saved for later use.
Restructure all generations indicator	If this field is set to Y, it indicates that all generations of a table have been restructured. If set to N, it indicates that only the most recent generation has been restructured.
Job name	This field contains the job name that is to appear on the job statement of the batch job to create a COBOL copybook for a View.
Accounting info	This field contains the accounting information that is to appear on the job statement of the batch job to create a COBOL copybook for a View.
Notify	This field contains the notify (user ID) that is to appear on the job statement of the batch job to create a COBOL copybook for a View.

Field name	Description
Copylib dataset name	This field contains the COPYLIB Dataset Name that contains the copybooks available to application developers for the compilation of application programs.
Member name	This field contains the name of the member in the COPYLIB that will contain the newly created copybook.
Field name prefix	This field contains a prefix that will be placed in front of all field names in the COBOL copybook, for example, WS-TBXX-. This prefix will also be placed in front of all field names in the command area, if included (see below).
Include Command Area	This field indicates whether or not the command area is to be included at the end of the COBOL copybook. It can be set to Y to include the command area or N to exclude it.
C structure name	The data in this field be used to name the C structure that is generated.
JCL to make copybook	This field indicates the name of the JCL used to make a copybook.
JCL to print table	This field indicates the name of the JCL used to print a table.
JCL to print View	This field indicates the name of the JCL used to print a view.
JCL to make C struct	This is the name of table that contains the skeleton JCL to be submitted to the Internal Reader when a C STRUCT is created. The table that is delivered for this purpose is TBOLJCL4. It is kept in TBAPPLB and must be modified to suit.

Field name	Description
Extended attributes	The balance of the fields in this table are extended attribute fields for use in controlling the appearance of color terminals. These fields should not be modified if any of the following conditions apply to your system: • Your terminals do not support these attributes. • Your CICS installation does not support them. • You are uncertain about the above conditions. If you do wish to make modifications to these attributes, experimentation is the best way to determine what function each of them performs and what effects are possible. Note: The tablesONLINE/CICS software does not perform error-checking on these fields, it passes the data to CICS as extended display attributes. You must know what attributes your terminals and the installed CICS support before attempting any experimentation. Attempting to use unsupported attributes may cause unpredictable and undesirable behavior from the terminal.

Table 14-7: Edit Row screen field explanation

Deleting entries in the User Profile table

It may become necessary for you to remove entries from the User Profile table for users who no longer require access to the system. To do this, select Edit User Profile from the **Administrator** menu (see [Figure 14-1](#) on page 83), find the entry for the user, and use the Delete line command to remove the row from the table.

Editing system help

Context-sensitive help is available throughout the tablesONLINE/CICS software. There are three different levels of help (in the order they are accessed):

1. multi-line error messages
2. short explanations to assist the user when entering data or using a menu
3. tutorial information

The error messages are contained in the tablesONLINE/CICSMSGs table. The System Help tables, which may be accessed and edited from the **Administrator** menu, contain the short explanations to assist the user. Finally, the tutorial information is contained in the Tutorial table, which may also be accessed and edited from the **Administrator** menu. This section describes the system help tables and how they are edited.

The entries that make up tablesONLINE/CICS help have been divided into a number of tables. Each table contains help related to a specific area of the product. For example, the ACT Help table

contains all of the help related to editing the ACT table and the Menu Help table contains all of the help related to editing the menu tables.

To begin editing a system help table, you must first open the **Help/Tutorial** menu. To open the **Help/Tutorial** menu, select option 5, Edit Help Tables, from the **Administrator** menu (see [Figure 14-1](#) on page 83). This will display the **Help/Tutorial** menu, as shown in [Figure 14-14](#) on page 109. Use <PF7> and <PF8> to scroll up and down respectively.

```
tablesONLINE ----- Administrator ----- --- Help/Tutorial Menu -----
COMMAND ==>

To select, enter number/symbol on command line:

H   EDIT HELP/TUTR HELP   - Add/Change/Delete Help/Tutorial Help Table Items
U   EDIT UTILITY HELP     - Add/Change/Delete Utility Help Table Items
VIEW EDIT DEFINITION HELP - Add/Change/Delete Table Defining Help Table Items
00  EDIT TBSYSUD HELP      - Add/Change/Delete Session ID Help Table Items
01  EDIT ACT HELP          - Add/Change/Delete ACT Help Table Items
02  EDIT MSGS HELP         - Add/Change/Delete Messages Help Table Items
03  EDIT DESC HELP         - Add/Change/Delete Description Help Table Items
04  EDIT CMDS HELP         - Add/Change/Delete Commands Alias Help Table Items
05  EDIT MENU HELP         - Add/Change/Delete Menu Help Table Items
06  EDIT PFKS HELP         - Add/Change/Delete PF Keys Help Table Items
07  EDIT LIBR HELP         - Add/Change/Delete Library Alias Help Table Items
08  EDIT XAUT HELP         - Add/Change/Delete Authorization Help Table Items

Enter HELP at any stage for help within tablesONLINE.
Enter PF for program function key assignments.
Enter X to suspend tablesONLINE and return to CICS.
```

Figure 14-14: Help/Tutorial menu

The **Help/Tutorial** menu contains an entry for each of the help tables. To edit a help table, insert the appropriate number or symbol on the command line and press <Enter>. This will display the **Identify** screen with key fields where you may insert a View name, field name and sequence number. If you wish to edit the information for a particular help entry, you can insert the appropriate row key and that row will be opened in the **Edit Row** screen. If these fields are left blank, the **Edit Table** screen is displayed. From this screen, you may select a help row to

edit using one of the line commands. This will display a screen similar to [Figure 14-15](#) on page 110.

```

tablesONLINE ----- Administrator ----- Edit the Application's Help/Tutorial --
COMMAND ==>

                                Key(s) Protected                Row Location : 9

VIEW NAME                      :
FIELD NAME                     : DISPLAY INDICATOR
SEQ                            : 2

INDIRECT LOOKUP IND           :
DISPL ATTR                    : 0
DISPL EXTATT                  : 00000000

HELP DATA                    : ....|....1....|....2....|....3....|....4....|....5
                                : Indicator to prevent the display of the selection
                                : ....|....6....|....7....|....
                                : on the menu display.

```

Figure 14-15: Application's Help/Tutorial sample row

[Table 14-8](#) on page 110 lists the fields of the **Application's Help/Tutorial** screen.

Field name	Description
VIEW NAME	Indicates the table that the help information applies to. If this field is left blank the help entry will apply to all the tables containing the field specified in the Field Name field.
FIELD NAME	Specifies the field within that table that the help applies to.
SEQ	The Sequence of items in the help table is controlled by a key composed of the View Name and Field Name fields together with the sequence number. The sequence number allows you to create multiple lines of help text for any table field
INDIRECT LOOKUP IND	This field allows you to create help message lines that are actually retrieved from another help table or another field name in the current help table or both.
DISPL ATTR DISPL EXTATT	Display attributes and display extended attributes control the visual presentation of the data on your screen. How these features are used will depend on your installation.

Field name	Description
HELP DATA	The Help data field contains the help message itself. This field may contain up to 79 characters of help data. Should you wish to create a help message that is longer than 79 characters, you may create multiple entries using the sequence number field. Help entries that have identical View name and field name will be displayed together in the order specified by the sequence numbers.

Table 14-8: Application's Help/Tutorial field explanations

If the help entry you are defining is to apply to the entire table rather than a specific field within that table, you should leave the field name blank. If the help entry you are defining is to apply to any table containing the specified field name you should leave the View name blank.

Creating an indirect help entry

To create an indirect help entry, first create a new row using the line commands. In the Application's Help/Tutorial menu (see [Figure 14-15](#) on page 110) insert an I in the Indirect lookup indicator field, clear the remaining non-key fields and press <Enter>. The system displays the **New Row Mode** screen, as shown in [Figure 14-16](#) on page 111. This contains several fields, Indirect table, Indirect key—table and Indirect key—field, that are used to define an indirect help entry.

```
tablesONLINE ----- Administrator ----- Edit the Application's Help/Tutorial --
COMMAND ==>
```

New Row Mode

Row Location : 39

```
VIEW NAME          :
FIELD NAME         : SCREEN-ID TBL-NAME
SEQ                : 5

INDIRECT LOOKUP IND : I
DISPL ATTR         :
DISPL EXTATT       : 00000000

INDIRECT TABLE    :
INDIRECT KEY - TABLE :
INDIRECT KEY - FIELD :
```

Figure 14-16: Application's Help/Tutorial—indirect help

In the Indirect table field, you must specify the name of the help table from which the indirect entry is to be retrieved. The Indirect key—table field allows you to specify the View name (that is, the first key field) for the help entry to be retrieved. The Indirect key—field field allows you to specify the field name (that is, the second key field) for the help entry to be retrieved. Indirect help lookup entries can be inserted anywhere in a help table but are acted on only once. It is not possible to nest or to create a recursive loop.

Editing system tutorials

The Tutorial table contains general program usage information. It includes a listing of all commands available at a given point in tablesONLINE/CICS and how those commands operate. To edit the Tutorial table, insert 6 on the command line (see [Figure 14-1](#) on page 83) and press <Enter>. This displays the **Identify** screen with key fields where you may insert a screen-ID, function, command/tutorial key and sequence number. If you wish to edit the information for a particular tutorial entry, you can insert the appropriate key and that row is displayed in the **Edit Row** screen, as shown in [Figure 14-17](#) on page 112. If these fields are left blank, the **Edit Table** screen is displayed. From this screen, you may select an item to edit using the line commands.

```
tablesONLINE ----- Administrator ----- Edit the Application's Help/Tutorial --
COMMAND ===>

                                Update Mode                                Row Location : 3

SCREEN-ID                       :
FUNCTION                        :
COMMAND /TUTR KEY               : CICS
SEQ                             : 4

DISPL ATTR                      : 8
DISPL EXTATT                    : 00000000

                                .....1.....2.....3.....4.....5
HELP DATA                      :          returned to this same screen and updatin
                                .....6.....7.....
                                : g can continue. It is
```

Figure 14-17: Application's Help/Tutorial—Edit Row screen

Rows in the Tutorial table are set up and edited in much the same way as the Help tables described above. The key fields for the Tutorial table, which identify the location that the tutorial information applies to, are different. The first key field for the tutorial table is the Screen-ID. This field identifies the screen that the tutorial information applies to. The next key field is the Function. This field allows tutorial information to be created for a screen that is dependent on the function being performed at the time. The third field identifies the individual commands that are available on the specified screen and to which the tutorial text applies. Finally, as with help, the sequence number allows multiline tutorial entries to be created.

[Table 14-9](#) on page 112 list some values for the screen ID and function, together with a description of where tutorial information will be displayed for each value.

Field	Entry	Tutorial applies to:
SCREEN-ID	blank	All menus
	MENU	Menu screen

Field	Entry	Tutorial applies to:
	IDENTIFY	Identify table/row screen
	EDIT-TBL	Edit table screen
	EDIT-ROW	Edit row screen
FUNCTION	blank	All functions and when SCREEN-ID is MENU
	EDIT	Editing Data Tables
	BROWSE	Browsing Data Tables
	BROWSEVIEW	Browsing Views
	EDITDSPL	Editing display order in Views
	EDITVIEW	Editing field definitions in Views

Table 14-9: Application's Help/Tutorial—ID/Function screen values

The remaining fields are those required to display a tutorial message. The Display Attributes and Display Extended Attributes values control the visual presentation of the data on your screen. How these features are used will depend on your installation.

The Help Data field contains the tutorial message itself. This field may contain up to 79 characters of data. Should you wish to create a message that is longer than 79 characters, create multiple entries using the Sequence Number field. Tutorial entries that have identical Screen-ID, Function and Command/Tutorial Key will be displayed together in the order specified by the sequence numbers.

System constants

The tablesONLINE/CICS software now allows you to modify the tablesONLINE/CICS system constants. The system constants are divided into three categories.

1. External names and DDNAMEs:
 - a. programs
 - b. CICS maps
 - c. DDNAMEs
2. User interface literals:
 - a. display literals
 - b. commands
3. Logical/physical literals required for the operations of tablesONLINE/CICS.

The constants in the External Names and DDNAMEs category should only be changed if the tablesONLINE/CICS program names conflict with currently-executing programs in your environment.

The user interface literals may be modified for multilingual support. If you wish to have your display literals in a language other than English, these literals may be modified for that purpose.

The third category of constants, the logical/physical literals, may be altered but will affect the operation of the tablesONLINE/CICS application-driving tables.

Warning: Because of the potential impact on the operation of the tablesONLINE software, DKL recommends that these literals only be altered in consultation with tableBASE customer support.

The tablesONLINE/CICS system constants are tied directly to a CICS region. If tablesONLINE/CICS is installed in more than one Application Owning Region in the MRO/ISC complex, each region may have a unique set of constants.

Editing the system constants

To edit the System Constants table, select option 9, Edit Constants, in the **Administrator** menu (see [Figure 14-1](#) on page 83) in the command line and press <Enter>. The system displays the screen shown in [Figure 14-18](#) on page 114.

```
tablesONLINE ----- Administrator ----- Edit ----- Identify Table/Row
COMMAND ==>

Please indicate the table required by entering the parameters below.
Data Table Library          ==> SYSTEM.TABLES
View Library (If different) ==> SYSTEM.TABLES
Table Object==> TBOLCONST
                                     Generation ==> 11

Enter row key for direct search or positioning on the edit table screen.

CNST CICS APPLID          ==>
```

Figure 14-18: Editing system constants

In this screen, the Data Table library, View library and Table Object are protected fields and may not be modified.

If you wish to edit an existing row in this table, insert the CNST CICS APPLID in the provided field and press <Enter>. This displays the row in the **Edit Row** screen, as shown in [Figure 14-19](#) on page 115.

```

tablesONLINE ----- Administrator ----- Edit Row -----
COMMAND ==>

Table Name : TBOLCNST      Key(s) Protected      Row Location : 1

      CNST CICS APPLID      : DEFAULT
                           External Program/DDName References .....
      TBAPPLIB              : TBAPPLB
      TBACTLIB              : TBACTLB
      TBDICTLB              : TBDICLB
      PROG TBDKAPPL         : TBDKAPPL
      PROG TBDKMENU         : TBDKMENU
      PROG TBDKIDEN         : TBDKIDEN
      PROG TBDKSITE         : TBDKSITE
      PROG TBDKEDIT         : TBDKEDIT
      PROG TBDKMSGR         : TBDKMSGR
      PROG TBDKHELP         : TBDKHELP
      PROG TBDKPFKS         : TBDKPFKS
      PROG TBDKMSTK         : TBDKMSTK
      PROG TBDKACTN         : TBDKACTN
      PROG TBDKLIST         : TBDKLIST
      PROG TBDKAPHK         : TBDKAPHK
      PROG SECURITY         : TBDKUSID
++++ SECURITY IO IND       : I

```

Figure 14-19: System constants—default row

If you want to see the list of rows, leave the CNST CICS APPLID field empty and press <Enter>. This displays all of the table rows in the **Edit Table** screen. From this screen, you may select a row to edit using the line commands. This will display a screen similar to [Figure 14-19](#) on page 115. Use <PF7> and <PF8> to scroll up and down respectively.

Note: Due to the size of the row, not all fields are displayed here. The balance of the fields in the default row of the System Constants table may be browsed online.

Editing MRO transaction IDs

The tableBASE software versions 5 and later provide you with the ability to install tablesONLINE/CICS in more than one Application Owning Region (AOR) in the MRO/ISC complex. The information necessary to allow this is contained in the TBOLMRO table. This table is only used if tablesONLINE/CICS is to be installed in more than one AOR. It is updated as part of the tablesONLINE/CICS installation process in this environment. Instructions for installing this facility are provided in the *tableBASE Installation Guide*.

Once installed, it is necessary to place entries in this table for each CICS region to be accessed, by editing the TBOLMRO table. To edit this table, from the **Administrator** menu (see [Figure](#)

[14-1](#) on page 83), select option M, Edit MRO Transaction ID, in the command line and press <Enter>. The system displays the screen shown in [Figure 14-20](#) on page 116.

```
tablesONLINE ----- Administrator ----- Edit ----- Identify Table/Row
COMMAND ==>

Please indicate the table required by entering the parameters below.
Data Table Library          ==> SYSTEM.TABLES
View Library (If different) ==> SYSTEM.TABLES
Table Object==> TBOLMRO
                                Generation ==> 6

Enter row key for direct search or positioning on the edit table screen.

MRO CICS APPLID             ==>
```

Figure 14-20: Edit—Edit MRO transaction IDs

In this screen, the Data Table library, View library and Table Object are protected fields and may not be modified. You may specify a generation of the table to edit, other than the current generation. Two generations of the MRO table are maintained by tablesONLINE/CICS.

If you want to edit an existing row in this table, insert the MRO CICS APPLID in the provided field and press <Enter>. This displays the row in the **Edit Row** screen, as shown in [Figure 14-21](#) on page 116.

```
tablesONLINE ----- Administrator ----- Edit Row -----
COMMAND ==>

Table Name : TBOLMRO          New Row Mode          Row Location : 2

MRO CICS APPLID      : DEFAULT
TBOL TRAN ID         : TBOL
MENU TRAN ID         : MENX
IDEN TRAN ID         : IDEN
SITE TRAN ID         : SITE
EDIT TRAN ID         : EDIT
TRIN TRAN ID         : TRIN
USR1 TRAN ID         :
USR2 TRAN ID         :
```

Figure 14-21: MRO Transaction IDs—default row

The first field is the key for this table. The fields on this screen are described in [Table 14-10](#) on page 117.

If you want to see the list of rows, leave the MRO CICS APPLID field empty and press <Enter>. This displays all of the table rows in the **Edit Table** screen. From this screen, you may select a row to edit using the line commands. This displays a screen similar to [Figure 14-21](#) on page 116.

The first field in this table contains either the CICS APPLID or DEFAULT. The APPLID is used as a search key to retrieve the corresponding transaction IDs for the CICS region. If the search fails, the DEFAULT entry is used. It is recommended that the DEFAULT entry not be deleted so that it may serve as a template for the creation of subsequent entries.

The next seven fields contain transaction IDs for various internal tablesONLINE/CICS programs. Unique names should be assigned to each of these programs for each application ID listed. [Table 14-10](#) on page 117 is an example.

Field	Transaction name		
MRO CICS APPLID	DEFAULT	AOL1	AOL2
TBOL TRAN ID	TBOL	TBO1	TB02
MENU TRAN ID	MENX	MEN1	MEN2
IDEN TRAN ID	IDEN	IDE1	IDE2
SITE TRAN ID	SITE	SIT1	SIT2
EDIT TRAN ID	EDIT	EDI1	EDI2
TRIN TRAN ID	TRIN	TRI1	TRI2
USR1 TRAN ID			
USR2 TRANID			

Table 14-10: MRO transaction IDs—sample table

The final two fields allow you to insert IDs for two user transactions which you may wish to identify.

Note: The transactions specified in the TBOLMRO table must also be defined to the Terminal Owning Region (TOR) of the MRO/ISC complex via CICS Resource Definition Online (RDO) as remote transactions.

Monitoring your system

There are a number of facilities provided within tablesONLINE/CICS that allow system monitoring. The first of these is the Session table. This table maintains a record of all tablesONLINE/CICS sessions that are active at any given time. The Session table may be used to delete user sessions that have become unstable. A second facility allows the system administrator to transfer to user applications in order to access those applications.

Session control

tablesONLINE/CICS includes a Session table that records active sessions. This table allows the system administrator to monitor the sessions that are active at any time as well as delete sessions that have become problematic.

When a user exits from tablesONLINE, the session is terminated. It is also possible for a user to suspend a session and subsequently return to that session at the point at which the user left it.

Session suspension and restoration

A session is terminated if the user ENDS out of the last active window. The user's entry in the Session table is then removed, as are temporary storage queues and related temporary tables. The next time the user signs on, he will begin at his defined starting point.

If a user enters the command CICS or its alias X, the session is suspended rather than terminated. The next time the user enters TBOL, the Session table will identify the user as already having a session in progress. The screen that is displayed will be the last screen used in the most recently created window and the session will be restored at this point.

The session restoration feature allows a user to:

- leave one terminal and continue the session at another terminal.
- leave tablesONLINE/CICS and continue with other work and to resume tablesONLINE/CICS at the point of departure.

Note: Leaving your tablesONLINE/CICS session on, while you are editing a table, is not recommended, as the table will be locked and other users will be prevented from updating the same table while your session is still on.

- continue a session after a terminal has temporarily been disconnected from CICS because of transmission or controller difficulties.

As well, the session restoration feature allows another user with authorization to pick up the session. This might be used by an administrator to pick up a user's session to assist with a user problem.

Terminating a session

It may become necessary to terminate a user's session if it becomes unstable, by deleting the row identifying that session from the Session table. For example, if a transaction abends in your session, because your session settings are saved by tablesONLINE/CICS, your session will abend again when you log into it. When the row identifying the session is deleted from the Session table, that user is returned to the application's first screen when he/she signs on.

The functions that may be performed on the Session table are limited. You may browse the table to see what sessions are running and delete any of those sessions. It is possible to open a row in the **Edit Row** screen (see [Figure 14-21](#) on page 116) to view the help available for each field but no changes may be made to any of the fields in this screen. New sessions cannot be created using this table, only the tablesONLINE/CICS signon process can create sessions on this table. There is no mechanism for changing the user ID for a running session.

To access the Sessions table, select option 2, Delete sessions, from the **Administrator** menu, as shown in [Figure 14-1](#) on page 83. The system displays the **Browse/Delete Active Users** screen, as shown in [Figure 14-22](#) on page 119.

```

tablesONLINE ----- Administrator ----- Browse/Delete Active Users -----
COMMAND ==>

Please indicate the table required by entering the parameters below.
Data Table Library          ==> SYSTEM.TABLES
View Library (If different) ==> SYSTEM.TABLES
Table Object==> TBSYSTMP
                                   Generation ==> 2

Enter row key for direct search or positioning on the edit table screen.

APPLICATION ID      ==>
USER ID            ==>

```

Figure 14-22: Browse/Delete Active Users screen

The Data Table library, View library, and Table Object on this screen are protected fields. If you wish to move to a particular row in the session table, enter the application ID of the user, either alone, or together with a user ID. This displays the **Edit Table** screen with the specified row displayed at the top of the screen. If no application ID or user ID is specified, the Session table is displayed with the first table row appearing at the top of the display.

The only operations that may be performed on this table are single row and block deletes. In the example in [Figure 14-23](#) on page 119, the session for USER3 is deleted.

```

tablesONLINE ----- Administrator ----- Browse/Delete Active Users -----
COMMAND ==>

                                   Row Location : 1

  APPL  USER  SES  WIN REMOTE  TABLE  LIBRARY
  ID    ID    SION DOW VTS   NAME      DATASET
  _____ _NO_ NO_ REGION _____ _NAME_
ADMN BILL0    1   1          TBSYSTMP TBDEV.V600.TBSYSLB
ADMN USER2    2   1
d  ADMN USER3    3   1
ADMN USER4    4   1

```

Figure 14-23: Example of how to delete a user session

As a safety precaution, error-checking code within tablesONLINE/CICS prevents the tableBASE system administrator from deleting his own session and will generate a message if this is attempted. In the example above, USER3 receives a message that the session has been terminated and he should speak to the tablesONLINE/CICS administrator. Following session termination, the user will be in native CICS.

If USER3 was in a menu or browsing a table at termination, all of the tables ONLINE/CICS resources are de-allocated cleanly. If they were editing a table, however, it may have been left in an open state.

To close a table that has been left open, use TBDRIVC. Set the eight character lock with the >L (Data to Lock) command so tableBASE thinks you are USER3. This requires a 19-character hex string on line one, as shown in [Figure 14-24](#) on page 120.

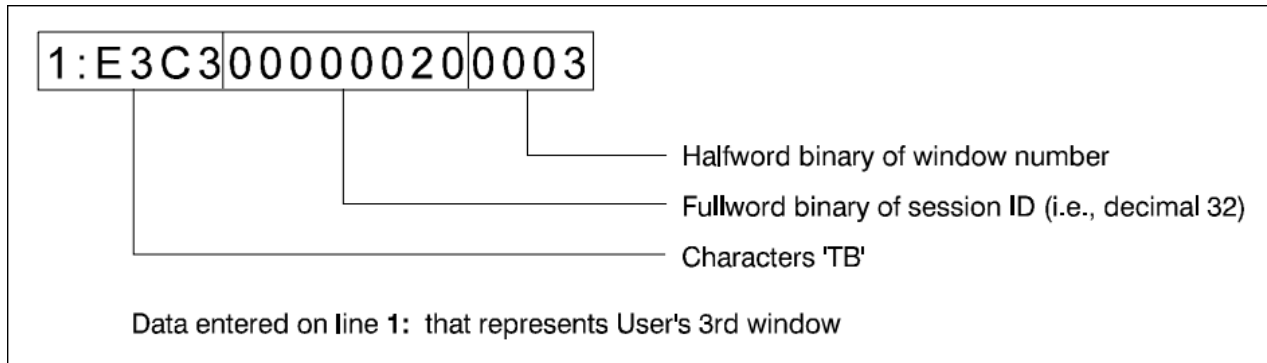


Figure 14-24: Entering the LOCK-LATCH of table that was left open

Alternatively, the lock may be set using the Master Password. For more information, see [Master Password Feature](#) on page 5. With the lock set, use the CL command on the appropriate table.

Terminating the administrator session

In some situations it may be necessary to delete the administrator session. This cannot be done from the Session table. Use the following procedure.

1. From the tableBASE CICS Driver, TBDRIVC, place low values in the lock by using the >L command with 16 zeros in the first line, as shown in [Figure 14-25](#) on page 120.

```

CMD: >1 TABLE: tbsystmp FOUND N I:  A:  ERR:  0 COUNT:  0 L:
ROW SZ:  0 KEY SZ:  0 FUNC ID:  0 FUNC PARM:  GEN:  1 ERRSUBCD:  0
1: 0000000000000000
2:
3:
HELP:  Enter a Command! (Use "HP" for general HELP information.)
.....1.....2.....3.....4.....5.....6.....7.....+

```

Figure 14-25: Place low values

2. Locate the entry for deletion in the Session table by using the Get Next and Get Previous commands, as shown in [Figure 14-26](#) on page 121. The rows displayed contains non-displayable information, but the user identification will be visible.

```

CMD: gn TABLE: tbsystmp FOUND N I:  A:  ERR:  0 COUNT:  0 L: .....
ROW SZ: 1292 KEY SZ:  0 FUNC ID:  0 FUNC PARM:  GEN:  1 ERRSUBCD:  0
1:
2:
3:
HELP:  Enter a Command!                (Use "HP" for general HELP information.)
      ....+....1....+....2....+....3....+....4....+....5....+....6....+....7....+

```

Figure 14-26: Locate entry for deletion

3. Delete the row, as shown in [Figure 14-27](#) on page 121.

```

CMD: dc TABLE: tbsystmp FOUND N I:  A:  ERR:  0 COUNT:  0 L: .....
ROW SZ: 1292 KEY SZ:  0 FUNC ID:  0 FUNC PARM:  GEN:  1 ERRSUBCD:  0
1:
2:
3:
HELP:  Enter a Command!                (Use "HP" for general HELP information.)
      ....+....1....+....2....+....3....+....4....+....5....+....6....+....7....+

```

Figure 14-27: Delete the row

It is not necessary to store the table, since the Session table does not reside on any library.

Transferring to user applications

A menu has been provided that allows an administrator to sign on to user applications, to assist with application debugging. You may want to grant the authority to use this function to supervisors in order to allow them to assist their application programmers with any problems that may arise. This function is particularly useful if you have individuals who are working at remote sites and require assistance with an application.

To access this function enter T on the command line of the **Administrator** menu (see [Figure 14-1](#) on page 83). This option was displayed in releases prior to Version 6. To make it visible on the Administrator menu, you have to change the DISPLAY INDICATOR to Y (for yes). See [tablesONLINE/CICS and VTS](#) on page 95 for details.

The system displays the **User Applications Select** menu, as shown in [Figure 14-28](#) on page 122.

```
tablesONLINE ----- Administrator ----- User Applications Select Menu -----
COMMAND ===>

To select, enter number/symbol on command line:

M    MARLON MENUS          - Transfer to Marlon's Application
PM   EDIT PAYROLL MENU    - Add/Change/Delete Payroll Application Menu Items
PT   PAYROLL SYSTEM       - Transfer to Payroll Application
TBM  EDIT TBOLMENU TABLE - Add/Change/Delete tablesONLINE System Menu Items

Enter HELP at any stage for help within tablesONLINE.
Enter PF for program function key assignments.
Enter X to suspend tablesONLINE and return to CICS.
```

Figure 14-28: User Applications transfer to screen

The entries that are displayed in this menu are examples of menu entries that transfer control to other sessions. Specific entries for your organization can be created by editing the TBOLMENU or XXXXMENU table as described in the chapter "Building Applications" of the *tablesONLINE/CICS User's Guide*. The entries that make up this menu are identified in the Menu table by the USERCNTL screen name.

Selecting an item from this screen brings up the application as the developer would see it, allowing the administrator or supervisor to examine it.

Use of temp tables and TS queues

If there is a need to clean up temporary tables and/or temporary storage queues (TSQs) in a CICS environment, take care not to prematurely delete those created by the tablesONLINE/CICS application. These items can be identified; do not prematurely delete the following:

- Internal tables: tables that contain the following prefixes:

```
DV<session-id><window-id>
DD<session-id><window-id>
SP<session-id><window-id>
EC<session-id><window-id>
ED<session-id><window-id>
MS<session-id><window-id>
```

MU<session-id><window-id>
LC<session-id><window-id>
RS<session-id><window-id>
SK<session-id><window-id>

- Temporary storage queues:

TA<session-id><window-id>
TB<session-id><window-id>

- Special purpose tables:

1. all tables with a TBSYS prefix.
2. tables used when performing auditing changes (i.e., for table named EXAMPLE, this would be tables named ExAMPLE or ExaMPLE)
3. tables with a AUD suffix (e.g., EXAMAUD)

If you plan to clean up your temporary tables as part of your process, you should confine all of your temporary tables into a VTS-TSR (see [tablesONLINE/CICS and VTS](#) on page 95).

Converting from Release 4.2.x

The internal structure of a View changed from Release 4.2.x to Release 5.0 and greater. tablesONLINE/CICS includes a special routine to convert Views to the new structure. In order to convert Views, tablesONLINE/CICS writes the new View structures to tableBASE libraries. In some sites, these libraries cannot be updated when tablesONLINE/CICS is executing. In this case, you must convert your Views using the program TBVIEWCV, which is supplied on the tableBASE installation tape.

In order to run TBVIEWCV you must know:

- your Master Password
- whether you use the exit program facility of tablesONLINE

Your Master Password is required so that any passwords associated with a particular View may be overridden. If you use the exit program facility, then you need to know whether your exit programs have been changed to take advantage of Release 5 or Release 6 capabilities.

This information – the Master Password and the exit program status – is passed to TBVIEWCV via the parm field of the EXEC statement. The parm field can contain two positional parameters. The first parameter, which is mandatory, defines the Master Password. The second parameter is optional. It specifies the version of the exit program that is being used with Views.

The Master Password parameter must be 8 bytes long. If your Master Password is less than 8 bytes long, pad the characters to the right with blanks. If you use the second parameter, it must begin at the ninth position of the parm field.

You will consider using the second parameter only if you are using the exit program facility of tablesONLINE. The second parameter, if specified, denotes the version of tableBASE for which your exit programs were created. It must contain either 4 (for releases prior to 5.0) or 5 (for

releases 5.0 or 6 [for releases 5.1 and later]). If your exit program version is blank and your exit program was written for earlier (pre-Release 5) versions of tableBASE, specify this parameter as 4. If you have modified your exit program to take advantage of Release 6 capabilities, specify this parameter as 6. [Figure 14-29](#) on page 124 shows a sample JCL.

```
//      Insert your job card here
//*****
//*
//*  PARS IN THE EXEC STATEMENT ARE:
//*      PASSWORD - THIS JOB REQUIRES THE MASTER PASSWORD TO
//*                  BE ABLE TO ACCESS THE VIEWS
//*      EXIT PROGRAM VERSION - DEPENDING ON WHICH VERSION YOUR
//*                          USER EXITS ARE AT
//*
//*      IF YOU CHANGE THE PARM IN YOUR EXIT, THE VERSION
//*      NUMBER ALSO HAS TO BE UPDATED. THIS WILL CHANGE
//*      YOUR EXIT VERSION GLOBALLY. THIS FEATURE IS TO BE
//*      USED WITH GREAT CAUTION. PLEASE PHONE DKL IF UNSURE.
//TBVIEWCV EXEC PGM=TBVIEWCV,PARM='MSTRPSWD4 '
//STEPLIB DD DISP=SHR,DSN=*YOUR.PREFIX*.TBASE.LOAD
//SYSOUT DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//TBLOG DD SYSOUT=*
//MAINLIB DD DISP=SHR,DSN=*your.data.library*
```

Figure 14-29: JCL for manually converting VIEWS

Note: In this case, the Master Password parameter is MSTRPSWD4, indicating a pre-Release 5 exit program. MSTRPSW 4, MSTR 4 are also valid; MSTRPSWD5, MSTRPSW 5 and MSTR 5 are valid for Release 5. MSTRPSWD6, MSTRPSW 6 and MSTR 6 are valid for Release 6.

Chapter

15

Best Practices

This chapter contains some information on optimization and organization that should be of benefit to tableBASE administrators.

Protecting tableBASE libraries

RACF (or equivalent) protection can be placed on tableBASE libraries. If an unauthorized user attempts to open a protected library, the open attempt will fail with appropriate messages sent to the job's JES log. An error code is returned to the user, which may result in an abend, depending on the abend status switch setting. In some environments, a RACF error will result in an abend of the task.

Concurrent multiple updates

When a table is identified for updating in a region with the use of the Open for Write command (OW), steps are taken to prevent other MVS tasks on the same or other MVS images from updating the same table. This is achieved by issuing an enqueue request. The enqueue minor name includes the Table Name and Table Library Dataset Name. The enqueue major name is DK1TCALL. The scope of the enqueue is SYSTEMS.

An installation's modifications to Global Resource Serialization (GRS) PARMLIB controls must not prevent the propagation of the major enqueue name throughout the complex. Failure to adhere to this can result in the corruption of the tableBASE libraries on DASD shared between several MVS images running tableBASE.

Since these requirements are beyond the scope of tableBASE software, you must ensure that these conditions are met if the same tableBASE libraries are to be shared between MVS tasks.

Read-Only libraries

tableBASE supports customers who wish to prevent updates to a tableBASE library. In batch and IMS, customers can use LABEL=(,,IN) on the DD statement. In CICS, customers can use the RDO statements READ(YES), ADD(NO), and/or UPDATE(NO). If an application attempts to update a read-only library, tableBASE will return error code 61 subcode 11 "Operation to write to a Library was terminated. Library read-only" or the calling task abends, depending on the abend switch setting.

Protecting tableBASE tables

tableBASE tables can be protected on the library (read and write passwords) and when loaded in-memory (LOCK-LATCH). It is important to note that these passwords offer only limited protection against unauthorized use of tableBASE tables. More formal security can be implemented via user exits and 3rd party measures such as Resource Access Control Facility (RACF), eTrust CA-ACF2 Security for z/OS and eTrust CA-Top Secret Security for z/OS.

In the event that a table password is lost or forgotten, please see [Master Password Feature](#) on page 5.

tablesONLINE exits provide security at the tables, fields and/or rows levels. For more information, please see the *tableBASE Programming Guide*.

Table read and write passwords

Passwords are used to open tables from the library. However once in memory, these tables can be accessed by any application with access to the memory. A read password protects a table from being opened for either read or write. A write password protects a table from being opened for write. For more information on passwords see the *tableBASE Programming Guide*.

TSR LOCK-LATCH

LOCK-LATCHES are used by tableBASE to protect a table in memory. A LOCK-LATCH is used when the table is opened for write and is needed for any future update or store access once it is set. LOCK-LATCHES are not checked during fetch access. For more information on LOCK-LATCHES see the *tableBASE Programming Guide* and the *tableBASE Concepts and Facilities Guide*.

Time out while waiting for a lock

To prevent processes from waiting indefinitely for a lock, use the TBOPT/TBOPTGEN LOCKTIMERC and LOCKTIMEWTO parameters. See [tableBASE Runtime Options](#) on page 135.

tablesONLINE temporary tables

tablesONLINE uses temporary tables that are critical to the operation of the application. If there is a need to clean up temporary tables and/or temporary storage queues (TSQs) in a CICS environment, take care not to prematurely delete those created by the tablesONLINE/CICS application. For more details, see [Use of temp tables and TS queues](#) on page 122.

If temporary tables are routinely deleted as part of your ongoing process, it is recommended that the tables that control the operation of tablesONLINE/CICS be confined to a VTS-TSR named DK1S.

Performance

Shared tables: Virtual Table Share (VTS)

Significant reduction of virtual storage utilization is possible when accessing tables from a VTS-TSR. If your installation uses multiple IMS regions or multiple CICS regions that each require access to the same tables, without VTS the tables are replicated in the TSR local to each region. Since each IMS or CICS region can access tables from a common VTS-TSR, the storage required for local TSRs can be greatly reduced or eliminated altogether. If a local TSR is not required after moving all the tables into a VTS-TSR, the local TSR size can be reduced to the absolute minimum by setting the TSRSIZE=0. For more information, see the TSRSIZE parameter description in Appendix A of the *tableBASE Installation Guide*.

If multiple users could be updating a table—whether online, in a VTS-TSR or other multitasking environment—be aware that a LOCK-LATCH is available to restrict updates to authorized users/applications. It is not required.

In a read/write VTS environment, it is important to ensure all tableBASE jobs operate at the same WLM priority level. This is to insure that a low priority job does not lock out a high priority job. This can occur when the low priority job is updating a tableBASE table and is interrupted by MVS.

Linking tableBASE

In a CICS environment, it is recommended that DK1TCALL be statically linked, and not defined to CICS. By statically linking DK1TCALL, the PPT is not searched for DK1TCALL and the call is very efficient. If it is called dynamically, it must be defined to CICS. In batch and other non-CICS environments, there may be performance impacts if tableBASE is not statically linked.

Using the TB-PARM Communication Area

DataKinetics recommends using a TB-PARM Communication Area on every call to DK1TCALL. If a TB-PARM is not used, performance can be significantly affected in some instances—for example, when DK1TCALL or DK1TROTB is in LPA or an authorized library. See [Batch/IMS Performance Degradation with Dynamic Calls to tableBASE](#) on page 128.

WLM priority level consideration

In a read/write VTS environment, it is important to ensure all tableBASE jobs operate at the same WLM priority level. This is to insure that a low priority job does not lock out a high priority job. This can occur when the low priority job is updating a tableBASE table and is interrupted by MVS.

Application performance with TBASEV or TBCALLV

To improve the performance of applications using the V5 TBASEV or TBCALLV interface, relink the application modules with the V6 TBASEV or TBCALLV (now aliases for DK1TCALL).

Batch/IMS Performance Degradation with Dynamic Calls to tableBASE

If tableBASE is called dynamically without a TBPARM and tableBASE modules are loaded in protected storage (loaded from LPA or an authorized library), there may be significant performance degradation. Performance degradation can also occur with a TBPARM if it is reinitialized before every call.

When tableBASE modules are loaded from an authorized library or LPA when a newly tested application is moved into production after being tested using tableBASE STEPLIBs, the application may experience performance degradation. The performance testing of an application should occur in the same environment as the production environment.

Batch, IMS and DB2 stored procedures accessing tableBASE may use excessive CPU time if tableBASE modules are in an authorized library or LPA and calls to tableBASE do not use a TBPARM parameter.

Single TBPARM per thread

A single TBPARM should be used per thread (for batch, this generally means just one; for IMS, one per transaction; for DB2 SP, one per SP). To be effective, it must be initialized at the first call to tableBASE and never re-initialized.

When tableBASE calls are issued with a TBPARM operand, our fast path "handle" is stored in the TBPARM area. If no TBPARM is used, tableBASE attempts to achieve similar performance by updating both DK1TCALL and DK1TROTB at initialization of tableBASE in the region. This will not be possible when these modules are in key 0 storage.

Re-entrant modules will be loaded into key 0 storage when they are loaded from LPA or from an authorized library.

If you are going to load tableBASE modules from LPA or an APF-authorized library AND you have applications which do not use a TB-PARM, then we recommend you relink DK1TCALL and DK1TROTB as serially reusable.

Relink control statements for DK1TROTB:

```
ORDER DK1T1153
INCLUDE loadlib(DK1TROTB)
MODE AMODE(31),RMODE(ANY)
SETOPT PARM(REUS=SERIAL)
ENTRY DK1T1153
NAME DK1TROTB(R)
/*
```

Relink control statements for DK1TCALL:

```
INCLUDE loadlib(DK1TCALL)
MODE AMODE(ANY),RMODE(24) <<--- USE IF RMODE(24) APPS EXIST
*MODE AMODE(31),RMODE(ANY) <<--- COULD USE IF NO MORE AMODE(24) APPS
SETOPT PARM(REUS=SERIAL)
ENTRY TBLBASE
ALIAS TBLBASE
ALIAS TBASE,TBBAFN01,TBCALL,MAINCALL
ALIAS TBCALLC,MAINROOT,TBASEC
ALIAS TBCALLI,TBROOTI
```

```
ALIAS    TBASEV, TBCALLV
ALIAS    TBROOT
NAME     DKITCALL (R)
/*
```

Table access optimization

Table access can be optimized by using a separate command area for each table. Instead of having to search a TSR directory each time to determine where the table resides in memory, tableBASE uses a reserved field in the command area (table handle) pointing to the position of the table in the TSR. By keeping a separate command area for each table accessed, this table handle is used to quickly locate a previously opened table.

Multitasking

tableBASE can be tailored to perform optimally in your environment by setting execution-time parameters and switches for both batch and online environments. In a batch environment only, the Multitasking parameter can be set to Y to allow multiple subtasks within a region to access tableBASE concurrently. The normal batch default is N. The default in a DB2 Stored Procedure Address Space is Y.

Note: For a full list of execution-time parameters see the *tableBASE Installation Guide*.

tableBASE library definition considerations

You can increase library blocksize to gain efficiencies in DASD utilization and reduce I/O access, in moving directory and table data between DASD and memory. You can also reduce DASD space requirements if the total data for each table in the library is significantly more than the current blocksize.

First, consider space utilization from the consideration of physical blocksize on 3390-style DASD (the current standard).

Library Blocksize

Each library can have an optimal blocksize selected at the time it is created.

Starting with tableBASE Version 6, library blocksizes are user-selectable. Previously, blocksize was fixed at 3120 bytes. Starting with Version 6, you can optimize the blocksize for your tables up to the size of a track (or 32,760 bytes, whichever is smaller). This increased flexibility allows you to decrease the time to load tables from the library to a TSR, since fewer I/Os are required. Consider putting all of your large tables together in one library and increasing the blocksize.

Changing the blocksize of a library that has been converted from Version 5 format prohibits it from being converted back to a Version 5 library. Libraries created starting with Version 6 can have any blocksize from 800 to 32760.

tableBASE library space usage

tableBASE attempts to use library space efficiently. If the BLKSIZE (RECLEN) is larger, it fits more directory entries into each block. But every table definition requires 1 full block for the definition.

The actual data for each table is stored in an integral number of full blocks, with the minimum being one. A very small table (e.g., 10 rows with a row length of 12 bytes) still requires a full block, which results in a very low utilization, even if a blocksize of 800 (the minimum allowed) is used. On the other hand, a very large table (e.g., 50000 rows with a row length of 200 bytes) will have over 99% utilization of its allocated space no matter what the blocksize.

BDAM/QSAM tableBASE libraries

For BDAM/QSAM libraries, the default block size of 3120 results in 15 blocks per track and 82.6% utilization of the space. Increasing block size to 3174 results in 84% utilization (still at 15 blocks per track). Increasing it to 27998 (2 blocks per track) results in 98.8% utilization. Reducing it to 820 (39 blocks per track) results in 56.4% utilization.

BDAM Fixed (F) libraries are supported; Fixed Block (FB) BDAM libraries are not supported.

VSAM tableBASE libraries

For VSAM libraries, the default blocksize of 3120 (RECSZ in IDCAMS DEFINE) results in a CISZ of 3584, 13 blocks per track and 71.6% utilization. Increasing the RECSZ to 3577 (the maximum in a 3584 CISZ) results in a 82.1% utilization. Increasing it to 4091 (the maximum in a 4096 CISZ) results in 86.6% utilization. Reducing it to 2041 (2048 CISZ) results in 75.6% utilization; reducing it to 1017 (1024 CISZ) yields 59.2% utilization.

VSAM tableBASE libraries and LSR pools

If a tableBASE VSAM library is allocated with DISP=OLD there are no restrictions on SHAREOPTIONS, buffering, strings and buffer pools.

However if it is allocated as DISP=SHR, tableBASE uses MVS Enqueues (SCOPE = SYSTEMS) and I/O techniques to ensure the library's integrity is maintained. Our standard recommendation has been to use SHAREOPTIONS(3,3) in all environments to prevent overlays of the tableBASE library directory and table data.

In CICS (and optionally batch), standard performance recommendations have been to add VSAM datasets to buffer pools (LSRPOOLS). However this would undermine tableBASE library integrity, so for libraries that it accesses, tableBASE will reset any LSRPOOLID value to zero, effectively making them NSR (Non-Shared Resources). Any software that changes buffer pool settings dynamically for VSAM datasets will create integrity problems for tableBASE libraries.

tableBASE cannot maintain library integrity with SHAREOPTIONS(3,3) and any VSAM buffering.

However, with CICS/TS 4.1, IBM is enforcing the restriction that CICS Transaction Isolation is not supported for NSR VSAM files. An AFDK abend will be issued when the library is opened. To accommodate this restriction (which does not apply to BDAM tableBASE libraries or libraries

allocated with DISP=OLD), tableBASE will ensure library integrity with SHAREOPTIONS(4,4) and LSRPOOLS for the CICS environment.

Based on IBM documentation, we recommend that the buffer pool used with SHAREOPTIONS(4,4) be set to a minimum size and not be shared with other users. The buffers will not benefit tableBASE and could interfere with other applications sharing the buffer pool. The library directory caching feature implemented with Version 6 should provide equivalent or better results than LSR pooling.

Enhanced Data Integrity option

tableBASE Release 7.0.3 is compatible with z/OS. z/OS runs only on 64-bit processors. But it fully supports 31-bit applications, which includes customers' CICS, IMS and DB2 applications (and tableBASE). IBM is also committed to continuing to support interaction between 31-bit and 64-bit applications in the future.

z/OS provides an option called "Enhanced Data Integrity" that can be enabled in SYS1.PARMLIB. This option is designed to protect physical sequential (DSORG=PS) datasets allocated with DISP=SHR from being concurrently opened for update by multiple users.

This can affect tableBASE users, since tableBASE libraries can be allocated with DSORG=PS (even though we only document BDAM (DSORG=DA) and VSAM). tableBASE internally protects tableBASE libraries when allocated with DISP=SHR, so "Enhanced Data Integrity" should not be enabled for tableBASE libraries.

The system programmer can bypass this option for specific datasets.

Ensure that tableBASE libraries allocated with DSORG=PS are listed in the exclude list in the IFGPSEDI member.

Library directory caching

The tableBASE library directory can now be cached in memory. This can dramatically reduce the time needed to open a large number of tables when a TSR is initially loaded.

Note: In an in-house test, load time for 40,000 tables from a tableBASE library into a local TSR was reduced from 11 hours to 7 minutes. This was an extreme stress test; your results may vary.

Caching for a library is controlled by coding OPTCD=C on the DD statement for a library. This feature is not supported in a CICS region. The initial loading of a TSR may benefit from the use of the library directory caching feature, and subsequent refreshes may also benefit if more than one table from the library is being refreshed. However, if updates are being done to the library while the loading or refreshing is being performed, these benefits will be lost. It is also important to note that the library directory caching feature is enabled only at region initialization.

Note: Caching has significant benefit when a library is accessed heavily by a region and other regions are not frequently updating the library. However, caching can have negative performance consequences when multiple regions are frequently updating the library directory.

This feature should provide equivalent or better results than LSR pooling. There is limited support for VSAM tableBASE libraries using LSR pools. For more information on using LSR pooling, refer to [VSAM tableBASE libraries and LSR pools](#) on page 130.

TSR Space Allocation

tableBASE allocates space for the TSR in 4K chunks and will only allocate space as needed as tables are opened or defined in the TSR. The minimum amount of space that can be allocated for a TSR is 28K and the maximum is 2G.

tableBASE initializes the first portion of the TSR to contain control information. For smaller TSR sizes of about 1M to 3M, about 20K of the space is used for control information. For larger TSR sizes, between 1 to 2 percent of the TSR space is used for control information.

When tableBASE initializes, it does a DSPSERV request for a private dataspace of the TSR size requested. For non-VTS regions, the dataspace is allocated as SCOPE=SINGLE. For VTS regions, the dataspace is allocated as SCOPE=ALL. The MVS IEFUSI exit at your site may limit the number and/or sizes of dataspaces allocated by a single job step.

If your region defines the TSR size as 1M, this results in the building of the MVS system control blocks to support a virtual storage of 1M. However, if you only open tables to fill up 500K of the TSR, tableBASE will only allocate about 520K of space for the TSR, 20K being the overhead for the control information.

If your region defines the TSR size as 100M, this results in the building of MVS system control blocks to support a virtual storage of 100M. However, if you only open or define tables that take up 20M of space, tableBASE will only allocate about 22M of space for the TSR, 2M being the overhead for the control information.

Taking into account the small overhead for the space required to hold control information on the TSR, if you anticipate a large variance in the amount of space that you will require, we recommend that you define your TSR size to be the upper limit of all space that you expect to use during the life of your region.

tableBASE modules that can reside in the Link Pack Area (LPA)

The following tableBASE modules are eligible and recommended to be in LPA for the Batch, DB2 and IMS interfaces:

DK1TCALL, DK1TNAME, DK1TNUCL, DK1TROT

If using systems or user exits, the following modules are recommended to be in LPA:

DK1TX071 and DK1TX072.

Note: When running CICS applications, tableBASE modules cannot reside in LPA.

Chapter

16

tableBASE Diagnostic Information

This chapter identifies and describes the tableBASE diagnostics tools for the tableBASE administrator.

tableBASE use of MVS Enqueues

In order to protect the integrity of tables stored in tableBASE libraries, tableBASE issues two MVS enqueues, a library enqueue and a table enqueue. The tableBASE software enqueues are issued with a scope of "SYSTEMS".

The enqueues have a major resource name of "DK1TCALL". For a library enqueue, the minor name consists of the letter "L" followed by the dataset name. For a table enqueue, the minor name consists of the letter "T" followed by the table name and the dataset name. These enqueues are independent of "SYSDSN" enqueues issued by the MVS allocation process, based on DISP=SHR or DISP=OLD.

Two changes for library enqueues were implemented with tableBASE Version 6:

- Since paged table support was dropped, the paged table enqueue and paged table generation enqueue were dropped.
- The table enqueue format was changed so that the table name precedes the dataset name instead of following it (this improves the usefulness of diagnostic display techniques which do not display the entire length of the minor name).

The tableBASE library enqueue (minor name "L...") protects the library directory and space allocation. It is obtained as a shared enqueue whenever the library has to be read. It is obtained as an exclusive enqueue when the library is being updated (for example, during a store (ST) of a table). The enqueue is held only during the operation of a single tableBASE command; it is always released before control is returned to the caller. The scope of "SYSTEMS" protects the library from other tableBASE accesses not only on the local system but also across any other systems in the GRS group.

The tableBASE table enqueue (minor name "T...") prevents the library copy of a table from being Opened for Write (OW) in more than one TSR. (With prior releases, this protection applied to a specific region, but starting with Version 6 tableBASE R/W VTS-TSRs, any region accessing the VTS-TSR obtains or releases the enqueue on behalf of the VTS-TSR). This enqueue is held across multiple tableBASE calls; it is released only when the table is no longer Open for Write in the TSR (typically released by a CClose or ReLease command for the table). The scope of "SYSTEMS" protects the table from any other attempts to update the library copy of this table

not only on the local system, but also across any other systems in the GRS group. Besides OW, commands CG, DG, RN, XT also obtain this enqueue.

tableBASE Library Diagnostics

In order to assist in the diagnosis of tableBASE Library integrity, two command update segments are written to Block 0 of a tableBASE library for each update made to the library. The first segment is written at the start of the update command and the second segment is written on successful completion of the update command. If a tableBASE command abends during the update of a library, the first segment will show the current command being processed and the second segment will contain information from the completion of the previous tableBASE library update.

Block 0 of the library can be viewed by running the job DK1TPRTM (TBPRNTM) which dumps library contents. DataKinetics Technical Support may request you to dump your library contents in order to assist with the diagnosis of the health of your library.

Appendix

A

tableBASE Runtime Options

This appendix lists the tableBASE run-time option parameters. Most occur in all interfaces. If you are running applications across multiple environments any changes to these options in one interface may need to be made in each installed interface. Options can be customized for your site at installation time, using the TBOPTGEN macro in DK1Txx34 modules. Applications can further override options on an individual basis, using the TBOPT dataset.

ALLOW_TB_LIB_IO—Allow tableBASE Library I/O

This parameter allows access to tableBASE libraries. When set to Y, access to tableBASE libraries is allowed; when set to N, access to tableBASE libraries is not allowed. By default, the parameter is set to Y, with the exception of the DB2 Stored Procedures interface, where it is set to N. While it is possible to set the parameter to Y for the DB2 Stored Procedures interface, it is not recommended as this could cause tableBASE to abend if more than 1 task in a DB2 WLM region access a tableBASE library at the same time.

This option can be specified for all tableBASE interfaces except the VTS interface.

CICSJRNL—CICS Journal File ID

This parameter identifies the CICS Journal File onto which the TABLE SPACE Report records are to be written by the system, if tableBASE strobe records are to be written. The delivered default value for the CICSJRNL parameter is 99.

This option can only be specified for the tableBASE/CICS interface.

This option applies to this region's users, regardless of which TSR they access.

DATERTNX—Date-Sensitive Processing Found Code

The DATERTNX parameter controls the value placed in the found code for a fetch by count (FC) operation on a table when date-sensitive processing is enabled.

With an FC command, it is possible that the row retrieved is not within the date range. When this occurs, the found code is normally set to N to reflect the logical condition that the retrieved row is not a match. For compatibility with previous releases, the found code returned under this condition can be made to be X. The default setting of DATERTNX=N provides this compatibility with previous releases.

If Date-Sensitive Processing is being introduced in a new application, or no FC commands are used in the application, it is strongly recommended that DATERTNX=Y be specified so the found code will never be set to X. The delivered default is DATERTNX=N for backwards compatibility.

This option applies to this region's users, regardless of which TSR they access.

This option can be specified for all tableBASE interfaces except the VTS interface.

FGDELIM—Fetch Generic Delimiter

This parameter identifies the character to denote the end of the high order part of the key used in the FG (Fetch Generic) command.

The delivered default value for FGDELIM is an asterisk (*), for example, FG,table-name,key*. We recommend that programmers use the key length override field in the tableBASE command area as this executes faster, is less error prone and allows an asterisk to be part of the key.

This option applies to this region's users, regardless of which TSR they access.

This option can be specified for all tableBASE interfaces except the VTS interface.

Note: If this option is changed for the tableBASE/CICS interface and you have the tablesONLINE/CICS product, then the value must be consistent with that specified for tablesONLINE/CICS. Please see the tablesONLINE/CICS section of the *tableBASE Installation Guide* for more details.

HASH_HI_DEN_LIM—High Density Limit for Hash Indexes

HASH_HI_DEN_LIM and HASH_LOW_DEN_LIM limit the density of the index for a hash table opened. These values are designed to prevent performance problems which can occur when inappropriately high values are used when defining hash tables (can result in a lot of key collisions in the table). Other problems occur if the difference between low and high density values is too small. A ratio of two thirds is now enforced: Low density may not be greater than two thirds of high density.

HASH_HI_DEN_LIM=nnn must be between 100 and 900 (10% and 90%); the default is 900. We recommend the value be lowered to 500.

This option applies to this region's users, regardless of which TSR they access.

This option can be specified for all tableBASE interfaces except the VTS interface.

HASH_LOW_DEN_LIM—Low Density Limit for Hash Indexes

See above ([HASH_HI_DEN_LIM—High Density Limit for Hash Indexes](#) on page 136).

HASH_LOW_DEN_LIM=nnn must be between 10 and 600; the default is 600. We recommend the value be lowered to between 200 and 300.

This option applies to this region's users, regardless of which TSR they access.

This option can be specified for all tableBASE interfaces except the VTS interface.

LDS—Linear Data Set Use

This specifies whether the TSR defined in this jobstep will be mapped to a LDS (Linear Data Set). The LDS must be pre-defined.

In a batch or VTS environment, this option can be set to Y to allow a mapping of the local TSR or VTS-TSR to a LDS. If set to N, there is no mapping to a LDS.

The delivered default is N.

This option applies to the TSR created by this region or job and all users accessing it.

Note: This parameter applies only if you have licensed the optional VTS Manager product.

LIBnn, ML—tableBASE Library List

Specify in sequence, the names of the libraries (DDNAMEs) in the tableBASE Library List (LIB-LIST). In TBOPTGEN, this is specified as a comma delimited list under ML. In the TBOPT file, the library names are specified as LIBnn=lib_name. Up to 10 LIBnn may be specified, as long as the numbers are contiguous. For example, LIB01=MAINLIB, LIB02=TESTLIB.

LIBnn numbers must be contiguous AND they must start with LIB01 or LIB02, otherwise tableBASE initialization will fail.

The delivered default is ML=MAINLIB which is equivalent to supplying a single parameter LIB01=MAINLIB. If no ML is set, tableBASE uses MAINLIB.

A VTS-TSR may also be included in the search list by specifying the name of the VTS-TSR in LIB-LIST instead of a library DDNAME. For example, if VTS: is the value specified for VTSPREFIX and the VTS name is DKL1, then VTS:DKL1 could be specified in LIB-LIST.

This option applies to this region's users, regardless of which TSR they access.

This option can be specified for all tableBASE interfaces.

LISTOPTIONS—List Parameter Options

Determines whether or not to list execution time parameters. If LISTOPTIONS=Y is specified, all default parameters and parameters overridden by the TBOPT dataset are listed in the JESMSG LG. LISTOPTIONS=N suppresses this list. The delivered default is LISTOPTIONS=N.

LISTOPTIONS=X is a special setting for TBOPTGEN (DK1Tnn34). It is the equivalent of LISTOPTIONS=N if the TBOPT DD was not present in the jobstream, and the equivalent of LISTOPTIONS=Y if the TBOPT DD is present. LISTOPTIONS=X applies to TBOPTGEN (DK1Tnn34) only; it does not apply to TBOPT.

This option applies to this region's users, regardless of which TSR they access.

This option can be specified for all tableBASE interfaces.

LOCKTIMERC—Lock Timer Wait Value

LOCKTIMERC=nnnnnn specifies the number of seconds (default 0) that tableBASE should wait for a lock. When the LOCKTIMERC interval has passed, RC=71 is returned or the call abends with a U071 or G071. A value of LOCKTIMERC=0 specifies that the process will never time out.

The lock controlled by LOCKTIMERC is used internally by tableBASE to maintain table integrity in the TSR. It is unrelated to the table ENQUEUE that occurs when a table is opened for write (OW).

The option applies to this region's users, who have the wait switch set, regardless of which TSR they access.

This option can be specified for all tableBASE interfaces except the VTS interface.

LOCKTIMEWTO—Lock Timer Message Wait Value

LOCKTIMEWTO=nnnnnn specifies the number of seconds (default 30) to wait before issuing messages that the process is waiting for a lock. A value of LOCKTIMEWTO=0 specifies that no warning message will be issued. To be effective LOCKTIMEWTO must be less than LOCKTIMERC.

The lock controlled by LOCKTIMEWTO is used internally by tableBASE to maintain table integrity in the TSR. It is unrelated to the table ENQUEUE that occurs when a table is opened for write (OW).

When applicable, messages are generated to report that some processing has been blocked.

The option applies to this region's users, who have the wait switch set, regardless of which TSR they access.

This option can be specified for all tableBASE interfaces except the VTS interface.

MAXNMTAB—Maximum Number of Tables

Determines the number of tables that can be opened in a given local TSR or VTS-TSR. (With the possibility of creating a TSR of up to 2G in size, there can be a great variation in how many tables will actually be in a TSR at any time.) If the value of the MAXNMTAB is not set by the user (or is set to zero) then a default value is calculated (see below).

If an attempt is made to open more tables than is indicated by MAXNMTAB, an error code of 20 will be returned to the application program opening or creating the table or alternate index.

This option applies to whichever TSR the application is accessing.

This option can be specified for all tableBASE interfaces.

Note: For a VTS-TSR region, VTSNMTAB is retained for backward compatibility and is treated as an equivalent.

User sets the value of MAXNMTAB

To optimize the memory required for system overhead, the user may choose to set the value of MAXNMTAB rather than use the default value. MAXNMTAB can be set to any integer, zero or greater. However, if MAXNMTAB is set to a value greater than the default for the given TSR size, the default is used and the user is notified with a warning message.

After the user sets the value of MAXNMTAB, memory is allocated and an index is created that contains enough entries to support the maximum number of tables indicated. Additional memory is allocated as tables are opened. This memory is reused as tables are closed and new tables opened. This approach keeps the system overhead in line with the number of tables that are opened.

The calculation of the default value of MAXNMTAB

If the user does not set the value MAXNMTAB or the value is zero, the tableBASE software calculates the default value based on the size of the TSR. Each table in the TSR occupies a minimum of 4K pages. The maximum possible number of tables is 51,455, for a 2G TSR; the default, when unspecified is one table for every 4K page in the TSR.

MTRETAIN—Retain Rows and Index Areas

MTRETAIN=Y or N determines whether the allocated rows and index areas are to be retained when an MT command is issued or whether they are to be reduced to the original allocation before table expansion. The default is "N", the additional space is not retained.

Also see [ZEROROWS—Zero Data Table Rows on De-allocation](#) on page 151.

This option applies to the TSR being accessed by this region or job.

This option can be specified for all tableBASE interfaces.

MULTITASKING—Multitasking

Specifies whether this region will allow multiple subtasks to access tableBASE.

This option can only be specified for the tableBASE batch interface. All other environments are multitasking in their nature, and will not accept this option.

The MULTITASKING option can be set to Y to allow multiple subtasks within a region to access tableBASE concurrently. Each subtask may have multiple concurrent subtasks of its own, with no limit to the level of multitasking. The delivered default is N (multitasking not enabled).

This option applies to this region's users, regardless of which TSR they access.

- Note: 1.** If a batch application attempts to issue a tableBASE command from a second TCB with Multitasking=N, it will abend with an S0C3 and display message: "DK100496E Multitasking requires TBOPT Multitasking=Y be set".
- 2.** Please contact DataKinetics Technical Support for assistance if writing applications that access tableBASE in a multitasking environment.

MULTOPNX—Multiple Alternate Index

MULTOPNX is a flag indicating whether to allow Multiple Alternate Indexes to a Data Table to be open concurrently.

This option is maintained for compatibility with previous releases, and is ignored in tableBASE starting with Version 6. The only valid value for this flag is Y and this is the delivered default.

OVERRIDES—Allow Changes to Status Switches

It is recommended that you do not allow this option to be changed using TBOPT. It should only be changed using TBOPTGEN.

The Status Switches may be changed by the application with the use of the Change Status (CS) command. Overriding individual Status Switches may be inhibited by the settings of the override controls. Overrides are specified as a string of Y's and N's; for example: YYNYNNNN. A value in any of the eight positions represents a change for that override. A * or blank in any of the eight positions represents no change for that override.

The defaults for these override values are YYYYYYYY which will allow the user to change all Status Switch values. This information is listed in [Table A-1](#) on page 140.

Position	Description	Override setting	Meaning
1	Allow Change To Abend On Errors	Y	The application is allowed to set the ABEND status switch.
		N	The application is not allowed to set the ABEND status switch.
2	Allow Change To Wait For Enqueued Tables	Y	The application is allowed to set the WAIT status switch.
		N	The application is not allowed to set the WAIT status switch.
3	Allow Change To Return Empty Rows In Hash Tables	Y	The application is allowed to set the HASH-EMPTYES-RETURNED status switch.
		N	The application is not allowed to set the HASH-EMPTYES-RETURNED status switch.
4	Allow Change To Permit Implicit Open Of Tables	Y	The application is allowed to set the IMPLICIT OPEN status switch.
		N	The application is not allowed to set the IMPLICIT OPEN status switch.

Position	Description	Override setting	Meaning
5	Allow Change To Trace tableBASE Commands	Y	The application is allowed to set the tableBASE TRACE status switch.
		N	The application is not allowed to set the tableBASE TRACE status switch.
6	Reserved for future use; default set to Y.		
7	Reserved for future use; default set to Y.		
8	Reserved for future use; default set to Y.		

Table A-1: Overrides default values

The values set by this option apply to this region's users, regardless of which TSR they access.

This option can be specified for all tableBASE interfaces.

RACF_LIBACCESS—Perform Checks on Protected Libraries

Specifies whether tableBASE should check if a tableBASE library is SAF interface protected.

This option is only available in the tableBASE Batch and IMS interfaces.

The following is a description of each parameter value:

- Y indicates that tableBASE will check whether a library is protected. Any update attempts without update access to the library will fail with error code 61-12.
- N indicates that tableBASE will not check whether a library is protected i.e. the behaviour will revert to pre-Release 7.0.3 prior to the implementation of library security checking (ref: TB-3041). Any update attempts without update access to the library will fail with a S913-38 for tableBASE BDAM libraries and a U301 for tableBASE VSAM libraries.

The delivered default for this parameter is N.

RACF_VTS—Perform Checks on SAF Interface Protected VTS-TSRs

Specifies whether tableBASE should check if a VTS-TSR is SAF interface protected.

This option is only available in the tableBASE VTSGate interface.

The following is a description of each parameter value:

- Y indicates that:
 1. Starting up or shutting down of a VTS-TSR will require an access authority of CONTROL or ALTER on the VTS

2. All tableBASE update commands on a VTS-TSR will be checked to ensure that the user has at least UPDATE access authority on the VTS
 3. All tableBASE read commands on a VTS-TSR will be checked to ensure that the user has at least READ access authority on the VTS
 4. tableBASE commands which do not fall into the READ or UPDATE categories will not require any special VTS-TSR access authority.
- N indicates that tableBASE will not check whether a VTS-TSR is SAF interface protected.

The delivered default for this parameter is N.

This option cannot be modified using the TBOPT DD statement. It can only be modified using the DK1V1134 source provided and relinking DK1VBASE. For more details on how to override the default, please see the chapter on *"Installing the VTS Gate product"* in the *tableBASE Installation Guide*.

For details on how to SAF protect a VTS-TSR to conform to tableBASE requirements, please refer to the *tableBASE Administration Guide*.

For a list of tableBASE command categories, please refer to the *tableBASE Programming Guide*.

STROBE—Strobe Interval

Specifies a numeric value, from 0 to 2147483647(2G – 1), to control the interval after which tableBASE strobe reporting is triggered. The delivered default is 0 and will produce a final STROBE report if DDNAME TBTSRPT is present.

For the VTS environment, this option is used in conjunction with STROBEMETHOD. The numeric value specified either represents a call count (STROBEMETHOD=0) or a minute interval (STROBEMETHOD=3). See [STROBEMETHOD—Strobe Method](#) on page 142 for more details.

For all other environments, the numeric value represents a call count.

The STROBE option can be specified for all except the DB2 Stored Procedure interface as strobe reporting does not work in this environment.

TBTSRPT is not relevant in the CICS environment. CICS strobe statistics are written to a CICS journal file (DFHJ99) from which the STROBE report is generated via a tableBASE utility (DK1TSTRB).

For more information on CICS strobe reporting, please refer to the *tableBASE Administration Guide*.

This option applies to the TSR created by this region or job and all users accessing it.

Note: The numeric value must be specified without commas.

STROBEMETHOD—Strobe Method

Specifies when tableBASE strobe reporting is triggered for the VTS-TSR.

This option applies only to the VTS environment.

Values for this parameter can be 0 and 3. The delivered default is 0.

The following is a description of each parameter value:

- 0 indicates that the traditional strobe reporting method, by number of calls, using the STROBE=n parameter, will occur.
- 3 indicates that strobe reporting will be triggered by time interval instead of call count. STROBE=n will indicate the number of minutes, n, after which a strobe is to be taken. Although STROBEMETHOD=3 was initially designed for hardware key protected VTS-TSRs, it can be used for any VTS-TSR.

If STROBE=n is not specified with STROBEMETHOD, the default of STROBE=0 is used and only 1 report will be produced at tableBASE termination.

STROBEMETHOD can be specified in the TBOPT DD statement or modified in TBOPTGEN source, reassembled and relinked into DK1VBASE for the VTS Agent job.

For STROBEMETHOD of 0, strobe processing has been improved so that tableBASE processing is prevented from waiting for the strobe report. If the number of calls in the STROBE parameter is set to a small value and tableBASE calls are made more frequently than strobe reports can be produced, the frequency of reports may change due to strobe reports being bypassed.

For STROBEMETHOD of 0, strobe processing has been improved so that tableBASE processing is prevented from waiting for the strobe report. If the number of calls in the STROBE parameter is set to a small value and tableBASE calls are made more frequently than strobe reports can be produced, the frequency of reports may change due to strobe reports being bypassed.

SUPPRESS_DUMPS—Suppress tableBASE Messages and Dumps

Specifies whether to suppress tableBASE messages and dumps when an abend occurs.

This option applies only to the CICS environment.

Values for this parameter can be NONE or SYMPATHETIC. The delivered default is SUPPRESS_DUMPS=NONE.

The following is a description of each parameter value:

- NONE indicates that all tableBASE messages and dumps will be produced.
- SYMPATHETIC indicates tableBASE error messages and dumps (TBDUMP and tableBASE CICS dumps) will be suppressed if a transaction abends outside tableBASE code (i.e., in the associated application code, or in some other non-tableBASE code). Any abends occurring within tableBASE code will still produce the tableBASE error messages and dumps.

SWITCHES—Status Switches

The Status Switches are a series of one-byte codes for altering the operation of the Application Programming Interface used by your programs when requesting a tableBASE service. Switches are specified as a string of Y's and N's; for example: YYNYNNNN. A value in any of the eight positions represents a change for that override. A * or blank in any of the eight positions represents no change for that override. For information about switch settings, see [Table A-2](#) on page 144.

The switches are stored in the parameter module for the interface. As each batch job or online task begins, the run-time values will be set according to the values in the parameter module but may be changed using the values in the TBOPT file. For more information, see Parameters in the MVS Batch section of the *tableBASE Installation Guide*.

The application also may change the values with the CS, the Change Status command (unless changing a particular switch is suppressed via the override controls).

The defaults for these values depend upon the interface and are described in the *tableBASE Installation Guide* in the appropriate chapter:

- tableBASE MVS batch
- CICS Interface product
- IMS TM Interface product
- DB2 Stored Procedure Interface product.

Position	Description	Switch setting	Meaning
1	Abend on Errors	Y	Abend processing is to be performed on tableBASE errors 0001-0099 or 1001-1099. Errors 100 to 999 and errors greater than 1099 always abend.
		N	Abend processing is not to be performed on tableBASE errors 0001-0099 or 1001-1099. User programs could handle these return codes.
2	Wait for Enqueued Table	Y	tableBASE is to wait for tables that are enqueued.
		N	tableBASE is not to wait for such enqueued tables. In this case tableBASE will terminate with a user 0072 abend if abend processing is enabled, or, will return an error code of 0072: TABLE UNAVAILABLE; NO WAIT in the command area if abend processing has been disabled.

Position	Description	Switch setting	Meaning
		Note: Waiting in an online environment is normally discouraged. In CICS and IMS, the default is set to N.	
3	Return Empty Rows in Hash Tables	Y	tableBASE is to return empty rows for hash tables.
		N	tableBASE is to suppress the return of empty rows for hash tables.
4	Allow Implicit Open Of Tables	Y	tableBASE is to automatically open tables for read on first access.
		N	tableBASE is to suppress automatic opens; an explicit open command, OR, OW, or IA must be issued to open a table.
		Note: In a multi-user environment like CICS, there may be advantages based on this feature.	
5	Trace tableBASE Commands	Y	tableBASE automatically records the last ten commands executed per thread. This is done for diagnostic purposes. Contact DataKinetics technical support for more information.
		N	tableBASE does not trace commands.
6	Reserved for future use; specify as N.		
7	Reserved for future use; specify as N.		
8	Reserved for future use; specify as N.		

Table A-2: Switches default values

The values set by this option apply to this region's users, regardless of which TSR they access.

This option can be specified for all except the VTS Interface.

TABLEWAITRC—Table Open Enqueue Wait Time

Specify a value to indicate the number of seconds that a user will wait to obtain the MVS enqueue to open a table for read or write before timeout. If the enqueue is not obtained before the timeout, tableBASE will return code 72, or will abend, if the *Abend on Errors* Status Switch is set to Y.

The delivered default is 0 (wait forever). See [TABLEWAITWTO—Table Open Enqueue Report Time](#) on page 146

The Wait for Enqueued Table switch (see [Table A-2](#) on page 144) must be set to Y for this parameter to have any effect.

This option applies to this region's users, regardless of which TSR they access.

This option can be specified for all interfaces except VTS.

TABLEWAITWTO—Table Open Enqueue Report Time

Specify a value in number of seconds to indicate the elapsed time before a message will be generated to report that the enqueue has not yet been received.

The delivered default is 30 seconds (0=no messages). If this value is greater than TABLEWAITRC no message will display.

The Wait for Enqueued Table Status Switch (see [Table A-2](#) on page 144) must be set to Y for this parameter to have any effect.

This option applies to this region's users, regardless of which TSR they access.

This option can be specified for all interfaces except VTS.

TPVM—VTS Group Manager Use

This parameter is not implemented for Version 7.0.3.

Specify a valid VTS Group Manager (TPVM) name to use for all accesses to VTS-TSRs in this jobstep. The name must be pre-defined, and can be up to eight characters in length.

The delivered default is *null* which maps to the *compat* VTS Group Manager.

This option applies to the TSR created by this region/job and all users accessing it.

Note: This parameter applies only if you have licensed the optional VTS Manager product.

TSR_ALGORITHM—Optimize TSR Usage

TSR_ALGORITHM=P or D allows the TSR Space Manager to determine if TSR space usage should be maximized or performance should be optimized when TSR space is allocated and deallocated for table usage. The TSR space manager is the tableBASE component responsible for tracking TSR space usage and for allocating and deallocating TSR space when tables are opened, closed, expanded or reduced in size.

The following is a description of each parameter value:

- P (Performance) indicates that the TSR will be optimized for performance. Space will be assigned to tables within the TSR so as to minimize CPU usage, which may result in a less than optimum use of space.
- D (Default) indicates that there will be no optimization, and there will be no messages regarding optimization. However, messages will be provided regarding current percentage of TSR capacity.

The delivered default is D.

This option applies to the TSR created by this region/job and all users accessing it.

This option can be specified for all tableBASE Interfaces.

TSR_KEY_PROTECT—Hardware Key Protection for VTS-TSRs

Specifies whether a VTS-TSR is hardware key-protected.

This option is only available in the tableBASE VTSGate interface.

The following is a description of the parameter value:

- 0 indicates that hardware key-protection is off. The VTS-TSR behaves as it did in previous releases. The delivered default for this parameter is 0.
- Any value from 1 to 7 indicates that:
 - hardware key-protection is active
 - the TSRACCESS option for the VTS will automatically be set to RO if not specified (see [TSRACCESS—Read Only \(RO\), Refreshable \(RF\), and Read Write \(RW\) TSR Access](#) on page 148)
 - STROBEMETHOD must be set to 3 (see [STROBEMETHOD—Strobe Method](#) on page 142) and tableBASE strobe reports for the VTS will be displayed by minute intervals instead of by call count
 - only authorized batch programs will be allowed to perform tableBASE update commands on the VTS-TSR, if the VTS has been switched to a RW state
 - tableBASE commands that do not update the VTS will be allowed without any special authorizations.

This option cannot be modified using the TBOPT DD statement. It can only be modified using the DK1V1134 source provided and relinking DK1VBASE. For more details on how to override the default, please see the chapter on "*Installing the VTS Gate product*" in the *tableBASE Installation Guide*.

For more details on hardware key-protection features, please refer to the *tableBASE Programming Guide* and the *tableBASE Administration Guide*.

TSR_WARNING_FREQ—Frequency of TSR Allocation Warnings

Specify a value to indicate the frequency of messages generated when the TSR allocation percentage (as defined by TSR_WARNING_PCT) is exceeded. 0 results in a message for every allocation over the specified percentage; 30 results in a wait of at least 30 seconds before repeating the message; 999 results in a wait of just over 16 minutes before repeating the message.

The delivered default is one warning every second (001).

This option applies to this region's users, regardless of which TSR they access. (Different regions or jobs accessing a VTS-TSR can set their own thresholds.)

This option can be specified for all tableBASE Interfaces.

TSR_WARNING_PCT—Percentage of TSR Allocation for Warning Activation

Specify a value to indicate the percentage of TSR allocation allowed before a message will be generated to report that the allocation percentage has been exceeded. The message will be repeated on every allocation of this percentage, subject to the TSR_WARNING_FREQ parameter.

The delivered default is 85 (85%).

This option applies to this region's users, regardless of which TSR they access. (Different regions or jobs accessing a VTS-TSR can set their own thresholds.)

This option can be specified for all tableBASE Interfaces.

TSRACCESS—Read Only (RO), Refreshable (RF), and Read Write (RW) TSR Access

Specifies whether a VTS-TSR will have Read-Only, Refreshable, or Read-Write access.

This option is only available in the tableBASE VTSGate and tableBASE VTSManger products.

The following is a description of each parameter value:

- RO (Read-Only) indicates that:
 1. only authorized batch programs will be allowed to perform tableBASE update commands on the VTS-TSR, if the VTS has been switched to a RW state
 2. tableBASE commands that do not update the VTS will be allowed without any special authorizations.
- RW (Read-Write) indicates that all tableBASE commands are allowed on this VTS-TSR.
- RF (Refreshable) indicates that it is a read only VTS-TSR. The RF Command is the only update command allowed.

The delivered default is *RW*.

For VTSGate, this option cannot be modified using the TBOPT DD statement. It can only be modified using the DK1V1134 source provided and relinking DK1VBASE. For more details on how to override the default, please see the chapter on *"Installing the VTSGate product"* in the *tableBASE Installation Guide*.

For a list of tableBASE command categories, please refer to the *tableBASE Programming Guide*.

TSRSIZE—tableSPACE Region Size

The TSRSIZE parameter is an integer representing the amount of storage to be used for the TABLE SPACE REGION (TSR). tableBASE uses Data Spaces for all TSRs, whether local or

VTS. This ensures that tableSPACE memory requirements do not affect other memory requirements in the region. The format for TSRSIZE is:

- nnnnnnnn = bytes
- nnnnnnnnK = kilobytes
- nnnnM = megabytes
- nG = gigabytes

Value recommendations

The delivered default value for the TSRSIZE is 100M for all interfaces. The current minimum size of a TSR is 28 KB (but this may change in subsequent maintenance releases). The maximum size of a TSR is 2G (or 2048M or 2097152K). TSRSIZE only accepts up to eight digits, thus you cannot specify 2G as TSRSIZE=2147483648.

Note: If you specify a value of 0, the minimum size will be substituted; if you specify a value of between 1 and the minimum, an error code will be returned.

Failure to set your TSR size parameter appropriately can produce the following error:

Error 92 There is insufficient tableSPACE region available. Increase TSR size.

- Note:**
1. TSREGION and TSR parameters are maintained for backwards compatibility and are treated as equivalents to TSRSIZE.
 2. There is a possibility that your system programmers have established limits on the maximum size or the maximum number of data spaces. All local TSRs are allocated as SCOPE=SINGLE data spaces; VTS-TSRs are allocated as SCOPE=ALL data spaces.

This option applies to the TSR created by this region or job and all users accessing it.

This option can be specified for all tableBASE Interfaces.

USEREXITS—User Exits

Specifies whether User Exits are used. Setting the tableBASE execution-time parameter USEREXITS=Y causes the DK1TX072 module to be dynamically loaded during the initialization of tableBASE. For more information, see *tableBASE User Exits* in the *tableBASE Administration Guide*.

This option applies to this region's users, regardless of which TSR they access.

This option can be specified for all interfaces except VTS.

VTSFIRST, VTSLAST—VTS Search Sequence

VTSFIRST and VTSLAST are two parameters that can be used to control the sequence in which tableBASE searches libraries when DataKinetics VTS product is installed.

VTSFIRST is the name of the VTS-TSR that is to be searched for tables before searching any libraries listed in the tableBASE library-list. VTSLAST is the name of the VTS-TSR that is to be searched for tables after searching any libraries listed in the tableBASE library-list.

If you wish to override the installation default for this parameter, it is strongly recommended that you use the TBOPT DD statement described later in this chapter to do so. If you set the VTSFIRST parameter by tailoring the installation parameters at installation time, problems may occur. For example, any processes that operate on the library copy of a table that is open in VTS may abend with a tableBASE error code of 3 or may return unpredictable results.

Note: If VTSFIRST or VTSLAST names are specified in the default parameters (DK1Txx34), you can override these names in the TBOPT dataset by setting VTSFIRST=0 or VTSLAST=0, for the specific region or job to turn the respective parameters off.

The delivered defaults are VTSFIRST, VTSLAST set to null strings. This sets them to be ignored.

This option applies to this region's users, regardless of which TSR they access.

To see how this method to access VTS-TSRs compares to other methods see [Virtual Table Share \(VTS-TSR\)](#)

These options can be specified for all interfaces except VTS.

Note: These parameters apply only if you have licensed the optional DataKinetics VTS product. Do not change the delivered defaults supplied for these parameters if you have not licensed VTS.

VTSNAME—Specifying the Name of a VTS-TSR

The VTSNAME option specifies the name to be assigned to the VTS-TSR when it is initialized by the VTS Agent. The name is used by other regions to access the shared TSR.

- Note:**
1. VTSNAME can be 1 to 8 characters. The naming restrictions are identical to those for DDNAMEs in IBM JCL. VTSNAMEs of 1 to 4 characters created in prior releases are still supported.
 2. If a VTSNAME is to be used in the ML List, care must be taken when assigning them. The combined length of the VTSPREFIX and the VTSNAME cannot exceed eight characters (see [VTSPREFIX—VTS Prefix Definition](#) on page 151 below).
 3. For compatibility with previous releases, where it was specified in the TBOPTV dataset of regions accessing VTS-TSRs, VTSNAME can be used to specify the default VTS-TSR accessed by the TBCALLV and TBASEV interfaces. Note that the tableBASE options for applications using TBCALLV and TBASEV may have changed. See [Virtual Table Share \(VTS-TSR\)](#) in the *tableBASE Programming Guide* for additional information.

This option applies to the TSR created by this region or job and all users accessing it.

VTSPREFIX—VTS Prefix Definition

The VTSPREFIX option specifies a prefix that will be used by tableBASE to denote a VTSname within the ML list.

The delivered default is *VTS:*. There can be 0-3 characters before the colon; the colon is mandatory. Example valid values:

- VTS:
- V:
- :

The maximum number of characters in total for the VTSPREFIX and the actual VTS name is eight. Therefore, a VTS name of eight characters could never be used in conjunction with the ML command. Example valid combinations:

- VTS:SHAR
- V:SHARE1
- :SHARE99

This option applies to this region's users, regardless of which TSR they access. See [Virtual Table Share \(VTS-TSR\)](#) for more information.

This option can be specified for all except the VTS interface.

ZEROROWS—Zero Data Table Rows on De-allocation

Related to the MTRETAIN parameter (see [MTRETAIN—Retain Rows and Index Areas](#) on page 139)—applies only when MTRETAIN = N.

ZEROROWS=Y or N determines whether the data rows area should be zeroed when it is deallocated. The default is Y, which is current processing. Note that the index area is never zeroed, except for hash indexes.

This option applies to the TSR created by this region or job and all users accessing it.

This option can be specified for all tableBASE interfaces.

Appendix

B

Setting up an organization

This appendix describes the initial set up of a tablesONLINE/CICS environment for a hypothetical organization. Topics such as the process of determining the individual authorities, defining the environment, and making the necessary entries into the Application Control table, User/Application Relationship table and Cross-Authorization table are discussed.

Defining authorities

Consider a hypothetical organization of people and departments as outlined below. All employees have CICS signon IDs, which are displayed in parentheses. All employees of the company not listed below are to have browse-only access to the Plant tables.

Plant department

The following individuals must be able to browse and to edit Plant tables.

- Plant manager (PL99)
- Foreman 1 (PL01)
- Foreman 2 (PL02)

Finance department

The following individuals must be able to browse and edit Finance tables. In addition, the finance manager must be able to assist the accountants by accessing their sessions while in progress.

- Finance manager (FI99)
- Accountant (FI01)
- Accountant (FI02)

Technical support department

Both of these individuals require unrestricted access to the system since they are responsible for tableBASE customer support.

- Technical Support 1 (TS01)
- Technical Support 2 (TS02) Backup

Application development department

Everyone in this department must be able to sign on to tablesONLINE/CICS as a Finance application user and a Plant application user in order to test and debug the user systems.

The following individuals must be able to sign on to tablesONLINE/CICS as an application developer.

- Development Manager (AD99)
- Developer 1 (AD01)
- Developer 2 (AD02)
- Developer 3 (AD03)

User AD01 has been designated as the tableBASE administrator who is responsible for setting up access to tablesONLINE/CICS and must have unlimited access to the system.

Defining the environment

All of the tables for the Plant application are stored on PLANTLIB, and those for the Finance application are stored on FINALIB. MAINLIB is reserved for administrative and application development. Each DDNAME points to a different tableBASE library.

[Table B-1](#) on page 154 lists the application IDs that are required to address the security and organizational needs of the Organization.

Application ID	Description
PLTB	Browse access to Plant tables for all employees.
PLTE	Update and edit access to Plant tables.
FINA	Finance staff with update/edit access to Finance tables.
TECH	Technical support people with access to all tables.
APPL	Application developers with update access to the Application-Driving Tables library. Application developers will not be defining new applications and do not require access to the Application Control Table library.

Application ID	Description
ADMN	the tableBASE administrator must be able to define new applications, therefore, the ADMN application requires update access to both the Application Control tables (on TBACTLB) and the application-driving tables (on TBAPPLB).

Table B-1: Application IDs and hypothetical organization

Note: Since the applications PLTB and PLTE only differ in their access specifications, they will use the same application and the same starting menu.

The APPL and TECH application IDs will use the same menu table as the tableBASE administrator, but with a different starting menu.

Note: The Menu tables, Command table, Help tables and other application-driving tables can be specific to each application or they may be shared by all applications. For this example, it is assumed that all application-driving tables will be shared, except the Menu table. There will be Menu tables for each of the two main applications and one for administrative functions.

Defining users and applications

There are three tables that are used to define users and applications to your tablesONLINE/CICS system. These tables are the Application Control table (TBOLACT), User/Application Relationship table (TBOLUSAP), and the Cross-Authorization table (TBOLXAUT). The following sections illustrate the entries which must be made to these tables based on the requirements set out above.

Application Control table

The first step is to define the applications by entering the Application IDs and related control information in the Application Control table. [Table B-2](#) on page 155 is an abbreviated View of the TBOLACT. It shows the entries that must be specified in the ACT for the organization.

Application ID	User ID	Start screen name	Start select symbol	tableBASE libraries	Application's Menu table	Update table indicator
PLTB PLTE FINA TECH		PLNTPRIM PLNTPRIM FINANCE TB ADMIN		PLANTLIB PLANTLIB FINALIB TBACTLB TBSYSLB MAINLIB PLANTLIB FINALIB	PLNTMENU PLNTMENU FINAMENU APPLMENU	N Y Y Y
APPL		APPLDEVL		TBAPPLB TBSYSLB MAINLB	APPLMENU	Y

Application ID	User ID	Start screen name	Start select symbol	tableBASE libraries	Application's Menu table	Update table indicator
ADMN		TB ADMN		TBACTLB TBAPPLB TBSYSLB MAINLB	APPLMENU	Y

Table B-2: Application Control table

For each application ID, we have identified:

- the starting screen name or starting menu,
- the tableBASE libraries to be accessed by the application, and
- a custom menu table.

Note: Although only unique Menu tables have been specified for these applications, customized versions of any of the application-driving tables, such as DESC or PFKS, may be assigned to an application. For more information on customizing these tables, see the *tablesONLINE/CICS User's Guide*.

The Update Table indicator controls the type of access permitted for tables accessed under this application ID. Since only browse access is required under the PLTB application ID, the Update Table indicator is set to N for this entry. Since update access is required in all other cases, the update indicator is set to Y for all other application ID entries.

There are additional options that may be set in the Application Control table. See [tablesONLINE/CICS Administration](#) on page 83.

In this table, we have left the user ID field for all the entries in the TBOLACT blank. If there are special privileges or libraries required for a specific user of an application, the specific application entry may be duplicated, modified as required, and identified by specifying a user ID in the appropriate field.

User/Application Relationship table

For this organization, we have defined the relationships between the user IDs and application IDs in the User/Application Relationship table.

When a user signs on to tablesONLINE/CICS, the User/Application Relationship table is the first table searched. The user ID and application ID are used as the search key for this table. If a match is found, the application identified as the target application ID is started using the settings specified in the TBOLACT. If no match is found, both user and application IDs are set to spaces. If there is a default entry in the User/Application Relationship table with a target ID and both the application ID and the user ID are set to blanks, that application is started.

All users for whom there is a user ID are presented with either the particular application based on their application ID, or, in the case of a mismatch (or no application specified), the default application.

[Table B-3](#) on page 157 shows the entries to be made for our hypothetical organization, based on the users and environment described earlier.

Signon user ID	Signon application ID	Target application ID
bbbb	bbbb	PLTB
AD01	bbbb	ADMN
AD01	APPL	APPL
AD01	FINA	FINA
AD01	PLTE	PLTE
AD02	bbbb	APPL
AD02	FINA	FINA
AD02	PLTE	PLTE
AD03	bbbb	APPL
AD03	FINA	FINA
AD03	PLTE	PLTE
AD99	bbbb	APPL
AD99	FINA	FINA
AD99	PLTE	PLTE
FI01	bbbb	FINA
FI02	bbbb	FINA
FI99	bbbb	FINA
PL01	bbbb	PLTE
PL02	bbbb	PLTE
PL99	bbbb	PLTE

Signon user ID	Signon application ID	Target application ID
TS01	bbbb	TECH
TS02	bbbb	TECH

Table B-3: User/Application Relationship table (TBOLUSAP) for hypothetical organizations

Note: In the table above, bbbb is used to represent blanks.

Note: Wildcard characters cannot be used for Signon user IDs.

The first entry in the User/Application Relationship table has both the user ID and the application ID set to blanks. It allows any user who is not specifically defined to the system to sign on. Any user with no other entry in this table will default to the application PLTB, the browse version of the Plant application.

Users like AD01 who are authorized to sign on to multiple applications have multiple entries in the User/Application Relationship table. These entries include a default application (ADMN in the case of AD01), indicated in the target application field, for that user. This default application is accessed when the user signs on without specifying an application ID.

There is also an entry in the User/Application Relationship table for each additional application that the user can access. These entries contain both the user ID and the application ID defined explicitly. When a user signs on to tablesONLINE/CICS specifying both a user ID and an application ID, the target application identified by those IDs will be accessed.

Defining Cross Authorizations

The set up of the hypothetical organization requires that the Finance manager be authorized to sign on to the system using the IDs of the each of the accountants. In order for this to occur, there must be an entry in the Cross-Authorization table matching the signon user ID for the Finance manager (F199) to the target user IDs for each accountant (F101 and F102). [Table B-4](#) on page 158 lists the entries in the table:

Signon user ID	Target user ID
F199	F101
F199	F102

Table B-4: Cross-Authorization table (TBOLXAUT) for hypothetical organizations

When the finance manager signs on, specifying a blank application ID and a target user ID, TBOLbbbbF102, the Cross Authorization table is checked. When an entry of FI99F102 is found in the table, the sign on proceeds as if F102 were signing on. If F102 is currently signed on, his session is suspended and brought up on the finance manager's screen. This allows the finance

manager to examine the accountant's work from another terminal. If F102 is not currently signed on to the application, a new session is initiated.

Appendix

C

tableBASE messages

This appendix contains all of the DataKinetics tableBASE error codes and messages that can be encountered during the normal installation, administration and operation of the product. The error codes and messages fall into the following categories:

tableBASE error codes—error conditions that may be encountered while running tableBASE (see [tableBASE return codes](#) on page 161).

tableBASE messages—conditions that may be encountered with running tableBASE in conjunction with the VTS agent, CICS, batch, IMS, and other programs/utilities (see [tableBASE messages](#) on page 178).

tablesONLINE/CICS error messages—conditions that may be encountered while running tableBASE in conjunction with the tablesONLINE/CICS application (see [tablesONLINE/CICS error messages](#) on page 194).

tablesONLINE/ISPF error messages—conditions that may be encountered while running tableBASE in conjunction with the tablesONLINE/ISPF application (see [tablesONLINE/ISPF error messages](#) on page 218).

DK1TEXEC error messages—conditions that may be encountered while running tableBASE in conjunction with the DK1TEXEC batch utility program (see [DK1TEXEC error messages](#) on page 230).

Note: More messages for the DataKinetics VTS Manager product can be found in the *VTS Manager Administration Guide*.

tableBASE return codes

The following tableBASE error codes and subcodes are returned by tableBASE as a result of events detected in the processing of commands. The error code is returned in the ERROR field of the COMMAND-AREA parameter, and the subcode is returned in the ERROR-SUB-CODE field.

To distinguish tableBASE abend codes from those of the operating environment when running tableBASE with CICS, the tableBASE abend codes are prefixed with a G for abends under 1000 (for example, G009) and prefixed with V for abends over 1000, where V replaces the 1 (for example, 1072 becomes V072). In all other environments, tableBASE abend codes appear as listed.

In addition, tableBASE internal logical errors are indicated in CICS as LGIC abends and in other environments as U301 abends. If an abend causes the tableBASE control blocks to be purposely dumped by tableBASE, the transaction dump code generated in CICS is DKL1.

Additional errors generated by VTS (prefixed by TBV1) and tablesONLINE (prefixed by TB-) may also be encountered.

If the "Abend on Errors" switch is set to 'N' the return code and subcode are returned to the caller in the command area. If the "Abend on Errors" switch is set to 'Y' the thread will abend, and a message will be written to the JES log. Below is an example:

```
THE SPECIFIED LIBRARY IS NOT SUITABLE          RC = 0060 SC = 0006
```

In this example, you would reference 60-06 in the following table. Depending upon the error condition, there may or may not be a subcode.

In the event that the application thread abends, the tableBASE error code is used as the user abend value, and any associated error subcode is used as the abend reason code.

Return Code	Text	Meaning / Instructions
00-00	tableBASE command executed successfully	No action is indicated.
00-01	Hash High and/or Low Density values for table adjusted based on region limits	tableBASE command executed successfully. Table density values were substituted with region limits because: - table high and low density values from library were greater than HASH_HI_DEN_LIM or less than HASH_LOW_DEN_LIM respectively - low density value may not have been greater than 2/3 of the high density value.
00-03	The table specified on the open command was already open.	tableBASE command executed successfully. Table specified was already open in the designated TSR.
01-00	The specified command is invalid.	The two-character Command code specified is not a valid tableBASE command.
02-00	The specified table must be open for this command.	Open the table before processing this command.
02-01	The table is in the process of being opened by another process.	This table is being opened and this command cannot wait for the open to complete. Retry the request.
02-98	Invalid parameter(s) were specified for the CL command.	Correct the parameters and resubmit the command (the CL command does not take any parameters).
03-00	The table is not closed.	The table must not be open in the TSR for this command.
03-01	The command requires that the table be closed	Close the table in the TSR and resubmit the command.
03-02	The table specified by the LD command has an incorrect row size.	Correct the row size for the LD table and resubmit the command. The row size depends on the DIRTYPE specified; see the <i>Programming Guide</i> .

Return Code	Text	Meaning / Instructions
03-03	The table specified by the LD command has an incorrect key size.	Correct the key size for the LD table and resubmit the command. The key size depends on the DIRTYPE specified; see the <i>Programming Guide</i> .
03-04	The table specified by the LD command has an incorrect key location.	The key location for the table created by the LD command is always 1. Correct the table definition and resubmit the command.
03-05	The table specified by the LD command is an alternate index.	The table used by an LD command must be defined as a data-table. Correct the table type and resubmit the command.
03-06	The table specified by the LD command is a linked VTS table	The table used by an LD command cannot be a linked VTS table.
03-07	The library specified by the LD command cannot be a VTSNAME	The LD command only works with tableBASE libraries on DASD. LD command usage with a VTS-TSR in the library list is not supported. Use the LT command to list tables in a VTS-TSR.
03-08	The table is already open from a library with a different DSN	The table cannot be opened from the specified DSN. It is already opened in the TSR from a different DSN.
03-09	The table is in the process of being opened by another process.	This table is being opened. Retry the request after closing the table.
03-10	The command requires that the table be closed.	The table is open in the local TSR. Close the table and resubmit the command.
04-00	The FG command is not allowed; the table is not sequential (S or D)	Change the table organization to sequential (S or D) and resubmit the command.
05-00	The indirect open search criterion was not found in the primary table	The indirect open search criterion set by the SI command did not match any keys in the primary table specified on this command. See the <i>Programming Guide</i> for details on using INDIRECT-OPEN.
06-00	The count specified is invalid	Correct the COUNT value in the COMMAND AREA and resubmit the command.
06-01	The COUNT must be a positive number.	The COUNT value in the COMMAND AREA cannot be negative or zero. Correct and resubmit the command.
06-02	The COUNT exceeds the number of rows	The COUNT value in the COMMAND AREA cannot exceed the number of rows in the table (except for hash tables). Correct and resubmit the command.
06-03	The COUNT points to an empty row in a hash table.	The COUNT value in the COMMAND AREA for RC and DC commands for a hash table must point to a valid row.
07-00	The library DSN or DDNAME has been changed. Use DV or DW before STore.	If the library DSN or DDNAME for an open table has changed, the DV or DW command must be issued for the table before the table can be STored.

Return Code	Text	Meaning / Instructions
08-00	The generation number specified for this table is invalid.	Correct the generation number specified and resubmit the command. Generation numbers can be specified as absolute numbers from 1 to 255 or as negative numbers relative to current generation. The current generation is 0. (See the <i>Programming Guide</i> for details on specific commands)
09-00	The table is not found.	The table is not open in the TSR and cannot be found in the current library list (ML-list).
09-01	The table could not be found in any of the libraries in the ML list.	Ensure the ML-list includes the library where the table can be found and resubmit the command.
09-01	Implicit open is not permitted when a VTS-TSR access is read-only or refreshable.	All tables read from a tableBASE library should be open while the VTS-TSR access is read-write before changing the access to read-only or refreshable.
09-02	The table specified could not be found in the linked VTS-TSR.	The table search is linked to a table in a VTS-TSR and the table is not open there.
09-03	The table specified could not be found in the designated library.	Either ensure the ML-list includes the library where the table resides or ensure the table is resident in the currently designated library. If the command specifies a specific library DDNAME, ensure the table exists on that library.
10-00	The STATUS-SWITCHES parameter provided for the CS command is invalid	Correct the STATUS-SWITCHES parameter and resubmit the command. (See the <i>Programming Guide</i> for details on specific switches)
11-00	The Number of Generations to keep in the DT parameter must be 1 to 9.	tableBASE supports up to 9 generations of a table on a library. The number of generations to keep in the DT parameter must be 1 to 9. Correct the value and resubmit the command.
12-00	The table name specified is invalid.	A valid table name is a string of 8 bytes that are not all blanks, all low values, all high values, or :TMPNAME. Please refer to the <i>Programming Guide</i> for more details. In some applications, such as TBOL/CICS, more restrictive rules may be implemented.
13-00	The command is invalid in this environment	The two-letter command code is invalid. It is not supported in this execution environment.
13-01	This command is only supported for a VTS Agent.	The strobe-request internal command is only supported in the VTS Agent.
13-02	The DL (Define Library) command is not supported in CICS.	The DL command is only supported in the Batch interface.
13-03	Library expansion process may not be initiated from CICS.	The library expansion process is only supported by the DK1TEXEC (DK1TEXEC) Batch utility running in a non-multitasking environment.
13-05	Updates not permitted on a linked VTS table.	Only READ access is permitted on a table linked to a table in a VTS-TSR. (See the <i>Programming Guide</i> for more details on linked tables)

Return Code	Text	Meaning / Instructions
13-06	Update commands are not permitted when a TSR is read only.	Update commands are not permitted when TSRACCESS=RO (TSR is read only).
13-07	Command not supported in this environment.	The command specified is not supported in this environment.
13-08	tableBASE Library Access is not allowed for this interface.	Access to a tableBASE Library is not allowed for this interface. Refer to run time option ALLOW_TB_LIB_IO for further details (see <i>tableBASE Administration Guide</i> or <i>tableBASE Programming Guide</i>).
13-11	The command is invalid against a VTS-TSR.	When a VTS is being accessed either directly via the TBPARM or as a linked list, this command is invalid.
14-00	The ROW-SIZE specified is invalid. Must be from 1 to 32767.	The row size specified must be from 1 to 32767. Correct it and resubmit the command.
15-00	The KEY-SIZE specified is invalid. Must be from 1 to 256 and fit in row.	The key size specified must be from 1 to 256, and must fit in the row based on key location. Correct the key-size and resubmit the command. Note that date-sensitive processing can affect this.
16-00	The KEY-LOCATION specified is invalid. Must be from 1 to row-size.	Correct the key-location specified and resubmit the command. Note that the key location must allow the key size to fit within the row size.
17-00	The key will not fit within the row.	The key will not fit within the row because either the key's location in the row or its length is beyond the bounds of the row. Correct the key-size/key-location and resubmit the command. Note that date-sensitive processing can affect this.
18-00	Insufficient space is available on the tableBASE library.	The library must be enlarged. A table cannot be stored on the library because there is insufficient free space available. Note that during the processing of a Store command there must be room in the library to hold 2 copies of the table on the library. There are 2 options: 1) create a new larger library and copy the existing library into it, or 2) expand the library. Both options are supported by the Batch Utility DK1TEXEC (DK1TEXEC). (See the <i>Batch Utilities Guide</i> for details.)
19-00	This table already exists on the target library.	A new table cannot be STored on a library when it already exists on that library. Preceding the STore with a DW command will allow the table to overlay the previous table.
20-00	The maximum number of open tables has been exceeded. Check MAXNMTAB.	The MAXNMTAB option for the region must be increased to allow more tables into the TSR. Temporarily, tables can be CLosed. To change this value permanently, override the value in a TBOPT DD or update DK1Txx34. (See the <i>Installation Guide</i> for the latter option.)
21-00	The Organization specified is invalid, must be blank, R, U, S, D, H. For date-sensitive processing, Organization can only be S.	Correct the organization specified and resubmit the command.

Return Code	Text	Meaning / Instructions
28-00	Paged tables are not supported in V6.	Paged tables are not supported starting with Version 6. All paged tables were converted to resident during the upgrade to V6. Storage Mode Code (SMC) of "P" (for Paged) is no longer allowed on the DT or CD command.
29-00	The secondary table is not located on same library as primary table	When the Open Indirect option is invoked for an Open command, the tables to be opened must reside in the same library as the primary table. (See the <i>Programming Guide</i> for details on Open Indirect, or the SI command for more details).
30-00	The password supplied is invalid.	Accessing this table on a tableBASE library requires a valid password. Supply the correct password and resubmit the command.
31-00	The write password is either missing or incorrect.	Opening this table for write access from this tableBASE library or updating it on this tableBASE library requires the Write Password. Correct the password and resubmit the command.
32-00	The ST command requires that the table be opened for write.	A table can be STored on a tableBASE library only if it is Open for Write in the TSR. Execute the OW command before attempting to STore the table.
33-00	A different generation of the table is already open.	Two generations of the same table cannot be open in a TSR at the same time. Close the table and open it again in the region in order to access the latest generation on the library.
39-00	The RN command failed. The new name already exists in the library.	The table rename (RN) command failed because a table with the new name already exists in the library.
40-00	The library DDNAME does not exist.	A library name specified in the library list (ML-list) or a command parameter does not correspond with an allocated DDNAME.
40-01	The specified library DDNAME is not allocated.	Correct the library names in the library list (ML-list) or the command parameter and/or allocate the DDNAME in the region (add a DD statement to the JCL) and rerun the job. For CICS, this could be due to a missing entry in the FCT (File Control Table for BDAM libraries) or the CSD (CICS system definition file for VSAM libraries). The CICS region will need to be recycled for newly added FCT entries.
40-02	The ST command failed because an ML entry is required.	When a new table is defined in the TSR without any library DDNAMEs in the library list, the library on which the table is to be stored must be provided either by a DV or DW command before the STore command is issued or by specifying a valid library DDNAME in the ML-list in effect when the STore command is issued.
40-03	A blank or null DDNAME was used as a library list entry.	A library DDNAME must be specified in the library list (ML-list) in effect when the command is being invoked.
40-04	The CA command failed because an ML entry is required.	A CA command stores the alternate index definition on the first library in the library list (ML-list). A null or blank library list is not valid for this command.

Return Code	Text	Meaning / Instructions
40-05	The library list DDNAME entry is not allocated.	The current library list (ML-list) contains an entry that does not correspond to an allocated DDNAME. Correct the library list or add the DDNAME to the JCL and rerun or ALLOCATE with AL. For CICS, this could be due to a missing entry in the FCT (File Control Table for BDAM libraries) or the CSD (CICS system definition file for VSAM libraries). The CICS region will need to be recycled for newly added FCT entries.
41-00	The STORAGE-MODE-CODE (SMC) specified must be R or blank.	Change the SMC value to "R" and resubmit the command. "P" for Page Table is not supported in V6.
42-00	The TBPARM sub parameter TB-FORMAT must be either 0 or A.	Correct the value in the TB-FORMAT sub-parameter and resubmit the command. (See the <i>Programming Guide</i> for details on TBPARM sub-parameters.)
43-00	The estimated NUMBER-OF-ROWS value in the DT block is out of range.	When a table is defined, the estimated number of rows times the row size cannot exceed 2G. It also cannot be negative.
44-00	The EXPANSION-FACTOR specified in the DT block must be from 1 to 999.	Correct the value in this field and resubmit the command. Consult the <i>Programming Guide</i> for the usage of this parameter.
49-00	This command requires more parameters than were supplied.	Supply the minimum number of parameters required for this command. Consult the <i>Programming Guide</i> and resubmit the command.
50-00	The DIRTYPE specified for the LD command must be V, D, T, L, or blank.	Correct the DIRTYPE used by the LD command and resubmit the command. (Consult the <i>Programming Guide</i> for the usage of the DIRTYPE parameter)
51-00	HIGH/LOW DENSITY specified in DT block must be from 1 to 999.	Correct the DENSITY value being used and resubmit the command. Consult the <i>Programming Guide</i> for the usage of these parameters. Note that the specified HIGH-DENSITY and LOW-DENSITY parameters can be adjusted dynamically by the HASH_HI_DEN_LIM and HASH_LOW_DEN_LIM run-time options.
55-00	The SEARCH-METHOD must be either S, Q, B, C, or H.	Correct the SEARCH-METHOD value being used and resubmit the command. Consult the <i>Programming Guide</i> for the usage of this parameter.
56-00	The SEARCH-METHOD is incompatible with ORGANIZATION specified.	Correct the SEARCH-METHOD and/or the ORGANIZATION field ensuring they are compatible and resubmit the command. (Consult the <i>Programming Guide</i> for the relationship of these two parameters).
58-00	The requested module cannot be found in the load libraries searched.	Add the missing module to the load libraries accessed through STEPLIB, LPA or linklist.
60-00	The specified library is not suitable.	The dataset specified is not a valid tableBASE library.

Return Code	Text	Meaning / Instructions
60-01	The specified library's data definition must be DASD	Ensure that the library DDName is allocated to DASD, and resubmit the command. Note that if you are using VSAM tableBASE libraries with LSR, they are not supported in the tableBASE Batch interface. For more information, see the "Best Practices" chapter in the <i>tableBASE Administration Guide</i> .
60-02	The specified library's BLKSIZE must be at least 600.	Increase the library BLKSIZE to at least 600 and resubmit the job.
60-03	The specified library must contain at least 10 blocks.	Increase the number of blocks in the library; 10 is the minimum. Correct the allocation and resubmit the command.
60-04	The number of allocated blocks exceeds the limit for a tableBASE library.	Reduce the number of blocks in the library. The maximum for a VSAM library dataset is the number of blocks that can be contained in 65,535 tracks (approximately 4369 cyl). Contact DataKinetics technical support for assistance.
60-05	The BLKSIZE of the source and target libraries differ.	The process being invoked requires that the source and target libraries have the same block size. Correct the block sizes and resubmit the command.
60-06	The size of the designated library is smaller than that of the source.	Reallocate the target library dataset to contain the same as or more blocks than the source dataset and rerun the job.
60-07	The specified library's BLKSIZE is not known	The library is not a valid tableBASE library. Ensure that the library has been successfully DEFINED with either the DL command or the DK1TEXEC (DK1TEXEC) Batch utility. Rerun this job after the library is successfully defined.
60-08	The library expansion is not complete. Recreate the library.	An I/O error was experienced performing the library expansion. Recreate the library and resubmit the job.
60-09	The target library cannot be a V5 Bridge library.	Library Expand failed. Ensure the target library is a V6 format library and resubmit the job.
61-00	The library status is invalid.	The tableBASE library DD is not allocated appropriately
61-01	The library DDNAME may not be concatenated with another dataset.	tableBASE Library DDNAME concatenation is not supported. Ensure each library has a separate DDNAME and resubmit the command. Note that tableBASE VSAM libraries using LSR (Local Shared Resources) is not supported by the tableBASE Batch interface.
61-02	The DSNNAME is open with another DDNAME.	Only 1 DDNAME is allowed per library DSNNAME. Correct the JCL and resubmit the job. Note that tableBASE retains the reference of a DDName to a DSName for the life of a region; that relationship is established the first time the library is accessed in the region. This applies only to DDName and DSName relationships established using JCL.
61-03	The DL command or library expansion failed, the library was open.	The library was already in use. Ensure the process has exclusive use of the library and resubmit the job.

Return Code	Text	Meaning / Instructions
61-04	The DL command or library expansion failed, DISP must be NEW or OLD.	Correct the library disposition in the JCL and resubmit the job.
61-05	The DL command failed; a VSAM library was not defined as reusable.	Redefine the VSAM library as reusable and resubmit the job.
61-06	The designated library access method must be BDAM or VSAM.	Correct the DSORG of the library and resubmit the job.
61-07	The library DISP must be OLD or SHR for this command.	Correct the library disposition in the JCL and resubmit the job.
61-08	OPTCD=C and Z mutually exclusive on tableBASE library DD statement.	Correct the OPTCD DD statement and resubmit the job. See the <i>Administration Guide</i> for documentation on tableBASE use of the OPTCD subparameters C and Z.
61-09	The BDAM library must be RECFM=F or FB.	This error will be returned in the following cases: 1. when initializing a BDAM library and RECFM is not F 2. when accessing a BDAM library and LRECL not equal to BLKSIZE. For BDAM library definitions, only RECFM=F is allowed. For BDAM library access, RECFM=F or FB are allowed if LRECL is equal to BLKSIZE.
61-10	The VSAM library definition must be RRDS (Relative Record dataset).	Correct the VSAM library definition and resubmit the job.
61-11	A write operation to the library was terminated, library read-only.	No updates are allowed to this library because it is defined with read-only access. In Batch jobs, LABEL=(,,IN) on the DD statement defines a library dataset as read-only; in CICS, the library file definition is read-only. If updates to the library are to be allowed, change the JCL or CICS file definition. If no updates are allowed, inform the owners of the application to change their practices.
61-12	The specified library is RACF protected for this type of access.	The RACF/AUTHORITY does not allow the specified access to the library dataset. Determine whether to request RACF changes or inform the owners of the application to change their practices.
62-00	Format of the specified library is not compatible with Version 6 or higher versions.	Starting with Version 6, tableBASE does not support libraries that have not been converted from V5 or defined under V6. To determine library format run job DK1TLCHK. Ensure the library is in a compatible format and resubmit the job.
64-00	The INDEX parameter specified must be P, T or blank.	Correct the index parameter and resubmit the command. (Consult the <i>Programming Guide</i> for the usage of the INDEX parameter)

Return Code	Text	Meaning / Instructions
68-00	The COUNT value is too small for the DU command to dump all rows.	This Error Code may be OK if only a portion of the rows were to be dumped. If all the rows in the table were to be dumped, increase the COUNT value to the total number of rows and resubmit the command. (Consult the <i>Programming Guide</i> for the parameters of the DUmP command).
70-00	tableBASE cleanup unable to complete due to CICS shutdown processing.	tableBASE's attempts to clean up control blocks in the CICS region at termination cannot complete due to CICS termination processing.
71-00	Lock is unavailable after waiting LOCKTIMERC seconds.	A tableBASE lock was not available for use after waiting LOCKTIMERC seconds. Resubmit the command. If a longer wait time for the resource is desired, update the TBOPT LOCKTIMERC value and resubmit the command.
72-00	The table is unavailable at this time; WAIT switch is off.	The table was in use at the time of the command. Retry the command.
72-01	The table is unavailable after waiting TABLEWAITRC seconds.	With the WAIT switch on, the table enqueue was unavailable after waiting TABLEWAITRC seconds. If the wait time is to be increased, update the TBOPT TABLEWAITRC value and resubmit the command. The table enqueue is an MVS enqueue with a major name of DK1TCALL and a minor name starting with T followed by the table name and the library DSN. It is held while a table is open for write in a TSR.
73-00	An update command was issued for a table that was not open for write.	An update command was issued for a table that has not been opened for write. See descriptions for other sub-codes for detailed explanation. The command being issued requires that the table be OPEN for WRITE (OW). For correct operation, issue an OW command for the table and retry the command.
73-01	An update command was issued for a table that is open for read in a local or VTS-TSR.	An update command was issued for a table that has been opened for read in a local or VTS-TSR and this is not a single-task application i.e. MULTITASKING=Y or it is a CICS, IMS region or DB2 SP Address Space. The command being issued requires that the table be OPEN for WRITE (OW). For correct operation, issue an OW command for the table and retry the command.
73-02	An update command was issued for a table that is open for read in a VTS-TSR.	An update command was issued for a table that has been opened read in a VTS-TSR and this is a single-task application i.e. MULTITASKING=N. The command being issued requires that the table be OPEN for WRITE (OW). For correct operation, issue an OW command for the table and retry the command.

Return Code	Text	Meaning / Instructions
73-03	An update command was issued against the local TSR for a closed table	An update command was issued against the local TSR for a table that is not open in the TSR, implicit open is ON, and this is not a single-tasking application. The table will not be opened after the command completes. The command being issued requires that the table be OPEN for WRITE (OW). For correct operation, issue an OW command for the table and retry the command.
73-04	An update command was issued against a VTS-TSR for a closed table.	An update command was issued against a VTS-TSR for a table is not open in that TSR and implicit open is ON. The table will not be opened after the command completes. The command being issued requires that the table be OPEN for WRITE (OW). For correct operation, issue an OW command for the table and retry the command.
73-05	An update command was issued for a linked table.	An update command was issued for a table that is opened for read as a linked table (this appears to always be preceded by error 13-5 so may never be encountered). The command being issued requires that the table be OPEN for WRITE (OW). For correct operation, issue an OW command for the table and retry the command.
74-00	The LOCK-LATCH specified is invalid.	If a lock latch is set on a table in a TSR, any update command must match the lock latch value. Correct the LOCK-LATCH value and resubmit the command. If the lock-latch is not known, the tableBASE Administrator can perform the required command overriding the lock-latch with the Master Password.
75-00	Dynamic allocation (AL) or un-allocation (UL) failed	tableBASE command AL (Allocate library) or UL (Un-allocate library) was not successful.
75-01	TBTSLIB not allowed as parameter in AL command.	TBTSLIB is obsolete in V6. It cannot be dynamically allocated with the AL command.
75-02	DDNAME specified in AL command already in use.	Ensure the DDNAME is not already allocated and resubmit the command.
75-03	DSNAME specified in AL command not catalogued.	Ensure the DSNAME for the DDNAME is catalogued and resubmit the command. The AL command only supports catalogued library datasets
75-04	DSN enqueue conflict with another region for AL command.	Determine which region has the SYSDSN enqueue for this table and resolve the conflict. Then reissue the AL command with the appropriate SHR parameter. (Consult the <i>Programming Guide</i> for the usage of the AL command)
75-05	Error in DDNAME specified in AL command.	Correct the DDNAME syntax and resubmit the command.
75-06	Syntax error in DSNAME specified in AL command.	Correct the DSNAME syntax and resubmit the command.
75-07	SVC99 failure during execution of AL command.	An MVS error was generated. Contact your MVS systems programmer to determine why the AL command failed.

Return Code	Text	Meaning / Instructions
75-08	DDNAME specified by UL command must be allocated by an AL command.	The UL (Unallocate) command failed because tableBASE was not used to allocate the DDNAME. Deallocate the DDNAME (TSO FREE) and use the tableBASE AL command to allocate the library dataset.
75-09	DDNAME specified by UL command not known or library not open.	Ensure the DDNAME is allocated.
75-10	SVC99 failure during execution of UL command.	An MVS error was generated. Contact your MVS systems programmer to determine why the UL command failed.
76-00	OI (Open Indirect) failed: primary table not sequential or KLOC not 9.	Ensure the Open Indirect primary table has sequential organization and the key location is 9. Resubmit the command. (See the <i>Programming Guide</i> for details on Open Indirect, or the SI command for more details.)
77-00	An alternate index is defined for a table that is not TYPE=P.	Ensure the data table is defined as a pointer table (TYPE=P) and resubmit the command.
78-00	The AL command failed, the SHARE-STATUS parameter must be S or O.	Correct the SHARE-STATUS sub-parameter of the LIBRARY-ALLOC parameter. (See the <i>Programming Guide</i> for the usage of the AL command.)
79-01	The request code for the VV command is invalid.	The request code value should be 0, 1 or 2. Correct the request code and re-submit the command. (See the <i>tableBASE Administration Guide</i> for more details.)
79-02	A VTS initialized as RW cannot be switched to RO mode.	Only VTS-TSRs initialized as RO can be switched back to RO mode after being switched to RW mode.
79-03	The job name does not match the job holding the RW lock for the VTS.	A VTS can only be switched back to RO mode by the same job step that switched the VTS to RW mode.
79-04	The time stamp of the job does not match that holding the RW lock for the VTS.	A VTS can only be switched back to RO mode by the same job step that switched the VTS to RW mode.
79-05	Lock reset failed when switching VTS access to RO.	This is an internal error and cannot be fixed by the client. Please contact DataKinetics Technical Support.
79-06	Attempt to switch VTS to RW unsuccessful; VTS already locked in RW mode.	The VTS is already locked in RW mode. If the job locking it is still running, wait for it to complete. If the VTS was left in RW mode due to an abend or some other error, it can be forced back to RO mode using the VV command. See the <i>tableBASE Administration Guide</i> for details.
79-07	Lock failed when switching VTS access to RW.	This is an internal error and cannot be fixed by the client. Please contact DataKinetics Technical Support.
79-08	VTS name mismatch	This is an internal error and cannot be fixed by the client. Please contact DataKinetics Technical Support.
79-09	No pointer to TSRCNTL	This is an internal error and cannot be fixed by the client. Please contact DataKinetics Technical Support.

Return Code	Text	Meaning / Instructions
79-10	No pointer to GCE	This is an internal error and cannot be fixed by the client. Please contact DataKinetics Technical Support.
79-11	VTS-TSR not available	Ensure that the VTS is not terminating or starting up when the command is issued.
79-12	Getmain failed	Try running the entire job in a larger region.
79-13	SAF authorization could not be obtained to switch the access mode for this VTS.	CONTROL or higher SAF authorization is required. Check with your security administrator to obtain the correct authorization.
79-14	VTS in RO mode cannot be switched to RO.	The VTS is already in RO mode. No switching is required.
79-15	Program not authorized for switching VTS access mode.	In order to switch the access mode of a VTS, a program must be linked with AC(1) and in an authorized library.
80-00	Data table is not open; the use of an alternate index not allowed.	The IA command requires the data table be opened first. The OR command will dynamically open a data table.
80-01	Data table must be opened before IA (Invoke Alternate) is issued.	Open the data table before issuing the IA command.
80-02	IA (Invoke Alternate Index) may not reference a linked VTS table.	Both the alternate and the data table must be opened in the VTS-TSR or opened in the local TSR for IA to succeed. The base table cannot be a linked VTS table.
80-03	Open/Invoke Alternate Index must reference a data table.	Ensure the IA command includes the BASE-TABLE-NAME parameter and resubmit the command. The IA or OR command must reference a data table, either in BASE-TABLE-NAME or the stored definition on the library.
80-04	Alternate Index cannot be opened as a linked VTS table.	The IA command specified an alternate index name which was found in a VTS-TSR. This is not supported. Either specify the complete alternate index definition in the IA command parameters, remove the VTS-TSR from the library list (ML-list), or close the alternate index table in the VTS-TSR and resubmit the command.
81-00	The specified Alternate Index definition is not found.	If the complete alternate index definition is not specified in the IA command parameters, the alternate index definition must be found in a library in the current library list (ML-list). Correct this error and resubmit the command.
83-00	Data table INDEX sub-parameter of the DT block must be defined as P.	If the base table is specified in the IA command, it must be defined as a pointer table (TYPE=P). Redefine the base table.
84-00	Library meta data parameter is invalid	If function-ID 16 is specified, NX command should include LIBMETA-DATA structure to return the Block size information.
85-00	The command is invalid for an Alternate Index	The command being invoked is not supported for use with an alternate-index. Correct the error and resubmit the command.

Return Code	Text	Meaning / Instructions
85-01	An alternate index may not have the same name as the data table.	Correct the alternate-index parameters ensuring that the alternate index and base table names are different and resubmit the command.
85-02	DV and DW commands are not supported for Alternate index tables	The DV and DW commands are only supported for base tables since they apply to ensuing SStore commands. An alternate index is never stored, only the base table is.
86-00	Data table of an Alternate Index may not be a linked VTS table.	The table failed to open because the alternate-index is in the local TSR, and the data-table is linked to a table in a VTS-TSR. The alternate index must be opened in the same TSR as the data-table.
87-00	The KEY-COUNT must be 1 for the definition of the Alternate Index.	Only a value of 1 for the key-count is supported when defining an alternate-index. Correct the key-count value and resubmit the command.
89-00	An invalid parameter was encountered in the TBOPT file.	During initialization of tableBASE in the region, an error was discovered in the tableBASE runtime options in the TBOPT dataset. Correct the TBOPT parameter(s) and resubmit the job.
90-00	Insufficient storage available in region: increase region size.	There is insufficient main storage available for tableBASE control blocks. Increase the executing region size and resubmit the job (modify the 'REGION=' parameter for the jobstep).
91-00	I/O Error.	Determine the reason for the I/O error from the JESMSG LG. Correct the problem and resubmit the command.
91-01	I/O Error while attempting to read/write to a tableBASE library.	Determine the reason for the I/O access error to the tableBASE library. Correct the problem and resubmit the command.
91-02	I/O Error while attempting to read from the TBOPT file.	Determine the reason for the error trying to read from the TBOPT file. Once resolved, resubmit the job.
91-03	Error in I/O subtask in multitasking environment.	The subtask was unable to complete due to an I/O error when TBOPT MULTITASKING=Y. Correct the problem and resubmit the process.
92-00	Insufficient Table Space Region (TSR) size.	The TSR is too small for the tables being opened. Either enlarge the TSR or close unnecessary tables in the TSR to free up the required TSR space.
92-01	The space request is bigger than the TSR.	The table to be opened is too large for the current TSR. The TSR size will need to be increased to accommodate the table to be opened.
92-02	Insufficient free space in the TSR: close tables or enlarge TSR.	The TSR is full and no longer has enough free space available to allow the table to be opened, defined, expanded or changed. Either enlarge the TSR appropriately or close unnecessary tables to free up the required TSR space.

Return Code	Text	Meaning / Instructions
92-03	There is insufficient free contiguous space in the TSR	The TSR is full. There is no longer any contiguous memory left to accommodate the index for the table being opened, defined, expanded or changed. Either enlarge the TSR or close unnecessary tables to free up required TSR space.
92-04	There is insufficient space in the TSR to initiate the TCE	The TSR is full. There is no room to accommodate the tableBASE control block created when opening a table into the TSR. Either enlarge the TSR or close unnecessary tables to free up the required TSR space.
93-00	The RF command did not complete	Correct the error and resubmit the command.
93-01	The table must be open for read in order to be refreshed	Open the table for read (OR) or ReLease the table before submitting the RF command.
93-02	The data table or alternates were closed since the table was opened	There is a change in either the alternate or the data-table that is preventing successful completion of the RF command. CL the table and re-open it from the desired library to get the most recent version of the table into the TSR.
93-03	The ROWSIZE changed since the table was opened	RF failed because the row size for the table on the library was different compared to the table opened in the TSR. CL and reopen the table to get the most recent version of the table into the TSR.
93-04	The KEYSIZE changed since the table was opened	RF failed because the key size for the table on the library was different compared to the table opened in the TSR. CL and reopen the table to get the most recent version of the table into the TSR.
93-05	The KEYLOC changed since the table was opened	RF failed because the key-location for the table on the library was different compared to the table opened in the TSR. CL and reopen the table to get the most recent version of the table into the TSR.
93-06	The ORGANIZATION changed since the table was opened	RF failed because the organization for the table on the library was different compared to the table opened in the TSR. CL and reopen the table to get the most recent version of the table into the TSR.
93-07	The SEARCH METHOD changed since the table was opened	RF failed because the Search-Method for the table on the library was different compared to the table opened in the TSR. CL and reopen the table to get the most recent version of the table into the TSR.
95-00	The transaction was terminated during tableBASE processing.	tableBASE cleanup was invoked because a transaction was terminated while tableBASE was processing. The subcodes document what tableBASE process was executing when the transaction was terminated. This is usually related to a transaction timeout or cancellation.
95-02	tableBASE found an invalid row (item) area address	See 95-00 description. This can be caused by an error in the input parameters provided.

Return Code	Text	Meaning / Instructions
95-03	tableBASE found an invalid row address in the Table Space Region (TSR)	See 95-00 description. Send the documentation (JESMSG LG, TBDUMP, any other dumps) to tableBASE support.
95-04	tableBASE abended during an index resorting process.	See 95-00 description.
95-05	tableBASE abended during an index expansion process.	See 95-00 description.
95-06	tableBASE abended during an index entry insertion process.	See 95-00 description.
95-07	tableBASE abended during an index entry deletion process.	See 95-00 description.
95-08	tableBASE abended during an open command.	See 95-00 description.
95-09	tableBASE abended during a close command.	See 95-00 description.
97-00	The ML or LL parameter list is invalid	Correct the ML or LL parameter list and resubmit the command.
97-05	An ML list must not contain a reference to TBTSLIB.	Remove the reference to TBTSLIB in the ML-list and resubmit the command.
97-08	The ML list is invalid in VTS. It contains a reference to a VTS-TSR.	Remove the reference to the VTS-TSR in the ML-list and resubmit the command.
98-00	An internal error occurred in tableBASE Shutdown processing.	tableBASE termination (XX command) problems.
98-01	The XX command is being processed on a subordinate task.	When multi-tasking is ON, the XX command must be performed by the persistent (mother) task. This command will fail if it is submitted by sub-tasks invoked by the persistent (mother) task. Send the documentation (JESMSG LG, TBDUMP, any other dumps) to tableBASE support.
98-02	tableBASE shutdown process did not complete.	If multi-tasking and a sub-task submitted the XX command, other tasks will first issue a warning, then they will abend. This is the warning. Send the documentation (JESMSG LG, TBDUMP, any other dumps) to tableBASE support.
98-03	The subtask that shutdown tableBASE has tried to invoke tableBASE.	If multi-tasking and a sub-task submitted the XX command, no tasks are allowed to issue tableBASE commands. Send the documentation (JESMSG LG, TBDUMP, any other dumps) to tableBASE support.
1072-00	VTS access failed because the specified VTS-TSR is not available.	VTS access failed because the specified VTS-TSR is not available. Ensure that the VTS job which creates the specified VTS-TSR name is running.

Return Code	Text	Meaning / Instructions
1072-01	VTs access failed because the specified PC Server was not running.	Access to VTs-TSRs depends on a PC Server at the same or higher level than the application is running on the same LPAR.
1072-02	VTs close connection failed.	An attempt to close the VTs connection failed. Refer to message DK100303I in the JESMSGLOG for more details.
1072-03	SAF READ access required for this operation.	These operations on the VTs require READ or higher SAF authorization.
1072-04	SAF UPDATE access required for this operation.	These operations on the VTs require UPDATE or higher SAF authorization.
1072-05	VTs could not be accessed due to getmain failure for the VTs control block.	Run the job in a larger region.
1072-06	VTs could not be accessed due to corrupt VTs control block.	This is an internal error and cannot be fixed by the client. Please contact DataKinetics Technical Support: 1-613-523-5588.
1072-07	VTs has stopped running.	The VTs is terminating or is no longer running.
1072-08	VTs could not be found in this version.	The VTs name does not match any VTs in this version.
1072-09	VTs could not be found in the region list of VTs names.	The VTs does not exist in the region list of VTs names for this version.
1072-10	VTs no longer running.	The VTs was running but has been quiesced.
1072-11	VTs re-initialization failed during VTs close.	This is an internal error and cannot be fixed by the client. Please contact DataKinetics Technical Support: 1-613-523-5588.
1072-12	VTs re-initialization failed during VTs open.	This is an internal error and cannot be fixed by the client. Please contact DataKinetics Technical Support: 1-613-523-5588.
1072-13	VTs not running.	The VTs is not running.
1072-14	VTs name in region list has been zeroed.	This is an internal error and cannot be fixed by the client. Please contact DataKinetics Technical Support: 1-613-523-5588.

Return Code	Text	Meaning / Instructions
1072-15	The VTS generation number or time stamp does not match the current generation or time stamp.	The generation number or time stamp of the VTS increases if it has been switched from RO to RW and vice versa or if it has been shut down and restarted. An S in the START-TBUOW field of the COMMAND-AREA can be used to access the most current generation of the VTS. See the <i>tableBASE Programming Guide</i> for more details. For a RO VTS, this error sub-code is returned if the VTS was switched to RW and has now been switched back to RO and the START-TBUOW field does not contain an S. If S is used in the START-TBUOW field after a 1072-15 is issued, and the VTS is still locked in RW mode, a 1072-18 will be returned.
1072-16	The VTS is not available for transaction processing.	The VTS is terminating or is initializing.
1072-17	SAF CONTROL access required for this operation.	These operations on the VTS require CONTROL or higher SAF authorization.
1072-18	The VTS is currently locked in RW mode.	When locked in RW mode, the VTS is only accessible to the RW locker until the job completes. This error sub-code will be returned for the first call, after the VTS is locked in RW mode, when it is accessed by the same transaction or job that had previously accessed it; on subsequent calls, a 1072-15 will be returned.
1072-19	SAF ALTER access required for this operation.	These operations on the VTS require ALTER SAF authorization.

Table C-1: tableBASE messages and error codes

tableBASE messages

Messages issued from the tableBASE components, such as the VTS Agent, CICS, Batch and others, vary from function to function, and are listed in the table below. These messages can usually be viewed on the JESMSG LG. Any Severe Errors (S) messages should be referred to tableBASE Customer Support. Messages are in the format DK1nnnnnA where DK1 is the prefix, followed by a five digit numeric code and an error code letter. The error code letter indicates the type and severity of the error message:

- **E**—Error (usually indicates user error)
- **I**—Information
- **W**—Warning
- **A**—Action (user action is required)
- **S**—Severe (no further processing is possible)

Msg Code	Text	Meaning / Instructions
DK100049S	NO CMDAREA IN CALL TO DK1TCALL. ABENDING.	There was no command area supplied. Ensure that a command area is supplied for the tableBASE call.
DK100100W	Unexpected non-zero return code from user exit	
DK100101S	Abend Requested by user exit; see abend code	Check your exit code for the abend you requested.
DK100107W	A message was truncated at 207 characters	Information only.
DK100200S	Initialization failed	
DK100200S	tableBASE Initialization failed	
DK100201S	tableBASE PC Server unavailable	The tableBASE PC Server is not running. It must be running in order for tableBASE to initialize in batch jobs, CICS regions, IMS regions, DB2 SPAs and VTS-TSRs.
DK100202I	tableBASE V<nnn> is initializing for <customername>	Information only.
DK100203I	tableBASE <interface> license expires on <yyyy/mm/dd>	Information only.
DK100204I	Executing in Step=<stepname>	Information only.
DK100205I	Executing in JOB Step=<stepname> PROC Step=<stepname>	Information only.
DK100206S	tableBASE PC Server does not support LOAD	Ensure the latest PC Server is running.
DK100207E	DK1TX071 not loaded; System Exits disabled	Contact DataKinetics Technical Support for assistance.
DK100207E	DK1TX072 not loaded; User Exits disabled	Contact DataKinetics Technical Support for assistance.
DK100208S	Unable to locate tableBASE GCA	Ensure the latest PC Server is running.
DK100209S	tableBASE PC Server unavailable	Ensure the latest PC Server is running.
DK100210E	Module DK1TNUCL not loaded	Contact DataKinetics Technical Support for assistance.
DK100210I	Unable to locate tableBASE GCA	Information only.
DK100211I	Unable to locate tableBASE GCA.	Information only.
DK100213E	Error in tableBASE PC server	Contact DataKinetics Technical Support for assistance.

Msg Code	Text	Meaning / Instructions
DK100214E	Create CMA Named Token failed	Contact DataKinetics Technical Support for assistance.
DK100214E	CMA GETMAIN failed	Contact DataKinetics Technical Support for assistance.
DK100214I	Delete CMA Named Token Failed	Information only. If this happens multiple times, contact DataKinetics Technical Support for assistance.
DK100216S	\$DOPC GRANDE Failed	Ensure the latest PC Server is running.
DK100217S	\$DOPC Failed to Load DK1TROTB.	Check that DK1TROTB is available for an MVS LOAD.
DK100218S	\$DOPC Free LX Index Failed	Contact DataKinetics Technical Support for assistance.
DK100221E	I/O ERROR:<text>	Contact DataKinetics Technical Support for assistance.
DK100222E	QSAM ERROR: FUNCTION=<xxxxxxx>,FILE=<ffffff>	Contact DataKinetics Technical Support for assistance.
DK100224E	I/O PROBLEM: FILE=<xxxxxxx> FUNC=<ffffff> RESP=<ddd> RESP2=<ddd>	Contact DataKinetics Technical Support for assistance.
DK100225W	JOB:<jobname>,ID:<userid>,TSR:<TSRname>, CMD:<cc>,TBL:<tablename>, - WAITING <nnnnnn> SECS FOR MAPLOCK HELD BY <jobname2> <task-id>	TSR space allocation is blocked by another process accessing this TSR. <jobname2> identifies the job holding the lock and <task-id> identifies the task id of the transaction holding the lock. Cancelling the job or transaction holding the lock may free up the lock. CAUTION: The operator must determine if it is safe to cancel the job or transaction. Note: This message is usually intermittent until the tableBASE command that is blocking the access ends. If the problem persists and is happening in a VTS-TSR, contact DataKinetics Technical Support for instructions on how to alleviate the problem.
DK100226W	Please change DD TBOPTV to TBOPT	Information only.
DK100226i	Please change DD TBOPTV to TBOPT	Information only.

Msg Code	Text	Meaning / Instructions
DK100227W	JOB:<jobname>,ID:<userid>,TSR:<TSRname>, CMD:<cc>,TBL:<tablename>, - WAITING <nnnnnn> SECS FOR GLOBAL TCELOCK HELD BY <jobname2> <task-id>	Access to the TSR table list is blocked by another process. <jobname2> identifies the job holding the lock and <task-id> identifies the task id of the transaction holding the lock. Cancelling the job or transaction holding the lock may free up the lock. CAUTION: The operator must determine if it is safe to cancel the job or transaction. Note: This message is usually intermittent until the tableBASE command that is blocking the access ends. If the problem persists and is happening in a VTS-TSR, contact DataKinetics Technical Support for instructions on how to alleviate the problem.
DK100227W	JOB:<jobname>,ID:<userid>,TSR:<TSRname>, CMD:<cc>,TBL:<tablename>, - WAITING <nnnnnn> SECS FOR TCELOCK HELD BY <jobname2> <task-id>	Access to the table in the TSR is blocked by another process using the table. <jobname2> identifies the job holding the lock and <task-id> identifies the task id of the transaction holding the lock. Cancelling the job or transaction holding the lock may free up the lock. CAUTION: The operator must determine if it is safe to cancel the job or transaction. Note: This message is usually intermittent until the tableBASE command that is blocking the access ends. If the problem persists and is happening in a VTS-TSR, contact DataKinetics Technical Support for instructions on how to alleviate the problem.
DK100228W	JOB <jobname> IN <VTSname> WAITING <nnnnnnnn> SECONDS TABLE ENQ <tablename>	Job is waiting for VTS-TSR table enqueue--not received yet.
DK100229W	JOB:<jobname>,ID:<userid>,TSR:<TSRname>, CMD:<cc>,TBL:<tablename>, - WAITING <nnnnnn> SECS FOR TLBLOCK HELD BY <jobname2> <task-id> - LIB DD:<DDname>,DSN:<DSNname>	Access to the tableBASE library is blocked by another process in the same region using the library. <jobname2> identifies the job holding the lock and <task-id> identifies the task id of the transaction holding the lock. Cancelling the job or transaction holding the lock may free up the lock. CAUTION: The operator must determine if it is safe to cancel the job or transaction. Note: This message is usually intermittent until the tableBASE command that is blocking the access ends.

Msg Code	Text	Meaning / Instructions
DK100230I	TBOPT Processing	Information only.
DK100231I	<keyword>=<value>	Information only.
DK100232E	Invalid value	The value entered is invalid.
DK100233E	Missing keyword	A keyword is missing.
DK100234E	Invalid keyword	The keyword entered is invalid.
DK100235E	Value exceeds maximum allowed	The value provided exceeds the maximum allowed value.
DK100236E	Keyword not allowed in this environment	The parameter provided is not allowed for this interface.
DK100237E	Missing "="	A "=" is missing in the parameter.
DK100238E	Invalid quoted string	The string is invalid.
DK100239E	Value field too wide	The value entered is too large. Correct and try again.
DK100240E	Value less than minimum allowed	The value entered is smaller than the minimum required.
DK100241E	Missing value	There is a missing value in the parameter.
DK100242E	Keyword (or alias) previously specified	The keyword has already been specified by using another alias name.
DK100243E	Option supported only in TBOPTGEN	This parameter must be entered in the distributed source for DK1V1134 or DK1Txx34 and rebuilt. It cannot be entered in the input for the TBOPT DD card.
DK100244I	MAXNMTAB set to <nnnnnn>	Information only.
DK100245W	VTSONLY not valid with LIB entries	
DK100246W	Invalid VTS prefix for LIB<nn>=<xxxxxxxx>	
DK100247W	Empty value: LIB<xx> - LIB<yy>	
DK100248W	LIB<xx> exceeds max allowed <mm>	
DK100249W	MULTOPNX=N is not valid	
DK100250I	TBOPT Parameters: (*= Parameters not taken from TBOPT input file)	Information only.
DK100251I	<keyword>=<value>.	Information only.

Msg Code	Text	Meaning / Instructions
DK100252I	<<<<<<< TBOPT Done >>>>>>>	Information only.
DK100253I	<message>	Information only.
DK100255E	HASH_LOW_DEN_LIM must be <= 2/3 HASH_HI_DEN_LIM	Reset the density limits.
DK100256I	TSRSIZE set to LDS size of <nnnnnnn>K	Information only.
DK100258I	MAXNMTAB set by preformatted LDS TSR	Information only.
DK100260I	DK1TX066 Exit Manager Initializing	Information only.
DK100262W	Exit <xxxxxxx> not loaded and disabled	
DK100263I	Exit <xxxxxxx> is loaded and enabled	Information only.
DK100264I	DK1TX066 Exit Manager Shutdown complete	
DK100265W	Exit name <xxxxxxxxxxxxxx> not found in current release	
DK100266E	PC Service unavailable; required for VTS_TSR use	Ensure that the correct version of the tableBASE PC Server is running.
DK100270E	Insufficient storage to process strobe data; strobe suppressed.	The strobe data file has run out of space so tableBASE strobe data will be suppressed.
DK100271E	Error on CICS WRITE JOURNALNUM; Strobe logging suppressed	
DK100272E	WRITE JOURNALNUM RESP=<1234>, RESP2=<1234>	
DK100273E	Dynalloc of strobe report failed. Strobe suppressed.	
DK100274I	STROBE=0 forced for DB2 Stored Procedure	Information only.
DK100275E	Module DK1TRSTA not found.	Contact DataKinetics Technical Support for assistance.
DK100280E	QSAM ERROR: FUNCTION=<xxxxxxx>,FILE=<ffffff>	Contact DataKinetics Technical Support for assistance.
DK100281E	STROBE OUTPUT SUPPRESSED	tableBASE strobe report output (for file TBTSRPT) has been suppressed due to some error in the file, possibly a file overflow. Try increasing the disk space for the strobe report output file, TBTSRPT.
DK100282E	I/O ERROR:	Contact DataKinetics Technical Support for assistance.

Msg Code	Text	Meaning / Instructions
DK100283E	TBTSRPT spin failed. See JESlog for messages.	
DK100284E	TBTSRPT test failed. See JESlog for messages.	
DK100286E	SYSPRINT DD MISSING. Run terminated.	A SYSPRINT DD card is required in the job.
DK100292W	DEBUG TRACE requested but not started	Contact DataKinetics Technical Support for assistance.
DK100292E	DEBUG TRACE requested but not started	Contact DataKinetics Technical Support for assistance.
DK100293S	tableBASE must be initialized on QR TCB. Use tran TBDR TBINIT or PLTPL.	
DK100295A	Specify character for PC server Named Token	
DK100300S	SYSTEM FAILURE: <xxxxxxx> Code=<cccc>/<hhhhhhhh>, Reason=<rrrr>/<hhhhhhhh>	Contact DataKinetics Technical Support for assistance.
DK100300S	Call from <xxxxxxx> Offset <xxxx>	Contact DataKinetics Technical Support for assistance.
DK100300S	R0-R7 <xxxxxxx xxxxxxxx xxxxxxxx xxxxxxxx>.	Contact DataKinetics Technical Support for assistance.
DK100300S	R8-R15 <xxxxxxx xxxxxxxx xxxxxxxx xxxxxxxx>	Contact DataKinetics Technical Support for assistance.
DK100301S	Logic error; Dumpid <xxx> taken to TBDUMP.	Contact DataKinetics Technical Support for assistance.
DK100301S	Logic error dump taken. Dumpcode is LGIC. Dumpid is <nnnnnnnn>.	Contact DataKinetics Technical Support for assistance.
DK100302S	System ENQ limit exceeded. Task abended.	Contact DataKinetics Technical Support for assistance.
DK100303I	CLOS-VTS <VTSname> Failed. Return Code = <nnnn>	tableBASE internal processing to end its access to a VTS-TSR failed because the VTS-TSR was canceled before the task terminated. This message is informational only.

Msg Code	Text	Meaning / Instructions
DK100310W	tableBASE TSR <nnn>% Allocated. TSR is local TSR. Owned by Job <jobname>	The series of space management algorithms used in prior levels of V6 was changed in level 8. It now only issues a message when the TSR usage rises above a threshold and does not indicate what algorithm is being used. Setting the TSR_WARNING_FREQ = 0 will cause all the messages to be displayed once the TSR space usage threshold is crossed from below the threshold to above it or exactly equal to it. The default FREQ setting is 1 (i.e. only display one message per second, at the most). Level 8 will issue this message anytime the TSR usage rises above the value in TSR_WARNING_PCT. (The Default TSR_WARNING_PCT is 85.) Level 8 warnings start when the usage is exactly the TSR_WARNING_PCT value or higher. For more details refer to the TBOPT parameter settings in the <i>tableBASE Installation Guide</i>.
DK100311W	tableBASE TSR <nnn>% Allocated. VTS-TSR name is <vtsname>. Owned by Job <jobname>.	Same as for message DK100310W, but applies to VTS-TSR.
DK100330I	DDNAME <ddname> is not a tableBASE Release 6 format library.	Information only.
DK100331W	LIBRARY FREE SPACE MAP INCONSISTENT WITH LEFT BLOCK COUNT	
DK100332I	Caching requested for DDname <ddname>. Caching not permitted on this library. Processing with no cache.	
DK100340W	Old form ML mixed with use of extended ML	
DK100362E	ABEND DURING UPDATE PROCESS RC=95 SC=<xx>	
DK100364W	LCKCOUNT=nnnnnnnn RESET FOR TBL:tblname CMD: xx (This is the job name id if running in CICS) TSR: vts-name or LOCAL_DS	A lock has been reset for a table. The LCKCOUNT is mainly used for debugging.
DK100365W	LCKCOUNT=nnnnnnnn FOR TBL:tbl-name TSR:vtsname or LOCAL_DS HELD BY CMD:xx (This is the job name id if running in CICS)	A reset was bypassed because the write lock was held by another command or task. The LCKCOUNT is mainly used for debugging.

Msg Code	Text	Meaning / Instructions
DK100400E	tableBASE User Error Abend <xxx>, Reason=<dddd/hhhh>	For more information, call DataKinetics Technical Support.
DK100400E	tableBASE User Error Abend <xxx>, Command=<xx>,<tablename>	For more information, call DataKinetics Technical Support.
DK100400E	tableBASE User Error Abend <xxx>, DK1TCALL Initializing	For more information, call DataKinetics Technical Support.
DK100400E	tableBASE User Error Abend <xxx>, Reason=<dddd/hhhh>, Command=<xx>,<tablename>	For more information, call DataKinetics Technical Support.
DK100400E	tableBASE User Error Abend <xxx>, Reason=<dddd/hhhh>, DK1TCALL Initializing	Abend code G200 indicates a CICS environment; 0200 indicates otherwise. For more information, call DataKinetics Technical Support.
DK100411E	TSR Create failed: Failed by site IEFUSI	Contact DataKinetics Technical Support for assistance.
DK100412E	TSR Create failed: MVS resource shortage	Contact DataKinetics Technical Support for assistance.
DK100413W	SNAP SUPPRESSED; GETMEM SHORTAGE	Contact DataKinetics Technical Support for assistance.
DK100414E	TSR Create failed: LDS Smaller than TSR SIZE	
DK100415E	TSRACCESS=RO using empty LDS	
DK100416I	TSR successfully mapped to LDS	Information only.
DK100417I	TSR successfully saved on LDS	Information only.
DK100471E	BDAM ERROR: File=<ffffff>, Function=<xxxxxxxx>	Contact DataKinetics Technical Support for assistance.
DK100472E	VSAM ERROR: File=<ffffff>, Function=<xxxxxxxx>, Reg15=<rrrrr>, Error Code=<xxxx>	Contact DataKinetics Technical Support for assistance.
DK100472E	I/O ERROR: <text>	Contact DataKinetics Technical Support for assistance.
DK100472E	<xxx> ERROR: File=<ffffff>, Function=<xxxxxxxx>, Resp=<rrrrr>, Resp2=<pppppp> (Resp2=VSAM Response)	Contact DataKinetics Technical Support for assistance.
DK100473E	VSAM ERROR: File=<ffffff>, Request=ATTR2, Retcode=<xxx>, Reason=<yyy>, R15=<zzz> DSNAME=<dsname>	Contact DataKinetics Technical Support for assistance.

Msg Code	Text	Meaning / Instructions
DK100476E	xxAM ERROR: FUNCTION=<xxxxxxx>,FILE=<ffffff>	Contact DataKinetics Technical Support for assistance.
DK100477E	I/O ERROR: <text>	Contact DataKinetics Technical Support for assistance.
DK100477E	VSAM ERROR: FUNCTION=<xxxxxxx>, REG15=<rrrrr>,ERROR CODE=<xxxxx>, FILE=<ffffff>	Contact DataKinetics Technical Support for assistance.
DK100490E	ERROR ON TBDEBUG FILE, REASON=<xxx>, DEBUG DISABLED	Contact DataKinetics Technical Support for assistance.
DK100494E	Persistent Task ended before end of step	Contact DataKinetics Technical Support for assistance.
DK100495I	IN ROTBVTSD	Information only.
DK100496E	Multitasking requires TBOPT Multitasking=Y be set	Reset MULTITASKING parameter in TBOPT
DK100497E	failed to delete ESTAE	Contact DataKinetics Technical Support for assistance.
DK100501S	EXEC CICS FAILURE: <xxxxxxx> Resp=<cccc/hhhhhhhh>, Resp2=<cccc/ hhhhhhh>	Contact DataKinetics Technical Support for assistance.
DK100511E	TSR Create failed: Failed by site IEFUSI	Contact DataKinetics Technical Support for assistance.
DK100512E	TSR Create failed: MVS resource shortage	Contact DataKinetics Technical Support for assistance.
DK100513W	SNAP suppressed; GETMEM shortage	Contact DataKinetics Technical Support for assistance.
DK100530I	No storage for QTCWA; QCT function not used.	Information only.
DK100531I	Module DK1TRSTA not found; QCT function not used.	Information only.
DK100532I	Attaching QCT x	Information only.
DK100533I	QCT X Detached	Information only.
DK100550I	tableBASE <version> - Resource manager initializing	Information only.
DK100551I	tableBASE Resource manager initialized	Information only.
DK100552E	tableBASE Initialization Error	Contact DataKinetics Technical Support for assistance.

Msg Code	Text	Meaning / Instructions
DK100553I	tableBASE Initialization Failed	Information only.
DK100554I	tableBASE Starting tblBASE	Information only.
DK100555I	tableBASE tblBASE started	Information only.
DK100556I	tableBASE Deactivating	Information only.
DK100557I	tableBASE Deactivated	Information only.
DK100558E	tableBASE Deactivation Error	Contact DataKinetics Technical Support for assistance.
DK100559I	tableBASE Deactivation Failed	Information only.
DK100560I	tableBASE Restarting	Information only.
DK100561I	tableBASE Verifying	Information only.
DK100562I	tableBASE Verify completed	Information only.
DK100563I	tableBASE - Error Info: Func(<functionname>/<subfunctionname>) Resp(<responsecode>) Resp2(<responsecode>)	This is error information received by tableBASE when requesting a CICS service function. Refer to the CICS manuals as required.
DK100564I	DK1TCIN used in PLT takes one of TBINIT (default) or INIT.	Information only.
DK100565I	Successful <xxxx> INIT must be done before <xxxx> TBCALL	Information only.
DK100566I	Verification failure	Information only.
DK100567E	The tableBASE PC Server must be running for tableBASE to initialize.	The tableBASE PC Server is not running. It must be running in order for tableBASE to initialize in CICS regions.
DK100568E	Program DK1TCIN must be defined as CONCURRENCY(QUASIRENT)	Define DK1TCIN as per message and try again.
DK100569E	Module DK1CBASE must be current to determine version.	Contact DataKinetics Technical Support for assistance.
DK100570I	<xxxx> takes TBINIT or TERM as a parameter	Information only.
DK100571I	Phase in of <xxxxxxxx> not allowed by CICS; current in-storage copy used.	Information only.
DK100572I	<xxxxxxxx> LP=<xxxxxxxx> EP=<xxxxxxxx> AM=<xx> RC=<xxxxxxxx> UC=<xxxxxxxx>	Information only.
DK100573I	Module <xxxxxxxx> not loaded	Information only.

Msg Code	Text	Meaning / Instructions
DK100574I	Module <xxxxxxx> not found.	Information only.
DK100575I	EX=<xxxxxxx> EN=<xxxxxxx> STST=<xxx> SHST=<xxx> TAST=<xxx> TALN=<xxxx> GALN=<xxxx>	Information only.
DK100576I	<xxxxxxxxxxxxxxxx/xxxxxx> OK	Information only.
DK100577I	<xxxxxxx> LP=<xxxxxxx> EP=<xxxxxxx> LEN=<xxxxxxx> AM=<xxx> KEY=<x>	Information only.
DK100578I	tableBASE - Shutdown Completed.	Information only.
DK100579I	tableBASE - Shutdown Aborted: Internal Resource Anchor Block (RCABLK) not found.	Information only. Contact DataKinetics Technical Support for further information about this message.
DK100580I	tableBASE - Shutdown Aborted: TRUE TBASECRM not enabled.	Information only. Contact DataKinetics Technical Support for further information about this message.
DK100590E	TBL BASE ERROR DEBUG WTD, REASON=<xxx>, DEBUG DISABLED	Contact DataKinetics Technical Support for assistance.
DK100595I	Trans error dump taken. Dumpcode is DKL1. Dumpid is <xxxxxxx>	Information only.
DK100596E	CICS Assign failed (Resp/Resp2=<nnnnn/ nnnn>); dump suppressed	Contact DataKinetics Technical Support for assistance.
DK100597E	CICS DUMP TRANSACTION failed (Resp/ Resp2=<nnnnn/nnnn>). Dump for userid <xxxxxxx> not taken.	Contact DataKinetics Technical Support for assistance.
DK100600I	tableBASE V<vrm> VTS <xxxxxxx> initialized	Information only.
DK100601E	GCA cannot be located; VTS stopping	The tableBASE PC Server is not running. It must be running before any VTS-TSRs can be brought up.
DK100602E	VTS Agent incompatible with GCA structure	Contact DataKinetics Technical Support for assistance.
DK100603E	Duplicate VTS attempting to start; auto-stopping	Another VTS with the same name is already running.
DK100604E	Program <xxxxxxx> is not APF authorized	The program must be APF authorized to run this task.
DK100605E	V<vrm> Environment not ready; VTS stopping	Neither the tableBASE PC Server nor compat VTS Group Manager are running. Either one must be running before any VTS-TSRs can be brought up.

Msg Code	Text	Meaning / Instructions
DK100606E	Incompatible DK1VBASE; VTS stopping	The version of DK1VBASE is not compatible with this release.
DK100608E	Not authorized to start VTS resource: <vtsresourcename>	This userid is not authorized to start the VTS resource named due to lack of RACF authorization. A minimum of CONTROL access is required to start a RACF-protected VTS.
DK100610E	Error in <xxxxxxx> macro; RC/ REASON=<xxx/yyyy>	Contact DataKinetics Technical Support for assistance.
DK100620E	VTS is running in a Non-level 8 GCA	Contact DataKinetics Technical Support for assistance.
DK100629E	Invalid non-level 8 GCA	Contact DataKinetics Technical Support for assistance.
DK100630I	Console interface is available	Information only.
DK100631E	Invalid command; command ignored	The command supplied is invalid.
DK100632W	'REFRESH not supported in tableBASE <vv> VTS	The RF command is not supported in VTS.
DK100633I	SHUTDOWN	Information only.
DK100634I	STOP command received, V<vrm> VTS <xxxxxxx> terminating	Information only.
DK100636W	switching TSRACCESS to RW, VTS <xxxxxxx>: <nnnn> users may be affected	Warning message.
DK100637W	TSRACCESS=RW forced RO for locking- <xxxxxxx>	Warning that the TSR access mode has been force switched to RO.
DK100638W	switching TSRACCESS to RO, VTS <xxxxxxx>: <nnnn> users may be affected	Warning message.
DK100639E	Not authorized to stop VTS resource: <vtsresourcename>	This userid is not authorized to stop the VTS resource named due to lack of RACF authorization. A minimum of CONTROL access is required to stop a RACF-protected VTS.
DK100701S	tableBASE PC Server unavailable, required for license options	The tableBASE PC Server must be running.
DK100702I	TSR size reset to LDS size	Applicable only for VTS Manager.
DK100703S	Invalid LDSTSR dataset	Applicable only for VTS Manager.
DK100704W	TSRACCESS forced to RO: LDSTSR DISP=SHR	Due to the disposition of the LDS dataset, the TSR access mode is forced to RO. Applicable only for VTS Manager.

Msg Code	Text	Meaning / Instructions
DK100705I	MAXNMTAB ignored for TSRACCESS=RO	Applicable only for VTS Manager.
DK100706W	SYSPLEX=Y incompatible with TSRACCESS=RW; TSRACCESS set to RO	Applicable only for VTS Manager.
DK100707E	TPVM incompatible with COMPAT VTS Agent	Applicable only for VTS Manager.
DK100708I	StrobeMethod 3 allowed only for VTS	Information only.
DK100709E	StrobeMethod 3 required for VTS with non-zero TSR_KEY-PROTECT	When a VTS is hardware key-protected, TBOPT option STROBEMETHOD=3 must be used.
DK100710W	TSRACCESS forced to RO: TSR_KEY_PROTECT set	Warning message. A hardware key-protected VTS can only be started up in read-only mode.
DK100711W	Macro <xxxxxxx> non-zero return code <nnnnn>	Contact DataKinetics Technical Support for assistance.
DK100712W	IFADREG token zero at entry to IFAEDDRG	Contact DataKinetics Technical Support for assistance.
DK100800I	tableBASE PC server V<vrm> available	The tableBASE PC Server is now running. It must be running in order for tableBASE to initialize in batch jobs, CICS regions, IMS regions, DB2 SPAs and VTS-TSRs.
DK100802E	PC server <xxxxxxx> terminating; see message DK100803I for reason.	Contact DataKinetics Technical Support for assistance.
DK100803I	<variable length message text>	Information only.
DK100804S	System must support 64-bit architecture	Starting with Version 7.0.3, tableBASE requires z/OS running on 64-bit hardware which supports the General Instructions Extension Facility (GIEF), the Long Displacement Facility, the Extended-immediate Facility and the Execute- extensions Facility. These are provided with the Z10-EC processor or higher.
DK100805W	License checking error <nnn> for <pppppppp>	The license checker encountered an error. Contact DataKinetics Technical Support for assistance.
DK100806W	Error <nnn>-<rrr> LOADIng DK1TLICK; <pppppppp> not licensed	Error loading the license checker. Contact DataKinetics Technical Support for assistance.
DK100807W	Error <nnn> ATTACHIng DK1TLICK for <pppppppp>	Error attaching the license checker. Contact DataKinetics Technical Support for assistance.
DK100811I	Invalid command; ignored: <ccccccc>	Information only.
DK100812I	STOP command received	Information only.
DK100820W	DK1TPC01/DK1TLICK interface mismatch <nnn/nnn>	Contact DataKinetics Technical Support for assistance.

Msg Code	Text	Meaning / Instructions
DK100821W	PC Server <vvvv> does not support <pppppppp>	Contact DataKinetics Technical Support for assistance.
DK100822W	<pppppppp> not supported by licensing routine	Contact DataKinetics Technical Support for assistance.
DK100823W	DK1TLICK not authorized to enter key 0	Contact DataKinetics Technical Support for assistance.
DK100824W	PC Server <vvvv> not found	Ensure that the PC Server for this release is running.
DK100825W	Licensing subroutine returned RC=<rrrr>	Contact DataKinetics Technical Support for assistance.
DK100826W	<pppppppp> license expired <yyyy-mm-dd>; in grace period	The license for the named product has expired and it is now running in the grace period. It will stop functioning after the grace period.
DK100827W	<pppppppp> license expired <yyyy-mm-dd>	The license for the named product has expired and the product has stopped functioning.
DK100828W	<pppppppp> license will expire <yyyy-mm-dd>	The named product is in the warning period for its license and will no longer be licensed on the specified date.
DK100829W	<pppppppp> license expires today <yyyy-mm-dd>	Just after midnight today, the named product will no longer be licensed. If there is a grace period, the product will continue to function during that period.
DK100830W	<pppppppp> license module DK1xBASE missing	The license module specified for the named product is missing.
DK100831W	<pppppppp> DK1xBASE <vvvv> mismatch <vvvv>	The license module specified for the named product is for a different version.
DK100832W	<pppppppp> DK1xBASE return code <rrrr>	Contact DataKinetics Technical Support for assistance.
DK100833I	<pppppppp> licensed to <customer>	Information only.
DK100834I	<pppppppp> is not licensed	Information only.
DK100844S	Customer Anchor Table DKL slot invalid: RC=<r>.	Contact DataKinetics Technical Support for assistance.
DK100847S	cannot start, too many DKL products	Contact DataKinetics Technical Support for assistance.
DK100848S	TPM may be running, run DK1PTERM	Applicable only for VTS Manager.
DK100849S	Replacing corrupted GCA	Contact DataKinetics Technical Support for assistance.

Msg Code	Text	Meaning / Instructions
DK100850I	deleting NT	Information only.
DK100851I	NT deleted	Information only.
DK100852I	freeing PC code	Information only.
DK100853I	PC code freed	Information only.
DK100854I	Freeing GCA	Information only.
DK100855I	Freeing GCAX	Information only.
DK100856I	GCA purged	Information only.
DK100857I	Replacing corrupted Dummy TPM block	Information only.
DK100858S	PC Server unable to obtain storage	Contact DataKinetics Technical Support for assistance.
DK100859I	Completing Dummy TPM/TPV blocks	Information only.
DK100870W	LXFRE Bad Return Code	Contact DataKinetics Technical Support for assistance.
DK100871S	\$DOPC GETMAIN Failed	Contact DataKinetics Technical Support for assistance.
DK100872E	LXFREE Getmain Failed	Contact DataKinetics Technical Support for assistance.
DK100873W	LXFREE Freemain Failed	Contact DataKinetics Technical Support for assistance.
DK100874S	\$DOPC SCHEDIRB Macro Failed	Contact DataKinetics Technical Support for assistance.
DK100875S	\$DOPC DK1TPC11 Routine Failed	Contact DataKinetics Technical Support for assistance.
DK100876S	\$DOPC LOADROTB Service Failed	Contact DataKinetics Technical Support for assistance.
DK100877I	DK1TVROT Named Token Failed	Information only.
DK100886W	Warning - GCA611 is already taken over.	Trying to take over the PVTE/GCA V611 when we already own it.
DK100887W	Warning - GCA611 is already released.	Trying to release the PVTE/GCA V611 when we already released it
DK100888I	GCA is taken over by job <xxxxxxx>	Information only. The PC Server job-name took over the PVTE/GCA V611

Msg Code	Text	Meaning / Instructions
DK100889I	GCA611 is being released by job <xxxxxxx>	Information only. The PC Server job-name released the PVTE/GCA V611
DK100890E	GCA611 is held by job <xxxxxxx>	An error - the PVTE/GCA is held by another job identified by job-name.
DK100891S	DK1PGCA6 failed - see message DK100803I for reason.	A major error occurred in the DK1PGCA6 load module.
DK100900S	Not licensed for VTS Manager options	Your installation is not licensed for this feature. Contact DataKinetics Technical Support for further information.
DK100901S	Not licensed for VTS Gate options	Your installation is not licensed for this feature. Contact DataKinetics Technical Support for further information.
DK100902S	Not licensed for DEFAULT_TSR options	Your installation is not licensed for this feature. Contact DataKinetics Technical Support for further information.
DK100990S	Incompatible ROOT and NUCLEUS versions	Contact DataKinetics Technical Support for assistance.
DK100991E	Parms module not loaded	Contact DataKinetics Technical Support for assistance.
DK100993E	Parms module corrupted	Contact DataKinetics Technical Support for assistance.

Table C-2: Messages and error codes for VTS Agent, CICS, Batch, etc.

Note: Abnormal terminations between 99 and 1000 or over 1099, are tableBASE errors and should be brought to the immediate attention of your tableBASE Administrator.

tablesONLINE/CICS error messages

The following table contains information on the tablesONLINE/CICS error messages that can be encountered.

MSG #	Text	Meaning / Instructions
TB-2000 I	The program TBDRIVC terminated successfully.	Information only.
TB-2010 E	Transfer Error: Transfer transaction TRIN was not found.	TRIN is delivered as part of the TBOL product. Consult the tableBASE installation instructions.
TB-2011 E	Transfer Error: Invalid transaction.	Target transaction is not defined properly to CICS.

MSG #	Text	Meaning / Instructions
TB-2012 E	Transfer Error: User not authorized to execute the transaction.	Target transaction is not defined properly to CICS.
TB-2013 E	Transfer Error: Transaction is currently disabled.	Target transaction is not defined properly to CICS.
TB-2014 E	Transfer Error: TWA size too small.	Target transaction is not defined properly to CICS.
TB-2015 E	Transfer Error: Cannot retrieve PCT entry (Lock/Busy condition).	Target transaction is not defined properly to CICS.
TB-2016 E	Transfer Error: Target program not found or is currently disabled.	Target transaction is not defined properly to CICS.
TB-5000 E	Unexpected tableBASE error XXXX occurred while processing 'XX'.	Refer to the tableBASE error return codes for more information about errors processing command 'XX'. If problem not explained, contact DataKinetics tableBASE support.
TB-5001 I	Enter Down command to view additional rows.	Information only.
TB-5002 I	Enter Up command to view additional rows.	Information only.
TB-5003 W	Enter Help command for a display of # lines of messages.	Information only.
TB-5004 I	Enter Up or Down command to view additional rows.	Information only.
TB-5005 W	You only have the authority to browse. Edit capability disabled.	Information only.
TB-5006 W	You do not have the authorization to perform this function.	Information only. Contact tableBASE administrator.
TB-5007 E	Indirect help could not be done because table could not be opened.	See <i>tablesONLINE/CICS User's Guide</i> for further information.
TB-5008 E	Indirect help could not be done because key could not be found.	See <i>tablesONLINE/CICS User's Guide</i> for further information.
TB-5009 E	Command 'XXXXXXXXXXXXXXXXXXXXXXX' is not supported for hash organization.	See <i>tablesONLINE/CICS User's Guide</i> for further information.
TB-5010 E	Duplicate keys allowed accepts only a Y or N response.	
TB-5011 E	The generation number supplied is incorrect.	Occurs when the generation does not exist.
TB-5012 E	Library: XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX not on XXXXXXXXXX table. Contact your system administrator.	Contact your tableBASE administrator.

MSG #	Text	Meaning / Instructions
TB-5013 E	Library selection failed. You are not authorized to modify 'xxxxxxx'. Please contact the tableBASE system administrator.	Contact your tableBASE administrator.
TB-5014 E	Table open failed, you may not access tableBASE library: xxxxxxxx.	
TB-5015 E	Indirect help not followed. Indirect points to indirect help.	Only one level of indirect help is supported.
TB-5016 E	Invalid COMMAND: 'XXXXXXXXXXXXXXXXXXXXX' for multi-user update mode.	See <i>tablesONLINE/CICS User's Guide</i> for further information.
TB-5017 W	There are no other windows established, enter 'NEW' to create another Window.	Or you can enter NW on the command line.
TB-5018 W	This View restricts you to browse only. Edit capability disabled.	Contact your tableBASE administrator.
TB-5019 E	Table 'xxxxxxx ' is accessed from a Read Only VTS region. Try Browse.	You accessed a table residing on VTS as a result of the VTS name appearing in the tableBASE library list. If you want to EDIT this table, your profile on the TBOLACT table must be changed. Contact your tableBASE administrator.
TB-5020 W	The last field has been searched. To repeat from start press FINDFLD.	The fields are searched from left to right or top to bottom starting from the cursor position. This message appears when the end of the fields have been reached.
TB-5021 W	The field name starting with ' ' is not found.	
TB-5023 E	View key field only accepts an 'Y' or 'N' entry.	
TB-5029 I	Changes to the view 'xxxxxxx' have been cancelled.	Information only.
TB-5030 E	The count entered is outside the table. Count of last row is xxxxxxxx.	TBOL returns the last available row.
TB-5031 I	Changes to table 'xxxxxxx' have been cancelled.	Information only.
TB-5032 I	The previous row examined has been deleted.	Information only.
TB-5033 I	The row has been added to the table.	Information only.
TB-5034 I	A row with this key is already on the table. The row has been added.	Information only.
TB-5035 I	The row has been updated.	Information only.
TB-5036 I	A row with this key is already on the table. The row has been updated.	Information only.

MSG #	Text	Meaning / Instructions
TB-5037 E	A row with this key is already on the table. Update rejected.	
TB-5038 W	The count entered is outside the table. You are positioned at bottom.	
TB-5039 E	The key entered does not match any row key on the table.	Re-enter command as required.
TB-5040 E	The exit program '*****' could not be loaded.	Exits from TBOL are defined in one of two places: in the View of the table being edited/browsed or in the TBOLACT table if processing commands.
TB-5041 W	The count entered is outside the table. You are positioned at the top.	Re-enter command as required.
TB-5042 W	The first available row is:	Warning occurs when attempting to locate a row before the previously displayed rows and a user exit program suppresses the display of rows according to the exit logic.
TB-5043 W	The last available row is:	Warning occurs when attempting to locate a row after the previously displayed rows and a user exit program suppresses the display of rows according to the exit logic.
TB-5044 I	Table already in use. Multiple user access in effect.	Information only. Multi user access is a View attribute.
TB-5045 W	This row is currently in use by 'xxxxxxx' in the 'aaaa' application.	When attempting to edit a row in use by another user xxxxxxxx.
TB-5046 I	Table accessed by multiple users has been closed.	Information only. Multi user access is a View attribute.
TB-5047 I	Multiple user access in effect. Others may be browsing or editing.	Information only. Multi user access is a View attribute.
TB-5049 E	Cannot close table 'xxxxxxx'. The table is open for write.	Another TBOL Window, or an unrelated process is holding this table opened for write. Contact your system administrator.
TB-5050 E	The password supplied is invalid.	
TB-5051 W	Changes have been made to table 'XXXXXXXX'. To confirm that you indeed wish to cancel the changes: Select 'CANCEL' To avoid having all the changes lost: Select 'ENTER' or 'END'.	Information only. Follow message instructions as required.

MSG #	Text	Meaning / Instructions
TB-5052 W	Changes have been made to table 'XXXXXXXX'. To confirm that you indeed wish to save the changes and create a new generation: Select 'ENTER' or 'END' To cancel the changes that have been made: Select 'CANCEL'. To go back to editing without saving or canceling: Enter 'RESHOW'.	Information only. Follow message instructions as required.
TB-5053 E	Changes have been made to table 'XXXXXXXX'. Changing the library, table, generation or password can only be done after work on the present generation is complete. In order to save the changes that have been made: Select 'ENTER' or 'END'. To cancel the changes that have been made: Select 'CANCEL'.	Information only. Follow message instructions as required.
TB-5054 W	A request to delete row(s) from the table has been entered. To confirm that you indeed wish to delete the rows: Select 'ENTER' or 'END'. To cancel the delete request: Select 'CANCEL'.	Information only. Follow message instructions as required.
TB-5055 I	Table 'XXXXXXXX' has been updated.	Information only.
TB-5056 I	View 'XXXXXXXX' has been updated.	Information only.
TB-5057 W	You have made changes to table 'XXXXXXXX' which may have been updated by other users. Several users may be updating this table simultaneously. The 'CANCEL' command may not have any effect because another user may have saved the table. To undo any changes you made... In order to save your changes and any changes made by other users: Select 'ENTER' or 'END'.	Information only. Follow message instructions as required. Multi user access is a View attribute.

MSG #	Text	Meaning / Instructions
TB-5058 W	You have made changes to table 'XXXXXXXX' which may have been updated by other users. In order to save your changes and any changes made by other users: Select 'ENTER' or 'END' To avoid saving the changes and cycling another generation: Select 'CANCEL'. To go back to editing without saving: Enter 'RESHOW'.	Information only. Follow message instructions as required. Multi user access is a View attribute.
TB-5059 W	You have made changes to table 'XXXXXXXX' which may be being browsed by other users. Changing the library, table, generation or password can only be done after work on the present generation is complete. In order to save your changes and any changes made by other users: Select 'ENTER' or 'END'. To avoid saving the changes and cycling another generation: Select 'CANCEL'.	Information only. Follow message instructions as required. Multi user access is a View attribute.
TB-5060 W	You are not authorized to use the M2M command. Contact DataKinetics tableBASE support.	Contact DataKinetics tableBASE support.
TB-5061 W	A range of nnnnnn row(s) has been selected for deletion. Because some rows may be suppressed from view, the actual number to be deleted may be less. To confirm that you indeed wish to delete the rows: Select 'ENTER' or 'END'. To cancel the delete request: Select 'CANCEL'.	Information only. Follow message instructions as required.
TB-5062 E	Listing the contents of the VTS attached to xxxxxxxx is not supported.	Contact DataKinetics tableBASE support.
TB-5063 E	Internal error '9999' creating a directory listing.	Contact DataKinetics tableBASE support.
TB-5064 E	You are trying to create a duplicate primary key. Update rejected. Primary key fields are hi-lighted with a different color.	View Attribute
TB-5065 E	Delete enqueue error. Non-existing key: xxxxxxxxxxxxxxxxxxxxxx	Record the Key. It may help pinpoint an issue.
TB-5066 E	The row with primary key: xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx has been deleted or moved by another user.	Record the Key. It may help pinpoint an issue.
TB-5500 W	Exit program called with invalid indicators:	User Exit error. Contact tableBASE administrator.

MSG #	Text	Meaning / Instructions
TB-5501 W	Field name is spaces.	
TB-5502 W	To indicate a key field code Y. Otherwise leave blank or code N. If the field is a Dynamic View Suffix, code S. If it is also a key field as well as the Dynamic View Suffix, code B. If this is a View for Editing a table using an Alternate Index and the Primary Key of the table is in a different location, code P. If a Primary Key Field and an Alternate Index Key Field overlap, code X.	
TB-5503 E	Invalid display or internal table format.	With Cursor on field, press Help (PF1).
TB-5504 W	Display length must be xxxx and has been substituted.	
TB-5505 E	Display length must be in the range xxxx through xxxx.	With Cursor on field, press Help (PF1).
TB-5506 W	Internal table length must be xxxx and has been substituted.	
TB-5507 E	Internal table length must be in the range xxxx through xxxxx.	With Cursor on field, press Help (PF1).
TB-5508 E	Invalid edit length code 'X' on TBOLHKFM table. Valid values are: blank, D, E, H, N, P, Q, 0.	
TB-5509 E	Invalid attribute. Pick N, C, F, D, Q, S, P, p, M, m, V, v, or blank.	With Cursor on field, press Help (PF1).
TB-5510 E	Invalid feature, must be D, N, Y, ':' or blank.	With Cursor on field, press Help (PF1).
TB-5511 E	Invalid action code. It must be Y, N, R, E, I, D, C, U, or blank.	With Cursor on field, press Help (PF1).
TB-5512 W	Program name 'xxxxxxxx' does not exist.	
TB-5513 E	Invalid direction, must be I for inbound to display or O for outbound.	With Cursor on field, press Help (PF1).
TB-5514 E	Invalid timing, must be B for before system action or A for after.	With Cursor on field, press Help (PF1).
TB-5515 E	Key fields must be contiguous. Check field =====>	
TB-5516 E	Display length must equal internal table length.	With Cursor on field, press Help (PF1).
TB-5517 E	Display length should equal table length or table length + 1.	With Cursor on field, press Help (PF1).
TB-5518 E	Display length should equal (2 * table length). Subtract 1 if no sign.	With Cursor on field, press Help (PF1).
TB-5519 E	Display length must equal 2 * table length.	With Cursor on field, press Help (PF1).
TB-5520 E	Display length should equal table length + 1 or table length + 2.	With Cursor on field, press Help (PF1).

MSG #	Text	Meaning / Instructions
TB-5521 E	Display length should equal (2 * table length). Add 1 if signed.	With Cursor on field, press Help (PF1).
TB-5522 W	Exit program entered without setting exit indicators.	User Exit error. Contact tableBASE administrator.
TB-5523 E	You may not move a field to a position before a key field.	In display mode of View edit the keys are always grouped first.
TB-5524 E	You cannot add new physical fields in display mode.	
TB-5525 E	You may not move a key field.	
TB-5526 E	You cannot delete a field in display mode.	
TB-5527 E	The number of key fields must be between 1 and 50 inclusive.	
TB-5528 E	The total length of the key fields must not be greater than 256.	
TB-5529 W	Application specific help table 'XXXXXXXX' not found on any library.	
TB-5530 E	Invalid field location for field: View may be corrupt - contact your technical support person.	If the error cannot be explained, contact DataKinetics tableBASE support.
TB-5531 E	The Data Table is currently in use.	Not used in Version 6 TBOL
TB-5532 W	Data/Index 'xxxxxxx' requires RSZ, KSZ, KLOC: nnnnnnnnnnnnnnnnnnn, in order to conform to the View just defined. Change the values of the row size, key size & location of the physical Data Table to conform to the View. If this is a View for an Alternate Index the warning above may not apply. Please note the KEY SIZE, and KEY LOCATION so that these values can be used to verify the Alternate Index Definition - Option 6 on the Define Table and View menu.	
TB-5533 E	Data Table 'xxxxxxx' is in use - try again later.	Occurs when attempting to edit a table.
TB-5534 E	You cannot change the keys or suffix in display mode.	For some applications, a different display order is possible. Contact DataKinetics tableBASE support.
TB-5535 E	You cannot change fields to be key fields while in display mode.	Go to DEFINE VIEW on the "Define Table and View" menu to change key fields.
TB-5536 E	The total length of the key fields must be greater than zero.	
TB-5537 E	Number of delimiters must be equal the display length.	
TB-5538 E	The display mask must include a valid sign symbol.	This applies to numeric Display Formats N and 0 through 9.
TB-5539 E	The display mask must have 'X' decimal places.	This has to match the number of decimals places of the Display Format

MSG #	Text	Meaning / Instructions
TB-5540 E	The number of delimiters must be one less than the display length.	
TB-5541 E	The trigger field name supplied is not valid. A trigger field must be defined with an action code of Y, R or E.	Verify the spelling of the Trigger field name. It must be a field already defined in the View.
TB-5542 E	Invalid decimal point encountered at position 'XX'.	
TB-5543 W	Edit pattern only allowed for display format of 'X'.	
TB-5544 W	Invalid or incomplete pattern delimiter encountered at position 'XX'.	
TB-5545 E	Number of edit pattern symbols must be equal to display length.	
TB-5546 E	View 'XXXXXXXX' is in use by another application.	Someone else is editing the View.
TB-5547 E	The field format and/or length differs from source key field.	
TB-5548 E	Source field format and/or length differs from importation field.	
TB-5549 E	Source field name required for duplication action.	
TB-5550 E	The source field name entered already has a duplication action defined	If you want multiple duplicates, always take the value from a single source field.
TB-5551 E	Source field format and/or length differs from duplication field.	
TB-5552 E	Create date or update date action codes apply only to date fields.	
TB-5553 E	The source field is not a key field as required for action Y, R or E.	
TB-5554 E	The source field specified is not the entire key of the corresponding source Data Table.	
TB-5555 E	Source field name not defined in the view.	
TB-5556 E	Importation action requires a previously defined field with an action code of Y, R or E.	
TB-5557 E	Internal view alternate table access failed.	Contact DataKinetics tableBASE support.
TB-5558 E	Source field must be the entire key of the Source Table.	
TB-5559 E	A FIELD FORMAT of P, F, or H cannot be used as a dynamic View suffix.	Only formats that define displayable characters are supported.

MSG #	Text	Meaning / Instructions
TB-5560 E	The source field name entered must not be the same as the target field	
TB-5561 A	A row on view 'xxxxxxx' cannot be found. Internal error. Abort task.	Contact DataKinetics tableBASE support.
TB-5562 E	tableBASE error 'XXXX' occurred creating internal table 'RSXXXXXX'.	You may have too small a TSR (tableSPACE Region).
TB-5563 E	tableBASE error 'XXXX' occurred storing internal table 'RSXXXXXX'.	Target library is Read Only.
TB-5564 E	The total length of all fields: 99999 exceeds 32,763 bytes.	32,763 bytes is the maximum row size.
TB-5565 W	The attribute V or v cannot be used with display format Y. The attribute has been set to a space.	
TB-5566 E	The target field must be 8 bytes to receive a DDNAME from the list or it must be 44 bytes to receive a DATASET NAME.	
TB-5567 E	The target field must be 8 bytes to receive a table or view name.	
TB-5568 E	More than one dynamic suffix. Check field =====>	
TB-5569 E	The total length of the Primary key fields must be less than 257.	The sum of the lengths of all fields comprising a key must be 256 or less.
TB-5570 E	The FIELD LENGTH/FORMAT must be '8' 'X' or 'U' respectively for the insert USERID action code 'B'	
TB-5571 E	'xxxxxxx' is not a valid TBOL view.	Verify view, and re-issue.
TB-5600 W	Exit called for a table it doesn't know about. Table is xxxxxxxx.	
TB-5601 E	You cannot delete your own session. Action ignored.	Information only.
TB-5700 W	Utility processing is bypassed as the current operating mode is 'X'.	Contact DataKinetics tableBASE support. Very likely the XXXXMENU table has errors.
TB-5701 W	Invalid action. Use S to select a utility.	Information only. Follow message instructions as required.
TB-5702 W	Table 'xxxxxxx' has been copied as 'xxxxxxx'.	Information only.
TB-5703 W	Table 'xxxxxxx' has been renamed to 'xxxxxxx'.	Information only.
TB-5704 W	Table 'xxxxxxx' has been deleted.	Information only.
TB-5705 W	View 'xxxxxxx' has been copied as 'xxxxxxx'.	Information only.

MSG #	Text	Meaning / Instructions
TB-5706 W	View 'xxxxxxx' has been renamed to 'xxxxxxx'.	Information only.
TB-5707 W	View 'xxxxxxx' has been deleted.	Information only.
TB-5708 W	A generation of table 'xxxxxxx' has been deleted.	Information only.
TB-5709 W	User profile has been updated.	Information only.
TB-5710 W	Application table 'xxxxxxx' has been copied as 'xxxxxxx'.	Information only.
TB-5711 E	Function cannot be performed: Table 'xxxxxxx' not found.	
TB-5712 E	Utility failed: Table 'xxxxxxx' is currently in use.	Try again later, or contact tableBASE administrator.
TB-5713 E	Utility failed: Table 'xxxxxxx' already exists on library.	
TB-5714 E	Utility: View 'xxxxxxx' parameters not currently defined.	
TB-5715 E	Utility failed: view 'xxxxxxx' already exists on the library.	
TB-5716 E	Utility cannot be performed: View 'xxxxxxx' not found.	
TB-5717 E	Utility failed: Prefix required to generate a new set of tables.	
TB-5718 E	Utility failed: Invalid tableBASE library. DDNAME: xxxxxxxx.	
TB-5719 W	DDNAME xxxxxxxx not on 'xxxxxxx' - contact your system administrator.	Contact your tableBASE administrator.
TB-5720 E	Utility failed: Invalid password supplied.	Try again, or contact tableBASE administrator.
TB-5721 W	Table 'xxxxxxx' has been defined.	Information only.
TB-5722 W	Table 'xxxxxxx' definition has been updated.	Information only.
TB-5723 W	Supplemental information of view 'XXXXXXXX' has been updated.	Information only.
TB-5724 E	Reducing row size may result in data loss. Press execute to continue.	Information only. Follow message instructions as required.
TB-5725 E	The passwords for table 'xxxxxxx' have been changed.	Information only.
TB-5726 W	Alternate Index 'xxxxxxx' has been defined for table 'xxxxxxx'.	Information only.
TB-5727 E	Table 'xxxxxxx' from a different library is currently in use.	
TB-5728 E	Illegal To prefix 'xxxx'.	

MSG #	Text	Meaning / Instructions
TB-5729 E	Blank table name. This is a required entry.	
TB-5730 E	Invalid generation.	
TB-5731 E	Function not performed - tableBASE library XXXXXXXXX is full.	
TB-5732 E	tableBASE library must be entered.	
TB-5733 E	The RSZ, KSZ & KLOC have been updated to conform to an updated view. If these values are not required and the original values of the data are to be used, select ENTER again immediately upon return. Press EXECUTE to change the Data Table to conform to the VIEW. This could result in data loss if you are reducing the row size.	Information only. Follow message instructions as required.
TB-5734 W	Table is not an alternate. Fields filled in from the Data Table.	
TB-5735 E	Table 'xxxxxxx' is an alternate for Data Table 'xxxxxxx'.	
TB-5736 W	The RSZ, KSZ & KLOC have been filled in from the view.	
TB-5737 W	View 'xxxxxxx' could not be found and was not renamed.	
TB-5738 W	Alternate Index 'xxxxxxx' has been updated for table 'xxxxxxx'.	
TB-5739 E	New name cannot be the same as old name.	
TB-5740 E	This operation is not supported for tables from/to the VTS server.	
TB-5741 E	Alternates should not be updated with this option. Use option 6.	The name used was that of an Alternate Index.
TB-5742 W	Utilities function error. There is no processing logic for row ID 'X'.	Contact DataKinetics tableBASE support.
TB-5743 E	Utility failed: Invalid password supplied for target.	
TB-5744 E	The SYSIN parameter starting with ' xxxxxxxx . . . ' cannot be found.	There is an error in the TBOLJCLx template table. Cannot identify the JCL statement. Contact DataKinetics tableBASE support.
TB-5745 E	The JES spool data set could not be opened, reason code '9999'.	Contact your CICS system administrator.
TB-5746 E	The JES spool data set could not be closed, reason code '9999'.	Contact your CICS system administrator.

MSG #	Text	Meaning / Instructions
TB-5747 E	The JES spool data set could not be written to, reason code '9999'.	Contact your CICS system administrator.
TB-5748 E	The JCL template table, 'xxxxxxx' does not exist. Please re-enter.	Re-enter command as required.
TB-5749 E	The variable ' ' in JCL statement 00000000 was not replaced.	There is an error in the TBOLJCLx template table. The template JCL statement is missing a search for value. Contact DataKinetics tableBASE support.
TB-5750 E	The JCL statement starting with '//xxxxxxx . . . ' cannot be found.	There is an error in the TBOLJCLx template table.
TB-5751 I	Job 'xxxxxxx' has been submitted to generate copybook 'mmmmmmmm'.	Information only.
TB-5752 E	The generated JCL from template 'xxxxxxx' for table/view 'tvvtvvtv' has been submitted with an error. Please examine the JES queue to ensure that the generated JCL contain the correct program name and that the keyword substitutions have been accurately performed for this function.	Examine the JES queue and determine the error before re-submission. You may need to change the accounting information.
TB-5753 I	Job 'xxxxxxx' has been submitted to print table 'ttttttt'.	Information only.
TB-5754 E	Alternate definition failed: Table 'xxxxxxx' already exists. Use Execute with 'UPDATE' on command line to replace alternate table.	Information only. Follow message instructions as required.
TB-5755 E	Copy cannot be performed: Table 'xxxxxxx' exists on target library.	
TB-5756 E	Update of alternate failed: Alternate table 'xxxxxxx' does not exist.	
TB-5757 E	Copy cannot be performed: View 'xxxxxxx' exists on target library.	
TB-5758 I	Job 'xxxxxxx' has been submitted to print View definition 'vvvvvvvv'.	Information only.
TB-5759 I	The alternate definition name 'xxxxxxx' is unavailable - (not found).	Information only.
TB-5760 E	Enter the name of a table that contains the restructuring rules.	Follow message instructions as required. This table would have been saved from a prior visit to the Data Restructuring Utility.
TB-5761 E	Re-enter a different name.	Re-enter a different name as required.

MSG #	Text	Meaning / Instructions
TB-5762 E	The restructuring table, 'xxxxxxx' does not exist. Please re-enter.	Re-enter restructuring table name as required.
TB-5763 E	Enter Y or N to indicate the generations you want to restructure.	Information only. Follow message instructions as required.
TB-5764 I	The restructuring table, XXXXXXXX is not a valid restructuring table.	Information only. The restructuring table contains information to restructure tables. It can only be processed by TBOL.
TB-5765 E	Restructuring tbl: XXXXXXXX requires RSZ/KSZ/KLOC:nnnnnnnnnnnnnnnnnnnn. The Data Table that has been entered cannot be restructured with this restructuring rules table.	The Restructuring table does not match the basic information of the data table that you are attempting to restructure.
TB-5766 E	Generation (-x) of the data has RSZ/KSZ/KLOC:nnnnnnnnnnnnnnnnnnnn. This does not conform to the latest generation. Restructuring aborted. Restructuring is possible with the most recent generation. Set the All Generations flag to N so only the most recent is restructured.	Older generations of the data table do not have the same layout as the latest generation from which the restructuring data was taken from.
TB-5767 I	N generation(s) of the table 'XXXXXXXX' has been restructured.	Information only.
TB-5768 I	The restructuring rules are saved in the table 'xxxxxxx'.	Information only.
TB-5769 E	If the restructuring rules are to be saved for future use with other tables, enter a restructuring table name.	Information only. Follow message instructions as required. The name should be recorded so it can be used at a later time or deleted if it is never used again.
TB-5770 I	The data is already restructured. Use END or CANCEL to exit utility.	Information only.
TB-5771 E	Although you have elected not to restructure the Data Table(s) at this time, it is essential to enter a name to save the restructuring table for use at a later date. If this is not a requirement, enter CANCEL to avoid a repeat of this message.	Information only. Follow message instructions as required.
TB-5772 E	FLD: xxxxxxxxxxxxxxxxxxxxxx Error: One of the rows from the existing table cannot be converted.	
TB-5773 E	Data Table 'XXXXXXXX' cannot be stored, library 'XXXXXXXX' is full. Free up some space on the target library, return to this option (7), and try again.	Open up another Window and delete an unused table to free up some space

MSG #	Text	Meaning / Instructions
TB-5774	The name 'XXXXXXXX' entered is either an alternative index or a paged table and cannot be restructured directly. For alternate indexes restructure the underlying base Data Table. For a paged table, change it to an in-memory table using the utility DK1TEXEC.	For alternate indexes: Restructure the underlying base Data Table. For a paged table: Change it to an in-memory table using the utility DK1TEXEC.
TB-5775 E	The variable '' in SYSIN statement 00000000 was not replaced.	There is an error in the TBOLJCLx template table. The template JCL statement is unable to replace a search for value. Contact DataKinetics tableBASE support.
TB-5777 W	The Multi-User table update option requires unique primary keys. If this is the VIEW for a table (not an Alternate Index), set the DUPLICATE KEY IND to 'N'. Press HELP for more details.	Follow message instructions as required.
TB-5778 E	The Multi-User table update option requires unique primary keys.	With Multi-User updates, TBOL cannot process tables that have duplicate primary keys.
	If this is the VIEW for an Alternate Index, set the DUPLICATE PRIME KEYS to 'N' when MULTIPLE USER UPDATE is set to 'M'.	With Multi-User updates, TBOL cannot process tables that have duplicate primary keys.
TB-5801 E	## - Action Error:	The ## identifies the field that has the Error: xxx.. Often pressing help gives you the list of values to pick from.
TB-5802 E	## - Import Error:	The ## identifies the field that has the Error: xxx..
TB-5803 E	## - Duplication Error:	The ## identifies the field that has the Error: xxx..
TB-5804 E	## - Mandatory Error: An EOF erase does not constitute a valid key stroke. Spaces are valid.	The ## identifies the field that has the Error: xxx..

MSG #	Text	Meaning / Instructions
TB-5805 E	## - Pattern: 'XXXXXXXX' does accept data as entered. Refer to legend below: Z - alphabetic; A - alphabetic or blank; 9 - numeric; I - numeric or blank; Y - alphabetic or numeric; X - alphabetic, numeric or blank; B - blank; C - any character (no validation); Literals - any set of characters between a pair of (!) marks. Use Help (F1) in Define View Option for more details.	The ## identifies the field that has the Error: xxx..
TB-5806 E	## - Verification Error:	The ## identifies the field that has the Error: xxx..
TB-5811 E	The source View 'xxxxxxx' not found.	The Source View is required to locate the import data.
TB-5812 E	The source Data table 'xxxxxxx' not found.	Data importation is not possible.
TB-5813 E	The source table 'xxxxxxx' is password protected.	Data importation is not possible.
TB-5814 E	tableBASE error 'xxxxx' occurred opening Source Table 'xxxxxxx'.	Refer to the tableBASE error return codes for more information.
TB-5815 E	Source Table, 'xxxxxxx' must be in descending order for x' action.	Data importation is not possible.
TB-5816 E	The length of field 'xxxxxxxxxxxxxxxx' and Source Table 'xxxxxxx' key field length do not match.	Data importation is not possible.
TB-5817 E	Source field 'xxxxxxxxxxxxxxxx' not found in table 'xxxxxxx'.	Data importation is not possible.
TB-5818 E	Source field 'xxxxxxxxxxxxxxxx' in table 'xxxxxxx' is not a key.	Data importation is not possible.
TB-5819 E	Source field 'xxxxxxxxxxxxxxxx' in source view 'xxxxxxx' has a different length than the field defined in the view.	Data importation is not possible.
TB-5820 E	Source field 'xxxxxxxxxxxxxxxx' in source view 'xxxxxxx' is not the entire key of the corresponding source Data Table.	Data importation is not possible.
TB-5821 E	Source field 'xxxxxxxxxxxxxxxx' in source view 'xxxxxxx' has a different format than the field defined in the view.	Data importation is not possible.
TB-5822 E	tableBASE error 'xxxx' occurred creating internal table 'SPxxxxx'.	Refer to the tableBASE error return codes for more information. If problem not explained, contact DataKinetics tableBASE support.

MSG #	Text	Meaning / Instructions
TB-5823 E	tableBASE error 'xxx' occurred emptying internal table 'ECxxxxxx'.	Refer to the tableBASE error return codes for more information. If problem not explained, contact DataKinetics tableBASE support.
TB-5824 E	tableBASE error 'xxxx' occurred creating internal table 'ECxxxxxx'.	Refer to the tableBASE error return codes for more information. If problem not explained, contact DataKinetics tableBASE support.
TB-5825 E	tableBASE error 'xxxx' occurred deleting internal table 'ECxxxxxx'.	Refer to the tableBASE error return codes for more information. If problem not explained, contact DataKinetics tableBASE support.
TB-5826 E	tableBASE error 'xxxx' occurred deleting internal table 'SPxxxxxx'.	Refer to the tableBASE error return codes for more information. If problem not explained, contact DataKinetics tableBASE support.
TB-5827 E	Upper range 'xxxxxxxxxxxxxxxx' in source view 'xxxxxxxx' has a different length than the lower range field.	Data importation is not possible.
TB-5828 E	Upper range 'xxxxxxxxxxxxxxxx' in source xxxxxxxx' has a different format than the lower range field.	Data importation is not possible. Fix the format mismatch first.
TB-5829 E	Lower range 'xxxxxxxxxxxxxxxx' in source view 'xxxxxxxx' has no corresponding upper range field for action R.	Data importation is not possible. Fix the format mismatch first.
TB-5830 E	Internal get storage failed for data importation processing.	Data importation is not possible.
TB-5832 E	View data importation pointers corrupted.	Contact DataKinetics tableBASE support.
TB-5833 E	View data duplication pointers corrupted.	Contact DataKinetics tableBASE support.
TB-5834 E	Upper range in source Data Table 'xxxxxxxx' is too short or missing.	Data importation is not possible.
TB-5835 E	Initial value does not conform to the display format or edit pattern.	
TB-5836 E	Highest bound does not conform to the display format or edit pattern.	
TB-5837 E	Lowest bound does not conform to the display format or edit pattern.	
TB-5838 E	The Lowest bound value is greater than Highest bound value.	
TB-5839 E	Both the Highest and Lowest bound values must be specified.	

MSG #	Text	Meaning / Instructions
TB-5840 E	The Initial value is not between the Highest and Lowest bounds.	
TB-5841 E	Initial value has longer length than the display length.	
TB-5842 E	Lowest bound value has longer length than the display length.	
TB-5843 E	Highest bound value has longer length than the display length.	
TB-5844 E	## - Value Lower than bound: ' '. Please enter a value greater than the lower bound indicated.	The ## identifies the field that has the Error. Enter a value greater than the lower bound indicated.
TB-5845 E	## - Value Greater than bound: 'uuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuu'. Please enter a value lower than the upper bound indicated.	The ## identifies the field that has the Error. Enter a value lower than the upper bound indicated.
TB-5897 E	QIDERR error: command code 'xxxx' response code 'xxxxxxxxxxxx'. A CICS related operational resource is unavailable. The table directory listing was aborted. Please try HELP again (PF1).	The table directory listing was aborted. Please try HELP again (PF1).
TB-5898 E	QIDERR error: command code 'xxxx' response code 'xxxxxxxxxxxx'. A CICS related operational resource is unavailable. You cannot continue to edit the table. Table rows may have been updated. In order not to lose any changes, You must exit the table editor and save any changes. After exiting this message screen, press END (PF3) from the Identify Table/Row screen.	Exit the table editor and save any changes. After exiting this message screen, press END (PF3) from the Identify Table/Row screen.
TB-5899 E	QIDERR error: command code 'xxxx' response code 'xxxxxxxxxxxx'. A CICS related operational resource is unavailable. You cannot continue to edit/browse the table. After exiting this message screen, press END (PF3) from the Identify Table/Row screen and try editing or browsing the table at a later time.	Try editing or browsing the table at a later time.
TB-5900 W	Help message unavailable, message 'xxxxxxx' not on table 'xxxxxxx'.	
TB-5901 E	Unable to open table 'xxxxxxx', reason code 9999.	Refer to the tableBASE error return codes for more information. If problem not explained, contact DataKinetics tableBASE support.
TB-5902 E	Unable to invoke alternate table 'xxxxxxx', reason code 9999.	Refer to the tableBASE error return codes for more information. If problem not explained, contact DataKinetics tableBASE support.
TB-5903 W	Help information for table 'xxxxxxx' is not found on table 'xxxxxxx'	

MSG #	Text	Meaning / Instructions
TB-5904 W	DESC: 'XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX' not on 'XXXXXXX'	This could occur when a DESCription with key has been deleted.
TB-5905 A	Error creating table 'XXXXXXXX', reason code 9999. Job aborted.	Refer to the tableBASE error return codes for more information. If problem not explained, contact DataKinetics tableBASE support.
TB-5906 W	Tutorial data 'XXXXXXXXXXXXXXXXXXXXXXX' not found on table 'XXXXXXX'.	
TB-5907 E	This PF key has not been defined on the PFK table.	
TB-5908 W	This application does not have items defined on menu table 'XXXXXXX'.	Refer to the <i>tablesONLINE/CICS User's Guide</i> for creation of Menu Items.
TB-5909 W	Help data 'XXXXXXXXXXXXXXXXXXXXXXX' not on table 'XXXXXXX'	Refer to the <i>tablesONLINE/CICS User's Guide</i> for creation of Menu Items.
TB-5910 E	Transfer program 'XXXXXXX' of menu option 'xxxx' not found/disabled.	Refer to the <i>tablesONLINE/CICS User's Guide</i> for creation of Menu Items.
TB-5911 E	Blank transfer parameters for menu option 'XXXX'	Refer to the <i>tablesONLINE/CICS User's Guide</i> for creation of Menu Items.
TB-5912 E	Invalid COMMAND or OPTION: 'XXXXXXXXXXXXXXXXXXXXXXX', please re-enter.	Re-enter command as required.
TB-5913 E	Transfer option 'XXXXXXXXXXXXXXX' is not found on menu table 'XXXXXXX'.	Refer to the <i>tablesONLINE/CICS User's Guide</i> for creation of Menu Items.
TB-5914 W	The value entered for this command should be numeric.	Re-enter command as required.
TB-5915 W	Additional tutorial information is not found on table 'XXXXXXX'.	
TB-5916 E	The keys are protected. The row cannot be deleted in this mode.	
TB-5917 E	Command or option 'XXXXXXXXXXXXXXX' not supported for browse.	See <i>tablesONLINE/CICS User's Guide</i> for further information.
TB-5918 E	Command or option 'XXXXXXXXXXXXXXX' unsupported for hash tables.	See <i>tablesONLINE/CICS User's Guide</i> for further information.
TB-5919 E	There are no rows on the table.	
TB-5920 E	View 'XXXXXXX' cannot be found on the library searched.	
TB-5921 E	View 'XXXXXXX' is empty.	

MSG #	Text	Meaning / Instructions
TB-5922 E	Unable to open view 'xxxxxxx', reason code 9999.	Refer to the tableBASE error return codes for more information. If problem not explained, contact DataKinetics tableBASE support.
TB-5923 W	Dynamic open of view 'xxxxxxx' failed, reason code 9999.	Refer to the tableBASE error return codes for more information. If problem not explained, contact DataKinetics tableBASE support.
TB-5924 W	Dynamic open failed. View 'xxxxxxx' not found.	
TB-5925 E	tablesONLINE system table 'xxxxxxx' is not on any library searched.	
TB-5926 E	Unable to open tablesONLINE system table 'xxxxxxx', reason code 9999.	Refer to the tableBASE error return codes for more information. If problem not explained, contact DataKinetics tableBASE support.
TB-5927 E	tablesONLINE system table 'xxxxxxx' is empty.	
TB-5928 E	tablesONLINE system table 'xxxxxxx' is in use by another application.	
TB-5929 W	Dynamic open failed. View 'xxxxxxx' is empty.	
TB-5930 E	## - Conversion Error:	The ## identifies the field that has the Error: xxx..
TB-5931 E	## - Conversion Error:	The ## identifies the field that has the Error: xxx..
TB-5932 E	Key field ## - Conversion Error:	The ## identifies the field that has the Error: xxx..
TB-5933 E	View 'xxxxxxx' is in use by another application.	
TB-5934 E	Data Table 'xxxxxxx' cannot be stored, library 'xxxxxxx' is full. In order not to lose the work just completed, enter another Window and free up some space by deleting unnecessary tables. Once this is done: Select 'ENTER' or 'END'. To cancel the changes that have been made: Select 'CANCEL'.	Follow message instructions as required.
TB-5935 E	Data Table 'xxxxxxx' is in use by another application. Try browse.	

MSG #	Text	Meaning / Instructions
TB-5936 E	View 'xxxxxxx' cannot be stored, library 'xxxxxxx' is full. In order not to loose the work just completed, enter another Window and free up some space by deleting unnecessary tables. Once this is done: Select 'ENTER' or 'END'. To cancel the changes that have been made: Select 'CANCEL'.	Follow message instructions as required.
TB-5937 E	View defines row size of 00000 which differs from table RSZ of 00000.	In order to use this View, you must make the Key size, Row size and Key location match the Data table.
TB-5938 E	View defines key location 00000 which differs from tbl location 00000.	In order to use this View, you must make the Key size, Row size and Key location match the Data table.
TB-5939 E	View defines key size of 0000 which differs from table key of 0000.	In order to use this View, you must make the Key size, Row size and Key location match the Data table.
TB-5940 E	Data Table 'xxxxxxx' is not on any library searched.	
TB-5941 E	Default Data Table 'xxxxxxx' is not on any library searched.	
TB-5942 E	You cannot create 'xxxxxxx'. The table already exists.	
TB-5943 W	Dynamic open failed. View row size 00000 does not match table 00000.	In order to use this View, you must make the Key size, Row size and Key location match the Data table.
TB-5944 E	View 'xxxxxxx' cannot be found. Enter NEW command to create view.	Enter NEW on the command line to create an View.
TB-5945 W	The table in memory is not the latest generation.	This occurs if another User or application has the table Open for Read in the TSR and you are trying to open the latest copy from the library, that was updated by another application since the table was opened in this TSR.
TB-5946 E	The table in TSR is not the latest generation. Update access denied.	This occurs if another user or application has the table Open for Read in the TSR and you are trying to open the latest copy from the library, that was updated by another application since the table was opened in this TSR. Close the table in the TSR and try the edit again. The tableBASE administrator can go to option 2 on the Administrators Menu to find the User that has the table open.
TB-5947 E	Table is in use by another application. Changes cannot be cancelled.	

MSG #	Text	Meaning / Instructions
TB-5948 W	Help information for menu 'xxxxxxx' is not found on table 'xxxxxxx'.	
TB-5950 E	The keys are protected. A new row cannot be inserted in this mode.	
TB-5951 A	This application is now continued at another terminal.	
TB-5952 A	This application is now suspended by the tableBASE administrator.	Contact your tableBASE administrator.
TB-5953 E	View defines Primary Key at 00000 which differs table location 00000.	In order to use this View, you must make the Key size, Row size and Key location match the Data table.
TB-5954 E	View defines Primary Key size of 0000 which differs table key of 0000.	In order to use this View, you must make the Key size, Row size and Key location match the Data table.
TB-5957 W	Line command 'xxxx' at row count '' requires target A or B.	
TB-5958 W	Line command 'xxxx' at row count '' requires action command.	
TB-5959 E	A hash table organization cannot support a block row delete.	
TB-5960 E	The line commands are conflicting. Row location(s): , .	Examine the conflicting location and place the line command appropriately.
TB-5961 I	Block command 'xxxx' at row count '' requires target A or B.	Information only.
TB-5962 I	Block command 'xxxx' at row count '' is incomplete.	Information only.
TB-5963 E	Invalid line command 'xxxx', please re-enter.	Re-enter command as required. Typically this occurs if you are using an update line command such as U, N or D while in BROWSE mode.
TB-5964 E	A sequential or hash organization of 'X' cannot support a row move operation. The organization must be U or R. This can be accomplished with a change definition command.	Use option 3 on the Table Define menu to change the table to be user ordered.
TB-5965 I	The number of rows deleted is	Information only.
TB-5966 I	The number of rows moved is x to location y.	Information only.
TB-5967 I	Line commands have been cleared.	Information only.
TB-5968 W	Line commands following the selected row have been deleted.	Under certain organizations multiple row line commands are not supported.

MSG #	Text	Meaning / Instructions
TB-5969 E	A table organization of 'X' cannot support a NEW nn command (insert at location nn operation). The organization must be 'U' (User ordered, Pointer or True) or the organization must be 'R' (Random ordered, True which is changed to User ordered by tablesONLINE). NOTE: A Random Pointer table is edited using a sequential index. In order to change the table organization, use option 3 on the Define Table menu.	
TB-5970 W	Table is empty. Enter data here. To leave table empty select CANCEL.	
TB-5971 E	Line command 'XXXX' invalid in multiple user table access. If you are trying to Delete a row, select the row for Update using the 'U' line command. Once the row is selected, type DELETE in the command line. (DEL or DE will also work.)	Because other users may be editing items, selecting an item first ensures that you have complete control first before deleting the selected row.
TB-5972 E	Multiple line commands are not supported for hash tables.	The rows are physically in random order.
TB-5973 E	tableBASE error 'xxxx' occurred storing table/view 'xxxxxxx'.	Refer to the tableBASE error return codes for more information. If problem not explained, contact DataKinetics tableBASE support.
TB-5974 E	Operation suppressed. You are not licensed to extend use of tableBASE.	
TB-5975 E	Editing of VTS server tables is not supported. Try browse.	You accessed a table residing on VTS that was accessed as a result of the VTS name appearing in the tableBASE library list as determined by your profile on the TBOLACT table. Contact your tableBASE administrator.
TB-5980 E	The table could not be opened because the Data Table 'xxxxxxx' is not a pointer table. The Data Table must be defined as a pointer table in order to be accessed as an alternate indexed table.	This occurs when the underlying data table is not defined as a pointer table. Change the INDEX attribute of the Data table to be pointer.
TB-5985 A	Error updating table 'xxxxxxx', reason code 9999.	Refer to the tableBASE error return codes for more information. If problem not explained, contact DataKinetics tableBASE support.
TB-5986 A	Table TBOLACT must be upgraded to Release 6. See Installation Guide.	See Installation Guide for information on updating tablesONLINE/CICS.
TB-5987 A	Table TBOLPROF must be upgraded to Release 6. See Installation Guide.	See Installation Guide for information on updating tablesONLINE/CICS.

MSG #	Text	Meaning / Instructions
TB-5988 A	Table TBOLCNST does not contain the default row, SIGNON terminated.	An important 'DEFAULT' row is missing from the TBOLCNST table. This row must exist to launch TBOL for this CICS region. Contact DataKinetics tableBASE support.
TB-5989 A	Table TBOLMRO does not contain the default row, SIGNON terminated.	An important 'DEFAULT' row is missing from the TBOLMRO table. This row must exist to launch TBOL for this CICS region. Contact DataKinetics tableBASE support.
TB-5990 A	An error occurred when attempting to terminate your session.	Contact your tableBASE administrator.
TB-5991 A	Unable to open table 'xxxxxxx', reason code 9999. Job Aborted.	Refer to the tableBASE error return codes for more information. If problem not explained, contact DataKinetics tableBASE support.
TB-5992 A	Transaction work area 99999999 too small. It should be 99999999. Abort	Contact your CICS system administrator.
TB-5993 A	Your application is not defined to the system, SIGNON terminated.	Contact your tableBASE administrator.
TB-5994 A	Descript. '****NO DESCRIPTION ' not on tbl 'xxxxxxx'	An important '**** NO DESCRIPTION' row is missing from the XXXXDESC table. This row must exist to launch TBOL. Contact DataKinetics tableBASE support.
TB-5995 A	Error creating table 'xxxxxxx', reason code 9999. Job aborted.	Refer to the tableBASE error return codes for more information. If problem not explained, contact DataKinetics tableBASE support.
TB-5996 A	tablesONLINE incomplete environment.	Check for further tablesONLINE or tableBASE messages.
TB-5997 A	Message '5900 W Help message unavailable, . . . ' not on tbl 'xxxxxxx'	An important '5900 W Help' row is missing from the XXXXMSGs table. This row must exist to launch TBOL. Contact DataKinetics tableBASE support.
TB-5998 W	Unable to save user profile. tBASE COND:	This can happen if the tableBASE administrator has the TBOLPROF table open for edit
TB-5999 A	CICS error: command code 'xx' response code 'xxxxxx'. Job aborted.	Contact your CICS system administrator.
TB-6000 A	The VTS region 'xxxx' needs to be started before launching TBOL.	Contact your tableBASE administrator.

MSG #	Text	Meaning / Instructions
TB-6001 E	The VTS region 'xxxx' needs to be started before 'ttttttt' can be opened. Another approach is to edit the TBSYSVTS table that specifies in which VTS region to open the table.	Contact your tableBASE administrator.
TB-9000 E	Exit program 'xxxxxxx' called with invalid indicators: xxxxxxxxxxxx.	Contact your tableBASE administrator.
TB-9001 W	Table xxxxxxxx called exit with indicators: xxxxxxxxxxxx in test mode.	Contact your tableBASE administrator.
TB-9999 A	CICS error: command code 'xx' response code 'xxxxxx'. Job aborted.	Contact your CICS system administrator.
TBA1000 E	There is no audit control record for this table. Contact administrator	Contact your tableBASE administrator.

Table C-3: tablesONLINE / CICS messages and error codes

tablesONLINE/ISPF error messages

The following table contains information on the tablesONLINE/ISPF error messages that can be cross-referenced by the message dataset. Note that the message numbers do not appear on-screen. Use your Browser FIND feature to search for the message text.

Msg #	Short format	Long format	Meaning
MSG000	TBSYSLB ALLOCATION ERROR	tablesONLINE/ISPF SYSTEM LIBRARY CANNOT BE ALLOCATED; SEE TBASE ADMIN	Contact system administrator.
MSG001	LIBRARY NOT ALLOCATED	LIBRARY IS ALREADY ALLOCATED OR IS NOT CATALOGUED OR DOES NOT EXIST	
MSG002	ERROR &TBERR HELP=TBERROR	REFER TO tableBASE REFERENCE CARD OR PRESS HELP KEY AGAIN	
MSG003	INVALID GENERATON NUMBER	THE GENERATION NUMBER IS NOT LEFT JUSTIFIED NUMERIC (SIGNED)	
MSG005	TABLE IS EMPTY	TABLE CONTAINS NO ROWS, BROWSE IS SUPPRESSED	
MSG006	NEW TABLE DEFINED	A NEW TABLE HAS BEEN DEFINED IN STORAGE	Information only.
MSG007	GENERATION DELETED	THE REQUESTED GENERATION OF TABLE &TABLE HAS BEEN DELETED	Information only.

Msg #	Short format	Long format	Meaning
MSG008	DEFINITION CHANGED	THE DEFINITION OF TABLE &TABLE HAS BEEN CHANGED SUCCESSFULLY	Information only.
MSG009	TABLE SAVED	THE TABLE HAS BEEN SUCCESSFULLY SAVED	Information only.
MSG010	FIELD NAME IS MISSING	THE FIELD NAME IS MISSING FROM THE LINE IDENTIFIED	
MSG011	DISPLAY LEN SUBSTITUTED	THE DISPLAY LENGTH HAS BEEN SUBSTITUTED BY THE VALUE IN THE EDIT TABLE	Information only.
MSG012	DUPLICATE ROW MODIFIED	THIS ROW HAS A KEY THAT IS IDENTICAL TO ANOTHER ONE ON THE TABLE	
MSG013	TABLE IS NOT NEW	A TABLE WITH THIS NAME ALREADY EXISTS ON THIS LIBRARY	
MSG014	KEY FIELDS NOT TOGETHER	THE KEY FIELDS ARE SEPARATED BY NON-KEY FIELDS	
MSG015	TABLE LENGTH SUBSTITUTED	THE TABLE LENGTH HAS BEEN SUBSTITUTED BY THE VALUE IN THE EDIT TABLE	Information only.
MSG016	ROW SIZE CONFLICT	SUM OF FIELD LENGTHS IS NOT EQUAL TO THE DEFINITION ROW SIZE	
MSG017	KEY SIZE CONFLICT	SUM OF KEY FIELD LENGTHS IS NOT EQUAL TO THE DEFINITION KEY SIZE	
MSG018	KEY LOCATION CONFLICT	FIRST KEY FIELD LOCATION IS NOT EQUAL TO THE DEFINITION KEY LOCATION	
MSG019	NO KEY FIELD SPECIFIED	NO KEY FIELD WAS IDENTIFIED AND AT LEAST ONE MUST BE SPECIFIED	
MSG020	TOO MANY KEYS DEFINED	TOO MANY FIELDS HAVE BEEN IDENTIFIED AS KEY FIELDS, MAX 10 FIELDS	
MSG022	VIEW NOT FOUND	A VIEW DEFINITION MUST EXIST FOR THIS FUNCTION.	
MSG023	VIEW EMPTY	NO FIELDS HAVE BEEN DEFINED FOR THIS TABLE.	

Msg #	Short format	Long format	Meaning
MSG024	TOO MANY FIELDS DEFINED	TOO MANY FIELDS HAVE BEEN DEFINED FOR THIS TABLE, SYSTEM LIMITATION	
MSG025	ROW NOT FOUND	THE REQUESTED ROW WAS NOT FOUND IN THE TABLE	
MSG026	SCROLL VALUE INVALID	VALUE MUST BE U (UP), D (DOWN) OR E (EXECUTE) FOLLOWED BY A NUMBER	
MSG027	FORMAT(S) ARE INVALID	THE DISPLAY AND/OR TABLE FORMAT IS INVALID. PLS SEE HELP FOR FORMATS	Refer to HELP or tablesONLINE documentation.
MSG028	TABLE PARMS DO NOT EXIST	THE TRAILER ROW IN THE VIEW DEFINITION DOES NOT EXIST, GO TO OPTION 1	
MSG029	VIEW DEFINITION SAVED	ONLY THE VIEW HAS BEEN SAVED	Information only.
MSG030	COMMAND CONFLICT	INVALID OR ILLEGAL SEQUENCE OF COMMANDS	
MSG031	ROW DELETED	THE ROW SELECTED HAS BEEN DELETED FROM THE TABLE	Information only.
MSG032	ROW ADDED	THE NEW ROW HAS BEEN ADDED TO THE TABLE	Information only.
MSG033	ROW UPDATED	THE ROW SELECTED HAS BEEN UPDATED AS REQUESTED	Information only.
MSG034	DUPLICATE ROW ADDED	THE ROW KEY DUPLICATES AN EXISTING KEY BUT THE ROW WAS ADDED ANYWAY	
MSG035	DISPLAY LEN OUT OF RANGE	THE DISPLAY LENGTH MUST BE BETWEEN &DMINL AND &DMAXL	
MSG036	TABLE LEN OUT OF RANGE	THE TABLE LENGTH MUST BE BETWEEN &TMINL AND TMAXL	
MSG037	TABLE FORMAT CONFLICT	THE TABLE FORMAT SPECIFIED IS IN CONFLICT WITH THE DISPLAY FORMAT	
MSG038	ROW IS EMPTY	TABLE ROW IS EMPTY OR THIS ROW DOES NOT EXIST ON THE TABLE	
MSG039	VIEW NOT FOUND,BASE USED	VIEW WITH DYNAMIC VIEW SUFFIX NOT FOUND, BASE VIEW USED INSTEAD	

Msg #	Short format	Long format	Meaning
MSG040	ALLOCATION ERROR	UNABLE TO ALLOCATE &LIB USING PARAMETERS SPECIFIED	
MSG041	ALLOCATION FAILED	&MLIB, PROBABLY NOT CATALOGUED OR ALREADY ALLOCATED	
MSG042	COPY ERROR - SAME LIB	COPY MUST BE TO ANOTHER LIBRARY IF THE SAME MEMBER NAME IS USED.	
MSG043	ALLOCATION FAILED	ALLOCATION FAILED BECAUSE FROM DATASET OR MEMBER DOES NOT EXIST.	
MSG044	FUNCTION FAILED	INVALID TABLE NAME(S) OR PASSWORD.	Re-enter command, or contact system administrator.
MSG045	DATA IS NOT FORMAT	ERROR &TBERR OCCURRED CONVERTING DISPLAY DATA TO TABLE FORMAT.	
MSG046	RENAME ERROR - SAME NAME	RENAME MUST BE TO DIFFERENT MEMBER NAME.	
MSG047	HOOK PROGRAM NOT LOADED	INSUFFICIENT REGION STORAGE FOR TBHOOK GETMAIN	
MSG048	HOOK PROGRAM MISSING	TBULOAD DATASET NOT FOUND.	
MSG049	HOOK PROGRAM MISSING	TBULOADDATASET COULD NOT BE OPENED.	
MSG050	DATE IS INVALID	ERROR &TBERR OCCURRED CONVERTING DISPLAY DATA TO TABLE FORMAT	
MSG051	DATA IS NOT NUMERIC	ERROR &TBERR OCCURRED CONVERTING DISPLAY DATA TO TABLE FORMAT	
MSG052	VALUE IS TOO HIGH	ERROR &TBERR OCCURRED CONVERTING DISPLAY DATA TO TABLE FORMAT	
MSG053	USER HOOK ERROR &TBERR	A USER HOOK RETURNED A NON-ZERO RETURN CODE	
MSG054	SERIOUS PROGRAM ERROR	ERROR &TBERR OCCURRED CONVERTING DISPLAY DATA TO TABLE FORMAT	

Msg #	Short format	Long format	Meaning
MSG055	ROW CONTAINS BAD DATA	ERROR &TBERR OCCURRED CONVERTING TABLE DATA TO DISPLAY FORMAT	
MSG056	EXIT PROGRAM MISSING	FIELD DEF SPECIFIES EXITNAME BUT EXIT PGM NOT FOUND IN LOAD LIBRARY	
MSG057	CANT INCREASE ROW SIZE	IF DATA EXISTS IN TABLE, CAN'T INCREASE ROW SIZE VIA tablesONLINE	
MSG058	LOCATE FLDNAME NOT FOUND	LOCATE MODE: NO MATCH FOUND ON SUPPLIED LOOKUP KEY	
MSG059	LOCATE FLDNAME -NO KEY	LOCATE MODE: LOOKUP KEY WAS NOT SUPPLIED	
MSG060	VIEW NOT DELETED	A GENERATION OF THE TABLE IS STILL RESIDENT ON THE TABLE LIBRARY	
MSG061	VIEW DELETED	THE VIEW DEFINITION FOR TABLE &TABLE HAS BEEN DELETED	Information only.
MSG062	INSERT BY COUNT FAILED	INSERT BY COUNT CAN ONLY BE PERFORMED ON A TABLE WITH ORGANIZATION = U	
MSG063	COUNT OUTSIDE TABLE	THE COUNT SPECIFIED IS OUTSIDE THE TABLE: LAST ROW COUNT IS &LASTCNT	
MSG064	COUNT SET TO LAST ROW	THE COUNT SPECIFIED IS OUTSIDE THE TABLE: COUNT SET TO LAST ROW	
MSG065	COUNT PAST LAST ROW	THE COUNT SPECIFIED IS OUTSIDE THE TABLE: COUNT SET 1 PAST LAST ROW	
MSG066	COUNT NOT NUMERIC	THE COUNT ENTERED IS BLANK OR CONTAINS INVALID CHARACTERS.	
MSG067	ROW RELOCATED IN TABLE	THE UNCHANGED ROW IS RELOCATED TO ANOTHER LOCATION IN THE TABLE	
MSG068	DUPLICATE KEYS NOT OK	THE TABLE HAS A HASH ORGANIZATION AND DUPLICATE KEYS ARE NOT SUPPORTED	

Msg #	Short format	Long format	Meaning
MSG069	COMMAND UNCLEAR; REENTER	PROBABLY ENTER WAS PRESSED WITH INCOMPLETE DATA IN COMMAND LINE	
MSG070	VIEW NOT FOUND	THE VIEW DEFINITION FOR TABLE &TABLE IS NOT ON THE LIBRARY	
MSG071	THE TABLE IS NOT FOUND	THE TABLE &TABLE IS NOT ON THE LIBRARY	
MSG072	TABLE & VIEW NOT FOUND	NEITHER THE TABLE &TABLE NOR ITS VIEW ARE ON THE LIBRARY	
MSG073	INVALID PASSWORD	TABLE=&TBLNAM IS PASSWORD PROTECTED; PASSWORD SUPPLIED IS INVALID	
MSG074	ROW SIZE IS > 10,000	TABLES WITH ROW-SIZE > 10,000 CAN'T BE ACCESSED VIA tablesONLINE/ISPF	
MSG075	TABLE EXISTS ON LIBRARY	TABLE = &ALTBL ALREADY EXISTS ON THE LIBRARY	
MSG076	BASE TABLE CANNOT BE ALT	BASE-TABLE CANNOT BE AN ALTERNATE TABLE	
MSG077	ALTERNATE INDEX CREATED	ALTERNATE INDEX HAS BEEN SUCCESSFULLY CREATED	Information only.
MSG078	ALTERNATE INDEX UPDATED	ALTERNATE INDEX HAS BEEN SUCCESSFULLY UPDATED	Information only.
MSG079	BASE TBL MUST BE POINTER	BASE-TABLE MUST BE OF POINTER TYPE INORDER TO CREATE AN ALT DEFINITION	
MSG080	CONVERSION COMPLETE	&CNTRC CONVERTED; &CNTRU UPDATED; &CNTRI NOT CONVERTED	Information only.
MSG081	NO TABLES IN LIBRARY	&LIB, HAS NO ISPF VIEWS IN IT	
MSG082	LIB(FROM) IS MANDATORY	FROM: LIBRARY NAME, IS A MANDATORY INPUT FIELD	
MSG083	INVALID PASSWORD	&CNTRC CONVERTED; &CNTRU UPDATED; &CNTRI NOT CONVERTED	
MSG084	ALLOCATION FAILED	LIBRARY=&MLIB, PROBABLY NOT CATALOGED OR DOES NOT EXIST	
MSG085	NO SPACE ON LIBRARY	&CNTRC CONVERTED; &CNTRU UPDATED; &CNTRI NOT CONVERTED	

Msg #	Short format	Long format	Meaning
MSG086	EXCLUDE=ALL NOT ALLOWED	ALL THE TABLES IN THE LIBRARY CANNOT BE EXCLUDED	
MSG087	TABLE DEFINITION UPDATED	THE DEFINITION OF THE TABLE HAS BEEN UPDATED	
MSG088	NEW VIEW DEFINED	A NEW VIEW HAS BEEN DEFINED	Information only.
MSG089	VIEW UPDATED	THE VIEW HAS BEEN UPDATED	Information only.
MSG090	DSPL AND TBLX LEN NOT EQ	DISPLAY AND TABLE LENGTHS MUST BE EQUAL	
MSG091	D-LEN = T-LEN + 1	DISPLAY LENGTH MUST BE EQUAL TO TABLE LENGTH PLUS ONE	
MSG092	D-LEN = (2 * T-LEN) <-1>	DISPLAY LENGTH MUST BE EQUAL TO TWO TIMES THE TABLE LENGTH (MINUS ONE)	
MSG093	D-LEN = 2 * T-LEN	DISPLAY LENGTH MUST BE EQUAL TO TWO TIMES THE TABLE LENGTH	
MSG094	D-LEN = T-LEN + 1 OR 2	DISPLAY LENGTH MUST BE EQUAL TO TABLE LENGTH PLUS ONE OR PLUS TWO	
MSG095	D-LEN = (2 * T-LEN) <+1>	DISPLAY LENGTH MUST BE EQUAL TO TWO TIMES THE TABLE LENGTH (PLUS ONE)	
MSG096	LENGTH FORMULA INVALID	AN INVALID LENGTH FORMULA HAS BEEN ENCOUNTERED IN THE EDIT TABLE	
MSG097	TBL DEFN CAN'T BE EDITED	TBL DEFN OF AN ALT TBL CAN'T BE EDITED, USE THE CREATE ALT OPTION	
MSG098	SUFFIX MUST BE 1-7 CHAR	THE DYNAMIC SUFFIX INDICATOR MUST BE A 1 TO 7 CHARACTER FIELD	
MSG099	ONLY ONE SUFFIX ALLOWED	IN A FIELD DEFINITION ONLY ONE SUFFIX FIELD IS VALID	
MSG100	CHANGES CANCELLED	CHANGES MADE TO THE TABLE WERE NOT SAVED	Information only.
MSG101	VIEW INCOMPATIBLE (CICS)	VIEW CONTAINS TBOL/CICS R5.0 FEATURES, EDITING MAY CORRUPT TABLE	Information only.

Msg #	Short format	Long format	Meaning
MSG102	VALIDATION ERROR	FIELD CONTAINS LOW VALUES, SPACES OR INVALID DATE	
MSG103	tablesONLINE - PRODUCT CODE INVALID	tablesONLINE - PROD CODE INVALID	
MSG104	tablesONLINE - CUSTOMER CODE MISMATCH	CUSTOMER CODE MISMATCH	
DK1105	tablesONLINE-LICENSE EXPIRED	tablesONLINE - LICENSE EXPIRED	
MTB000	COMMAND WAS SUCCESSFUL	THE COMMAND WAS PROCESSED SUCCESSFULLY	Information only.
MTB001	INVALID COMMAND	THE SPECIFIED COMMAND IS INVALID	
MTB002	TABLE NOT OPEN	THE SPECIFIED TABLE MUST BE OPEN FOR THIS COMMAND	
MTB003	TABLE IS NOT CLOSED	THE COMMAND REQUIRES THAT THE TABLE BE CLOSED	
MTB004	FG NOT ALLOWED	FG COMMAND IS NOT ALLOWED; TABLE NOT SEQUENTIAL (S OR D)	
MTB005	NO SECONDARY TABLE	THE SEARCH CRITERION WAS NOT FOUND IN THE PRIMARY TABLE	
MTB006	COUNT IS INVALID	THE COUNT SPECIFIED IS INVALID	
MTB007	DSN / DDNAME CHANGED	DSN OR DDNAME HAS BEEN CHANGED. USE DV OR DW BEFORE STORE	Information only.
MTB008	INVALID GENERATION	THE GENERATION NUMBER SPECIFIED FOR THIS TABLE IS INVALID	
MTB009	TABLE NOT FOUND	THE TABLE COULD NOT BE FOUND IN THE LIBRARY(S) SEARCHED	
MTB010	CS PARAMETER INVALID	THE STATUS-SWITCHES PARAMETER HAS INVALID ENTRIES	
MTB011	GENERATIONS INVALID	THE NUMBER OF GENERATIONS TO KEEP IN THE DT MUST BE 1-9	
MTB012	TABLE NAME INVALID	THE TABLE NAME SPECIFIED IS INVALID	

Msg #	Short format	Long format	Meaning
MTB013	COMMAND INVALID	THE COMMAND IS INVALID IN THIS ENVIRONMENT	
MTB014	ROW-SIZE IS INVALID	THE ROW-SIZE SPECIFIED IS INVALID. MUST BE FROM 1-32767	
MTB015	KEY SIZE IS INVALID	THE KEY-SIZE SPECIFIED IS INVALID. MUST BE FROM 1-256	
MTB016	KEY LOCATION IS INVALID	THE KEY-LOCATION SPECIFIED IS INVALID. MUST BE IN THE ROW	
MTB017	KEY NOT WITHIN ROW	THE KEY WILL NOT FIT WITHIN THE ROW	
MTB018	tableBASE LIBRARY FULL	INSUFFICIENT SPACE IS AVAILABLE ON THE tableBASE LIBRARY	
MTB019	TABLE ALREADY EXISTS	THIS TABLE ALREADY EXISTS ON THE TARGET LIBRARY	
MTB020	MAXNMTAB EXCEEDED	THE MAXIMUM NUMBER OF OPEN TABLES SETTING IS EXCEEDED	
MTB021	ORGANIZATION INVALID	ORGANIZATION SPECIFIED IS INVALID, USE R,U,S,D,H OR BLANK	
MTB022	tableBASE ERROR: 22	tableBASE ERROR: 22	Call technical support.
MTB023	tableBASE ERROR: 23	tableBASE ERROR: 23	Call technical support.
MTB024	tableBASE ERROR: 24	tableBASE ERROR: 24	Call technical support.
MTB025	tableBASE ERROR: 25	tableBASE ERROR: 25	Call technical support.
MTB026	tableBASE ERROR: 26	tableBASE ERROR: 26	Call technical support.
MTB027	tableBASE ERROR: 27	tableBASE ERROR: 27	Call technical support.
MTB028	PAGED NOT ALLOWED	PAGED TABLES ARE NOT SUPPORTED. USE DK1TCNV TO CONVERT	
MTB029	LIBS NOT THE SAME	THE SECONDARY TABLE NOT ON SAME LIBRARY AS PRIMARY TABLE	
MTB030	PASSWORD INVALID	THE PASSWORD SUPPLIED IS INVALID	
MTB031	WRITE PASSWORD INVALID	THE WRITE PASSWORD IS EITHER MISSING OR INCORRECT	
MTB032	TABLE NOT OPENED WRITE	THE ST COMMAND REQUIRES THAT THE TABLE BE OPENED FOR WRITE	

Msg #	Short format	Long format	Meaning
MTB033	DIFFERENT GENERATION	A DIFFERENT GENERATION OF THE TABLE IS ALREADY OPEN	
MTB034	tableBASE ERROR: 34	tableBASE ERROR: 34	Call technical support.
MTB035	tableBASE ERROR: 35	tableBASE ERROR: 35	Call technical support.
MTB036	tableBASE ERROR: 36	tableBASE ERROR: 36	Call technical support.
MTB037	tableBASE ERROR: 37	tableBASE ERROR: 37	Call technical support.
MTB038	tableBASE ERROR: 38	tableBASE ERROR: 38	Call technical support.
MTB039	NEW NAME ALREADY EXISTS	RN COMMAND FAILED. THE NEW NAME ALREADY EXISTS ON LIBRARY	
MTB040	DDNAME DOESNT EXIST	THE LIBRARY DDNAME DOES NOT EXIST	
MTB041	SMC VALUE INVALID	THE STORAGE-MODE-CODE (SMC) SPECIFIED MUST BE R OR BLANK	
MTB042	TB-FORMAT INVALID	THE TBPARM SUBPARAMETER TB-FORMAT MUST BE EITHER 0 OR A	
MTB043	ESTIMATED ROWS INVALID	THE ESTIMATED NUMBER-OF-ROWS IN DT BLOCK IS OUT OF RANGE	
MTB044	EXPANSION FACTOR INVALID	THE EXPANSION-FACTOR SPECIFIED IN DT BLOCK MUST BE 1-999	
MTB045	tableBASE ERROR: 45	tableBASE ERROR: 45	Call technical support.
MTB046	tableBASE ERROR: 46	tableBASE ERROR: 46	Call technical support.
MTB047	tableBASE ERROR: 47	tableBASE ERROR: 47	Call technical support.
MTB048	tableBASE ERROR: 48	tableBASE ERROR: 48	Call technical support.
MTB049	TOO FEW PARAMETERS	THIS COMMAND REQUIRES MORE PARAMETERS THAN WERE GIVEN	
MTB050	DIRTYPE INVALID	DIRTYPE SPECIFIED FOR THE LD COMMAND MUST BE V,D,T, BLANK	
MTB051	DENSITY INVALID	THE DENSITY SPECIFIED IN THE DT BLOCK MUST BE FROM 1-999	
MTB052	tableBASE ERROR: 52	tableBASE ERROR: 52	Call technical support.
MTB053	tableBASE ERROR: 53	tableBASE ERROR: 53	Call technical support.

Msg #	Short format	Long format	Meaning
MTB054	tableBASE ERROR: 54	tableBASE ERROR: 54	Call technical support.
MTB055	SEARCH-METHOD INVALID	THE SEARCH-METHOD MUST BE EITHER S,Q,B,C,H	
MTB056	INCOMPATIBLE METH/ ORG	SEARCH-METHOD IS INCOMPATIBLE WITH ORGANIZATION SPECIFIED	
MTB057	tableBASE ERROR: 57	tableBASE ERROR: 57	Call technical support.
MTB058	MODULE NOT FOUND	THE REQUESTED MODULE CANNOT BE FOUND IN THE LOADLIB	
MTB059	tableBASE ERROR: 59	tableBASE ERROR: 59	Call technical support.
MTB060	LIBRARY INVALID	THE SPECIFIED LIBRARY IS NOT SUITABLE	
MTB061	LIBRARY STATUS INVALID	THE LIBRARY STATUS IS INVALID SEE ERROR MSG 61 SUBCODES	
MTB062	LIBRARY FORMAT ERR	FORMAT OF SPECIFIED LIBRARY INCOMPATIBLE WITH VERSION 6	
MTB063	tableBASE ERROR: 63	tableBASE ERROR: 63	Call technical support.
MTB064	INVALID INDEX PARAMETER	THE INDEX PARAMETER SPECIFIED MUST BE P, T OR BLANK	
MTB065	tableBASE ERROR: 65	tableBASE ERROR: 65	Call technical support.
MTB066	tableBASE ERROR: 66	tableBASE ERROR: 66	Call technical support.
MTB067	tableBASE ERROR: 67	tableBASE ERROR: 67	Call technical support.
MTB068	COUNT TOO SMALL ON DU	THE COUNT VALUE IS TOO SMALL FOR DU COMMAND TO DUMP ROWS	
MTB069	tableBASE ERROR: 69	tableBASE ERROR: 69	Call technical support.
MTB070	tableBASE ERROR: 70	tableBASE ERROR: 70	Call technical support.
MTB071	LOCK WAIT EXCEEDED	WAIT FOR LOCK EXCEEDED LOCKTIMERC VALUE	
MTB072	TABLE UNAVAILABLE	THE TABLE IS UNAVAILABLE AT THIS TIME, WAIT=N IS IN EFFECT	
MTB073	NO UPDATING ALLOWED	THIS TABLE IS OPEN FOR READ, NO UPDATING ALLOWED	
MTB074	LOCK-LATCH INVALID	THE LOCK-LATCH SPECIFIED IS INVALID	

Msg #	Short format	Long format	Meaning
MTB075	ALLOC/DEALLOC FAILED	DYNAMIC ALLOCATION (AL) OR UN-ALLOCATION (UL) FAILED	
MTB076	PRIMARY TABLE ERROR	OPEN INDIRECT FAILED, PRIMARY NOT SEQUENTIAL OR KLOC NOT 9	
MTB077	TABLE NOT POINTER	ALTERNATE INDEX DEFINED FOR A TABLE THAT IS NOT A POINTER	
MTB078	ALLOC DISP INVALID	AL COMMAND FAILED, SHARE-STATUS PARAMETER MUST BE S OR O	
MTB079	tableBASE ERROR: 79	tableBASE ERROR: 79	Call technical support.
MTB080	DATA TABLE NOT OPEN	DATA TABLE NOT OPEN. USE OF ALTERNATE INDEX NOT ALLOWED	
MTB081	ALT DEF NOT FOUND	THE SPECIFIED ALTERNATE INDEX DEFINITION IS NOT FOUND	
MTB082	tableBASE ERROR: 82	tableBASE ERROR: 82	Call technical support.
MTB083	TABLE NOT POINTER	THE DATA TABLE INDEX SUBPARAMETER OF DT BLOCK MUST BE P	
MTB084	tableBASE ERROR: 84	tableBASE ERROR: 84	Call technical support.
MTB085	INVALID ALT COMMAND	THE COMMAND IS NOT VALID FOR ALTERNATE TABLE	
MTB086	OPEN INDEX FAILED	OPEN OF THE INDEX FAILED ON A LINKED TABLE IN A VTS-TSR	
MTB087	KEY-COUNT MUST BE 1	THE KEY-COUNT MUST BE 1 FOR DEFINITION OF ALTERNATE INDEX	
MTB088	tableBASE ERROR: 88	tableBASE ERROR: 88	Call technical support.
MTB089	ERROR IN TBOPT CARD	AN INVALID PARAMETER WAS ENCOUNTERED IN THE TBOPT FILE	
MTB090	INSUFFICIENT MEMORY	INSUFFICIENT MAIN STORAGE AVAILABLE. INCREASE REGION SIZE	
MTB091	I/O ERROR	I/O ERROR WHILE ATTEMPTING TO READ/WRITE TO LIBRARY/FILE	
MTB092	INSUFF TSR SIZE	INSUFFICIENT TABLE SPACE REGION (TSR) SIZE	
MTB093	RF INCOMPLETE	THE RF COMMAND DID NOT COMPLETE	

Msg #	Short format	Long format	Meaning
MTB094	tableBASE ERROR: 94	tableBASE ERROR: 94	Call technical support.
MTB095	TRANSACTION FAILED	THE TRANSACTION (JOB STEP) FAILED	
MTB096	tableBASE ERROR: 96	tableBASE ERROR: 96	Call technical support.
MTB097	ML/LL LIST INVALID	THE ML OR LL PARAMETER LIST CONTAINS A REFERENCE TO A VTS	
MTB098	INSTALL PARMS INVALID	THE INSTALLATION PARMS ARE NOT VALID FOR THIS OPERATION	
MTB099	tableBASE ERROR: 99	tableBASE ERROR: 99	Call technical support.

Table C-4: tablesONLINE/ ISPF messages and error codes

DK1TEXEC error messages

Error and audit messages generated by the batch utility program DK1TEXEC are identified and described in alphabetical order in the following table.

Message	Text	Meaning/Instructions
Change definition failed	The attempt to change the definition of the table has failed for the reasons noted above the message.	
Change is not for Alternate Index	The CHANGE ALT= command specified a table that is not an Alternate Index. Use the CHANGE TBL= version of the command.	
Change key size/location failed	The attempt to change the key size and/or key location has failed for reasons noted in preceding messages or to the right of this message.	
Change maxgen failed	The attempt to change the number of generations to be kept has failed for reasons noted in preceding messages or to the right of this message.	
Change not made for reasons above	None of the changes in the above change were performed. The reason for the rejection of this command is noted above the message.	
Change successful	The table definition has been changed as requested.	Information only.
CMD requires specified table to be open	This command requires the specified table to be open. This error should not occur when using DK1TEXEC.	
Command is invalid	The specified command is not a DK1TEXEC command.	

Message	Text	Meaning/Instructions
Command is invalid for an Alternate Index	The command cannot be performed on an Alternate Index.	
Command rejected for reason above	The preceding DK1TEXEC Command was rejected for the reasons noted above the message.	
Command requires more parms than given	DK1TEXEC called tableBASE incorrectly.	
Copy complete	The requested operation completed successfully.	
Copy complete except where indicated	The requested copy operation was successfully completed except as noted above the message.	
Copy failed	The requested operation failed. Diagnostic messages are issued to give the reason(s) for the failure.	
Copy of table (all generations) complete	The requested copy operation of all generations of a table was successfully completed.	
Create Alternate Index definition failed	The attempt to create an Alternate Index definition failed for the reasons noted in preceding messages or to the right of this message.	
Create Alternate Index definition successful	The Alternate Index definition has been created successfully.	Information only.
Data table index must be defined as P	DK1TEXEC attempted to invoke an Alternate Index of a table that is not defined to be type P (indexed or pointer).	
Defaults have been set	The defaults entered have been set.	
DEFINE TEMPORARY TABLE FAILED	Internal error defining temporary table for <filename>.	Please increase TSR size and rerun.
Delete failed	The requested delete operation has failed for reasons noted in preceding messages or to the right of this message.	
Delimiter is invalid, = was expected	The next symbol after a keyword must be an equal sign.	
Density parameter must be from 1-999	The density must be numeric and in the range 1-999.	
Destination library too small	The destination library is too small to hold the table(s) to be copied.	
Directory is empty	The directory on the specified library is empty.	
DT command (maxgen parm) must be 1-9	The maximum generations to be kept must be numeric in the range 1-9 inclusive.	
Duplicate keyword for this command	The same keyword was specified twice for a command. Fix the command sequence and try again.	Fix the command sequence and try again.

Message	Text	Meaning/Instructions
End of data - tableBASE utility ended	End of data has been reached on CNTLCARD, the DK1TEXEC input file.	Information only.
Error In table definition	An error has been detected in the table definition created from the parameters supplied. Refer to the TBCALL Error Code for the reason.	
Estimated number-of-rows is out of range	The estimated number of rows is non-numeric, or is too large.	
Expand library failed	The requested operation failed. Diagnostic messages are issued to give the reason(s) for the failure.	
Expand library successful	The requested operation completed successfully.	Information only.
Expansion factor must be from 1 To 999	The expansion factor must be numeric and in the range 1-999.	
Field is greater than 8 characters	All of the keywords and keyword values have a maximum length of eight characters.	Ensure field is 8 characters or less.
Format of library incompatible with Version 6 and higher versions.	The library format has changed starting with Version 6. It needs to be converted to be compatible. Please see the tableBASE Installation Guide, Appendix B.	
Generation number specified is invalid	The generation must be numeric and in the range 0-255.	
Generation requested has been copied	The COPY operation has been completed successfully. A new table (generation number 1) has been added to the target library.	Information only.
Index parameter must be P, T or blank	The Index field must be T for true tables, P for indexed (Pointer) tables or blank. In previous releases, tableBASE allowed two types of tables: Pointer and True. Starting with Version 6, the concept of True tables still exists, however they are treated within tableBASE as Pointer tables as all memory is now in segmented memory that requires Indexes. The True table Indexes will be transparent to the application program.	
Initialization successful	The new tableBASE library has been initialized successfully.	Information only.
Insufficient local tableSPACE region	The TSR is not large enough to contain the table and tableBASE internal tables. tableBASE uses a small part of the TSR for its own internal tables.	
Insufficient space on new library	There is not enough space on the target library for the tables being copied.	
INV storage-mode-code: Must be R, Blank	The storage mode code (SMC) must be R or blank.	

Message	Text	Meaning/Instructions								
Key size invalid: must be 1-256	The key size must be a number from 1-256, inclusive.									
Keyword and value are incompatible	If the keyword requires a numeric value, the value specified with it must be numeric. Alternatively, the ALLGEN keyword must be followed by = YES.									
Keyword invalid for this command	The keyword specified was not recognized.									
Keyword is incomplete	The keyword has not been followed by an equal sign (=) and a value.									
Keyword is invalid	The keyword specified is not used with this command.									
KLOC invalid: must be 1 to row-size	The key location must be numeric in the range 1 to the row size, inclusive.									
Library is not empty	The target or destination library of an EXPAND LIBRARY request contains at least one table. The target library must be empty for an EXPAND operation.									
Library not initialized	For the reason cited to the right of this message, the library was not initialized.									
Load failed	The requested operation failed. Diagnostic messages are issued to give the reason(s) for the failure.									
Mthd/org parameters are incompatible	This combination of search method and organization will not work. Valid combinations are:<table><tr><th>Organization</th><th>Method</th></tr><tr><td>R, U</td><td>S</td></tr><tr><td>S, D</td><td>S, B, C</td></tr><tr><td>H</td><td>H</td></tr></table>	Organization	Method	R, U	S	S, D	S, B, C	H	H	
Organization	Method									
R, U	S									
S, D	S, B, C									
H	H									
New generation loaded successfully	A new generation of this table has been created from the contents of the FROM dataset.	Information only.								
Newlib DDNAME not Assigned -- check JCL	The requested initiation of the DDNAME specified by the NEWLIB parameter could not be performed since the DDNAME is not defined in the JCL.	Alter JCL as required.								
nnnnn/mmmmm tables exported successfully	nnnnn tables were exported from the tableBASE library successfully. mmmmm tables were requested to be exported.	Information only.								
nnnnn/mmmmm tables imported successfully	nnnnn tables were imported to the tableBASE library successfully. mmmmm tables were requested to be imported.	Information only.								
No changes specified	A change command did not identify any fields to be changed.	Identify fields as required.								

Message	Text	Meaning/Instructions
Not copied, new name same as old name	A COPY table request specified the same value for NEWNAME as for TBL.	
Not enough space on library	There is not enough space on the tableBASE library for the table being defined, copied, loaded, or expanded.	
Old Alternate Index definition deleted	During a copy of an Alternate Index definition, the old definition was deleted from the target library, but the new definition could not be copied from the source library to the target library.	
On dest lib - insufficient space for copy	The copy operation has been requested to a destination library with insufficient space on it to receive all generations of the tables on the FROM library.	
Org parm invalid: must be R, U, S, D or H	The Organization must be R, U, S, D, or H.	
Paged tables are not supported	Starting with Version 6, paged tables are no longer supported.	
Print request completed successfully	The requested print operation has finished with no errors. The listing can be found in TBRPT.	Information only.
Rename failed	The requested operation failed. Diagnostic messages are issued to give the reason(s) for the failure.	
Rename successful	The requested rename operation was successfully performed.	Information only.
Requested generation has been cleared	A new generation of this table has been created using the definition of the generation specified. This new generation contains no items.	Information only.
Requested generation has been deleted	The generation of the table specified has been deleted.	Information only.
Requested generation(s) deleted	The requested operation completed successfully.	Information only.
Required keyword missing for this command	One of the required keywords for this command was not contained in the command sequence.	Correct command sequence as required.
RN failed. New name already exists.	The table was not renamed because a table with the new name already exists on the library.	
Row size invalid: must be 1-32767	The row size must be numeric in the range 1-32767 inclusive.	

Message	Text	Meaning/Instructions								
Search-method incompatible with organization	The search method must be S, Q, B, C, or H and must match the organization. Valid combinations are: <table><tr><th>Organization</th><th>Method</th></tr><tr><td>R, U</td><td>S</td></tr><tr><td>S, D</td><td>S, B, C</td></tr><tr><td>H</td><td>H</td></tr></table>	Organization	Method	R, U	S	S, D	S, B, C	H	H	
Organization	Method									
R, U	S									
S, D	S, B, C									
H	H									
Semi-colon is missing from command	The command and its keywords must be followed by a semicolon (;) to indicate the end of the keyword list for the command sequence.									
Source library is empty	The source library in a copy operation contains no tables.									
Table already exists on library	The table could not be defined on, or copied to, the new library because a table of the same name already exists on the target library.									
tableBASE error detected	The requested operation failed. Diagnostic messages are issued to give the reason(s) for the failure.									
tableBASE internal program error - xxxx	The requested operation failed. Diagnostic messages are issued to give the reason(s) for the failure.									
Table definition successful	The table was defined successfully and generation 1 was stored on the library.									
Table is not closed	This error should not occur when using DK1TEEXEC.	Contact Technical Support.								
Table is not found	The table was not found on the given library.									
Table name specified is invalid	A valid tableBASE table name is a string of 8 bytes that are not all blanks, all low values, all high values, or :TMPNAME.									
Table opened for read cannot be stored	A table has been opened for read-only access and a store or write command has been issued.									
Table type is invalid	The table type must be F, V, A, or X.									
Table unavailable. No wait in effect	DK1TEEXEC could not store the table because it is locked by another application. The NO WAIT parameter is in effect.									
Table unloaded successfully	The specified generation of the specified table has been unloaded to the TO dataset.	Information only.								
Table updated as requested	The requested update operations on the specified table were performed successfully.	Information only.								
Table updated successfully	The requested operation completed successfully.	Information only.								

Message	Text	Meaning/Instructions
Table xxxxxxxx could not be opened	For reasons noted in preceding messages or to the right of this message, the table xxxxxxxx could not be opened. Since the table must be opened to perform this DK1TEXEC command, the DK1TEXEC command could not be successfully completed.	
Table xxxxxxxx export failed	The export of table xxxxxxxx failed for the reasons given in other messages.	
Table xxxxxxxx import failed	The import of table xxxxxxxx failed for the reasons given in other messages.	
Table xxxxxxxx import replace failed	The import of table xxxxxxxx failed for the reasons given in other messages.	
Table xxxxxxxx sort error	The sort required for this operation failed for the reason noted.	
Table xxxxxxxx was not stored	The table xxxxxxxx was not stored for the reasons noted in preceding messages or to the right of this message.	
tablebase error detected	A DK1TCALL error was detected in the processing of this command. The explanation of the error is found to the right of this message.	
The count specified is invalid	The count is less than one, or greater than the number of rows in the table.	
The key will not fit within the row	The end of the key cannot exceed the end of the row.	
The library DDNAME does not exist	The LIB=DDNAME is not in the JCL.	Alter JCL as required.
The library status is invalid	tableBASE library disposition must be NEW for new libraries, and SHR or OLD for existing libraries.	
The password supplied is invalid	An incorrect password has been specified.	
The specified command is invalid	The specified command is not a valid DK1TEXEC command.	
The specified library is not suitable	There are a number of conditions that can create this error. See tableBASE error codes.	
Write password is missing or incorrect	The operation requires access to a write-protected table, but the write password has not been specified or specified incorrectly.	
xxxxxxx field is greater than 8 characters	All keywords and keyword values must be eight characters or fewer.	

Message	Text	Meaning/Instructions
<tablename> NOT FOUND IN SOURCE LIBRARY	Specified table is not in the source library.	Please specify correct table associated with EXCLUDE parameter and/or EXCLFILE parameter file.
SOURCE BLKSIZE(xxxx) AND TARGET(xxxxx) DIFFER	INCREASE TARGET LIB SPACE ALLOCATION FOR COPY TO SUCCEED	

Table C-5: tableBASE batch utility messages

VTSManger Error messages

Command Interpretation Errors

Command interpretation errors are detected during the parsing of command sequences. When a command interpretation error occurs, an error message is issued and DK1MEXEC proceeds with the next command sequence.

Message	Description Action
End of data - DK1MEXEC Utility ended	Informational
Delimiter is invalid, = is expected	An equal sign (=) must follow the keyword in the parameter
Command is invalid	The command is not recognized
Keyword is invalid	The Keyword is not recognized
xxxxxxxx ... field is greater than 8 characters	xxxxxxxx is the first eight characters of the field that is too long
Semi-colon is missing from the command	Place a semi-colon at the end of each command sequence
Keyword is incomplete	
Keyword is invalid for this combination	
Required Keyword missing for this command	
Duplicate Keyword for this command	A valid keyword was specified more than once in a command sequence. This is not allowed.

Message	Description Action
Invalid name specified	Alias and VTS names must be one to eight characters. The first character must be alphabetic or one of the national characters @, \$, or #. Remaining characters must be alphabetic, numeric or one of the national characters @, \$, or #.
COMMAND REJECTED FOR ABOVE REASON	Informational. One or more previous messages describe the error.
SHOW REQUEST COMPLETED	Informational
DEFAULTS SET SUCCESSFULLY	Informational
DK1MEXEC INTERNAL PROGRAM ERROR	Contact DataKinetics technical support for assistance.
SERIOUS ERROR. DK1MEXEC IS ABENDING	Contact DataKinetics technical support for assistance.

Table C-6: Command Interpretation messages

Command Processing Errors

Command processing errors are detected during the execution of commands. When a command processing error occurs, an error message is issued and DK1MEXEC proceeds with the next command sequence.

Code	Message	Description/Action
99	DK1M2063 MODESET Key 0 Error	Program was not run from an APF-authorized library.
820	PC01/TLICK Mismatch	
821	PC Server does not support Product	
822	Product not supported by licensing	
823	Program not authorized for Key 0	DK1MEXEC must run from an APF-authorized library to invoke the LICENSE command.
824	PC Server not found	The PC Server for this version is not running on this LPAR.
825	License subroutine returned error	Contact DataKinetics Technical Support for assistance.
830	DK1xBASE missing	VTSManager requires the DK1MBASE module of the correct version to be available. The LICENSE command could not find the DK1xBASE module for this product.

Code	Message	Description/Action
831	DK1xBASE version mismatch	VTSManger requires the DK1MBASE module of the correct version to be available.
832	DK1xBASE Returned an error	Contact DataKinetics Technical Support for assistance
850		An internal error has occurred. Contact DataKinetics Technical Support for assistance.
851		An internal error has occurred. Contact DataKinetics Technical Support for assistance.
852	Product or Version not supported	
853	DK1TLICK could not find TPM	Contact DataKinetics Technical Support for assistance.
2303	Invalid Service Code	Contact DataKinetics Technical Support for assistance.
2305	Alias name already in use	
2311	Alias not found	
2312	VTs not running under this Alias	
2313	VTs not found in this GCE	
2314	FROMVTs not associated with this Alias	
2315	VTs not running	
2317	VTs not read-only	An Alias may only be assigned to a Read Only VTs
2318	FROMVTs not found	
2322	VTs already SWITCHed to this Alias	
2323	VTs is associated with a different Alias	An Alias may only be assigned to one VTs at a time.
2330	TPVM control block not found	Contact DataKinetics Technical Support for assistance.
2602	VTs Manager is not licensed	VTSManger license is not set or has expired.

Table C-7: Command Processing Messages

Licensing Warnings and Errors

Some VTSManger messages and messages issued by other products indicate that your continued use of VTSManger may be impacted by the potential expiration of a license. Such messages

are written to both the MVS system console and the job's JES message log, to alert Systems Administrators and Operations staff, in addition to end users.

Message Number	Message	Description/Action
DK100805W	License checking error <nnn> for <pppppppp>	The license checker encountered an error. Contact DataKinetics Technical Support for assistance.
DK100806W	Error <nnn>-<rrr> LOADIng DK1TLICK; <pppppppp> not licensed	Error loading the license checker. Contact DataKinetics Technical Support for assistance.
DK100807W	Error <nnn> ATTACHIng DK1TLICK for <pppppppp>	Error attaching the license checker. Contact DataKinetics Technical Support for assistance.
DK100820W	DK1TPC01/DK1TLICK interface mismatch <nnn/ nnn>	Contact DataKinetics Technical Support for assistance.
DK100821W	PC Server <vvvv> does not support <pppppppp>	Contact DataKinetics Technical Support for assistance.
DK100822W	<pppppppp> not supported by licensing routine	Contact DataKinetics Technical Support for assistance.
DK100823W	DK1TLICK not authorized to enter key 0	Contact DataKinetics Technical Support for assistance.
DK100824W	PC Server <vvvv> not found	Ensure that the PC Server for this release is running.
DK100825W	Licensing subroutine returned RC=<rrrrr>	Contact DataKinetics Technical Support for assistance.
DK100826W	<pppppppp> license expired <yyyy-mm-dd>; in grace period	The license for the named product has expired and it is now running in the grace period. It will stop functioning after the grace period.

Message Number	Message	Description/Action
DK100827W	<pppppppp> license expired <yyyy-mm-dd>	The license for the named product has expired and the product has stopped functioning.
DK100828W	<pppppppp> license will expire <yyyy-mm-dd>	The named product is in the warning period for its license and will no longer be licensed on the specified date.
DK100829W	<pppppppp> license expires today <yyyy-mm-dd>	Just after midnight today, the named product will no longer be licensed. If there is a grace period, the product will continue to function during that period.
DK100830W	<pppppppp> license module DK1xBASE missing	The license module specified for the named product is missing.
DK100831W	<pppppppp> DK1xBASE <vvvv> mismatch <vvvv>	The license module specified for the named product is for a different version
DK100832W	<pppppppp> DK1xBASE return code <rrrr>	Contact DataKinetics Technical Support for assistance.
DK100833I	<pppppppp> licensed to <customer>	Information only

Table C-8: License-Related Messages

Appendix

D

PC Server implementation in a CICS environment

Implementation and Restart/Recovery Considerations

The tableBASE PC Server provides support for running CICS threadsafe transactions which access tableBASE. The tableBASE PC Server support was introduced in Modification Level 7 of tableBASE Release 6.0.2. It also supports a task-end cleanup exit for jobsteps accessing tableBASE.

In order to prevent tableBASE application problems, tableBASE (at Modification Level 7 and above) will not initialize in a CICS region if a PC Server at the same or higher maintenance level is not running when tableBASE is first initialized in a CICS region.

It is intended that the tableBASE PC Server be started at IPL of an LPAR and never stopped. If it is stopped, it should be immediately restarted. Since the PC Server does not execute any code except during its initialization or termination, it will not abend. However, operations or other personnel authorized to stop or cancel jobs on a z/OS system could stop it.

Once the tableBASE PC Server is successfully started, the PC-routine module is resident in memory. This module, which is used by tableBASE transactions, remains in memory and is not released when the PC Server goes down. Therefore, tableBASE transactions running in regions on that LPAR will continue to process tableBASE transactions without problems. (They do not access the PC Server directly; they use the PC-routine module in memory.)

However, if these CICS regions are recycled without the PC Server running, the in-memory PC-routine module will not exist, and therefore tableBASE will not initialize in these regions and no tableBASE transactions will work. The message displayed in these regions will be:

```
"DK100567E The tableBASE PC Server must be running for tableBASE  
to initialize."
```

If the tableBASE PC Server is successfully restarted before the CICS regions are recycled, the in-memory PC-routine will be generated, and any newly recycled CICS regions will access the new copy of the in-memory PC-routine. tableBASE will initialize successfully and tableBASE transactions will work. The existing CICS regions will continue to access the old copy of the in-memory PC-routine and will continue to work.

In order to move from an environment in which no PC Server was running on an LPAR or an older version of the PC Server was running, the following procedures are recommended.

Steps to Implement the PC Server

When the current level is prior to tableBASE maintenance level 7:

If the z/OS system will be IPLed, start the higher maintenance level PC Server before starting any CICS regions or other jobs accessing tableBASE.

If the system will not be re-IPLed, the following steps allow the PC Server to be introduced in a running system.

Assuming there are CICS regions running current-level tableBASE and the current-level PC Server is running:

1. Stop the current-level PC Server.

- Use the MVS stop ("p") command
- Non-threadsafe CICS transactions accessing tableBASE will continue to work.
- While the PC Server is down, the following messages may appear on the MVS log when jobsteps accessing tableBASE initialize:

```
'DK100840I GCA does not exist.'  
'DK100201I tableBASE PC Server not available'.  
(These are informational messages and can be ignored.)
```

2. Bring up the higher-level PC Server (while CICS is running current-level tableBASE code).

- tableBASE will continue to run in all regions.
- non-threadsafe tableBASE CICS transactions will continue to run.
- threadsafe CICS transactions accessing tableBASE may abend or hang since pre-level07 tableBASE code doesn't support CICS threadsafe transactions.

3. Start CICS regions with higher-level tableBASE code.

- non-threadsafe CICS transactions accessing tableBASE will continue to work.
- threadsafe CICS transactions accessing tableBASE will now work.

When the current level is tableBASE maintenance level 7 or higher:

Review the Release Notes for the new level of maintenance to determine if any additional steps are necessary.

If the z/OS system will be IPLed, start the higher maintenance level PC Server(s) before starting any CICS regions or other jobs accessing tableBASE.

If the system will not be re-IPLed, the following steps allow the PC Server(s) to be introduced in a running system.

Assuming there are jobs, including CICS regions, running current-level tableBASE and the current-level PC Server(s) are running:

1. Stop the current-level PC Server(s).

- Use the MVS stop ("p") command

- Running jobs accessing tableBASE will continue to work.
- While the PC Server(s) are down, the following messages may appear on the MVS log when jobsteps accessing tableBASE initialize:

```
'DK100201S tableBASE PC Server unavailable'  
'DK100567E The tableBASE PC Server must be running for  
tableBASE to initialize'
```

The DK100201S message results in a U201 abend of the step.

The DK100567E message, which appears in a CICS region, allows the CICS region to initialize, but tableBASE transactions will not work.

2. Bring up the higher-level PC Server(s) (while CICS is running current-level tableBASE code).
 - tableBASE will continue to run in all regions
 - tableBASE CICS transactions will continue to run.
3. Start jobs, including CICS regions, with higher-level tableBASE.
4. Start jobs, including CICS regions, with current-level tableBASE.
 - The higher-level PC Server(s) are backward compatible

RESTART/RECOVERY Considerations

A successful start of the PC Server(s) will display the following message on the system console:

```
DK100800I tableBASE PC Server Vvrm available
```

If the PC Server(s) are stopped or cancelled by the operator while jobsteps, including CICS regions, are accessing tableBASE, the jobs will continue to run and access tableBASE. However, new jobsteps will not be able to start accessing tableBASE until the PC Server(s) are restarted.

To prevent operational problems related to the PC Server(s) not running, whether after an IPL or due to the PC Server step being stopped or cancelled, DataKinetics recommends the following procedures be considered:

1. The PC Server(s) should be started automatically at IPL before any batch jobs, CICS regions, IMS MPRs or DB2 SPAS that access tableBASE are started.
2. The appearance of message DK100800I could be used to trigger the start of any jobs that access tableBASE.
3. Any current facility which monitors availability of critical subsystems should include monitoring the PC Server(s) jobname.
4. If the PC Server(s) are not running at any given time, a restart of the job should be automatically triggered.
5. If the monitor does not trigger it, the appearance of message DK100201 or DK100567 should be set up to trigger it.
6. There is no danger in attempting to start the PC Server(s) multiple times; subsequent PC Server tasks will end with a message indicating that the PC Server is running (for message details, see [tableBASE messages](#) on page 161).

7. If CICS regions have started while the PC Server(s) are not running, tableBASE will fail to initialize; a message will be returned indicating this condition (for message details, see [tableBASE messages](#) on page 161).

tableBASE will not be available in these regions even if the PC Server(s) are subsequently started. After the PC Server(s) are started, this message could be used to trigger recycling these regions or, alternatively invoking the TBST TBINIT transaction (supplied by DataKinetics) in the affected regions. (Contact DataKinetics for further information on using this alternative technique).

NOTES and CAUTIONS

- The PC Server(s) PC routines are loaded into memory when it is started. They do not overlay any previous tableBASE PC routines. Although the routines are quite small, if the PC Server(s) were stopped and restarted thousands of time, they could theoretically finally occupy all memory available.
- Only one version of the PC Server(s) can be running on an LPAR at any time (whether it is initiated as a started task OR as a batch job). The prior version PC Server(s) must be stopped before a new one can be started.

Appendix

E

VTSGate and VTSManger Compatibility

Scope

When both VTSManger and VTSGate are licensed and in use, and that the RACF security feature of VTSGate is enabled throughout. All compatibility behavior described here is predicated on the assumption both products are licensed and in use. If RACF protection is not in use, the access control rules described below do not apply.

It is also important to distinguish two different types of dynamic switching discussed in this document to avoid confusion:

- **VTSManger Alias Switching** — VTS Manager's ability to dynamically switch a VTS-TSR from one VTS name (true name) to an alias name, or vice versa.
- **VTSGate Access Swapping** — VTS Gate's ability to dynamically swap a VTS-TSR's access mode between RO (read-only) and RW (read-write) using the VV command.

These are distinct operations and should not be confused with one another.

VTSManger with RACF protection from VTSGate

Description of the VV Command

The VV command is designed to vary the access mode of a VTS that has been initialized in read-only mode — that is, where the TSRACCESS option is set to RO. The VTS name must be specified in the TB-SUBSYSTEM field of TB-PARM, and the desired operation is determined by a request ID supplied in the FUNCTION-ID field of the COMMAND-AREA.

When a VTS is swapped to read-write mode via the VV command, the job performing the swap acquires a read-write lock on the VTS. No other job or region can access the VTS while this lock is held. The VTS remains locked in read-write mode until it is explicitly swapped back to read-only by the same job step that holds the lock.

Compatibility Consideration: VV and TSRACCESS=RF

When both VTSManger and VTSGate are licensed, users have access to the VV command. However, **VV is incompatible with TSRACCESS=RF**. The VV command cannot override or bypass the RF access setting, and attempting to use VV against a VTS initialized with TSRACCESS=RF will result in the following message:

DK100710W TSRACCESS forced to RO: TSR_KEY_PROTECT set

This incompatibility is a meaningful constraint when planning VTS configurations that are intended to support both VTSManger alias switching and VTS Gate access swapping.

VTS Configurations with LDS Backing and VTS Manager Alias Switching

TSRACCESS=RF

When an LDS-backed VTS is configured with TSRACCESS=RF, it is designed to support VTSManger alias switching operations. To enable this, a RACF profile must be defined for the VTS before it is started. Any user who needs to start the VTS, or who requires full access including both read and write operations, must be explicitly granted CONTROL-level access in the associated RACF profile.

Regarding RACF profile definitions for true names and alias names: both the true VTS name and any alias names associated with a given VTS-TSR resource must have consistent RACF definitions, meaning that end users generally need READ access to both.

From an operational standpoint, VTSManger SWITCH and RESET commands remain subject to the standard rules governing VTS Manager. When issuing tableBASE commands against such a VTS, the issuing user must hold READ or WRITE RACF privileges appropriate to the type of command. Failure to satisfy these requirements results in errors 1072-3 or 1072-4, respectively.

Despite allowing RACF WRITE privilege assignments, a VTS running under TSRACCESS=RF is treated as effectively read-only. The only permitted update-type operation is the RF (Refresh) command, which reloads tables into the VTS. Any other update commands — such as IK or RK — will fail with error 13-6 regardless of the user's RACF WRITE privilege level. As noted above, the VV command from VTS Gate is also not compatible with this mode.

TSRACCESS=RO

An LDS-backed VTS may also be configured with TSRACCESS=RO to support VTSManger alias switching. The same foundational RACF setup applies: a profile must be defined, CONTROL-level access is required to start the VTS or obtain full access, and consistent RACF definitions are needed for both true and alias names. VTSManger command boundaries and tableBASE privilege enforcement apply identically to this configuration as they do under TSRACCESS=RF.

Behaviorally, a VTS under TSRACCESS=RO is fully read-only. Any update commands such as IK or RK will fail with error 13-6, even when the issuing user holds RACF WRITE privileges.

TSRACCESS=RW

The Administration Guide explicitly advises against this configuration, noting that LDS-backed Read/Write VTS-TSRs are not recommended.

An LDS-backed VTS can nonetheless be configured and started with TSRACCESS=RW, and the same RACF CONTROL-level access rules from the RF configuration apply. However, a critical operational limitation exists: VTS Manager is not designed to perform an alias switch when the VTS is not in RO or RF mode. Attempting such a switch will produce the following error:

ERROR: VTS not Read-Only or RF

In terms of data access behavior, TSRACCESS=RW operates as a true read-write VTS. Users holding RACF WRITE privileges may execute both fetch and update commands. Users holding only RACF READ privileges are restricted to fetch operations.

VTS Configurations without LDS Backing and VTS Manager Alias Switching

Important Context

The configurations described in this section involve VTS-TSRs that are not managed or owned by VTS Manager. Because these VTS-TSRs are not VTSManger-managed, TSRACCESS=RF, which exists specifically to support VTSManger alias switching in LDS-backed environments, is not applicable here and is therefore not supported by VTS Gate in this context.

However, VTS Gate provides its own mechanism for dynamic access swapping via the VV command. Because VTSManger can still perform an alias switch while a non-LDS VTS-TSR is in RO mode, it is possible to coordinate VTS Gate access swapping with VTSManger alias switching: VTS Gate swaps the VTS to RW for table preparation, then swaps it back to RO, at which point VTSManger's alias switch can be performed.

Read-Refresh Access (TSRACCESS=RF)

VTS Gate does not support TSRACCESS=RF for non-LDS-backed VTS configurations. This mode is not applicable outside of the VTSManger LDS-managed context and cannot be used.

Read-Only Access (TSRACCESS=RO)

A non-LDS-backed VTS can be configured with TSRACCESS=RO, and the same RACF rules that govern LDS-backed TSRACCESS=RO configurations apply here as well — covering profile definitions, CONTROL-level access, consistent naming definitions, VTSManger command boundaries, and tableBASE privilege enforcement.

Beyond these standard rules, this configuration unlocks the primary operational advantage of using VTS Gate in this context. Using the VV command, a designated batch job can swap the VTS from RO to RW, perform necessary table preparation or updates, and then swap it back to RO. While the VTS-TSR is in RO mode, VTSManger's alias switch can be performed normally. This combination of VTS Gate access swapping and VTSManger alias switching provides a flexible and controlled mechanism for managing table content while preserving the read-only production state.

Read-Write Access (TSRACCESS=RW)

Non-LDS-backed VTS instances configured with TSRACCESS=RW follow the same rules as their LDS-backed counterparts described above.