



GT.M V5.5-000



GT.M Messages and Recovery Procedures Manual

GT.M Message and Recovery Procedures Manual

Publication date February 27, 2012

Copyright © 2012 Fidelity Information Services, Inc. All rights reserved.

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts and no Back-Cover Texts.

GT.M™ is a trademark of Fidelity Information Services, Inc. Other trademarks are the property of their respective owners.

This document contains a description of GT.M and the operating instructions pertaining to the various functions that comprise the system. This document does not contain any commitment of FIS. FIS believes the information in this publication is accurate as of its publication date; such information is subject to change without notice. FIS is not responsible for any errors or defects.

Revision History

Revision V5.5-000

27 February 2012

The following messages were added:

- BADREGION
- COREINPROGRESS
- DBROLLEDBACK
- DSEWCREINIT
- GTMASSERT2
- IGNBMPMRKFREE
- INSNOTJOINED
- INSRLECHANGE
- INSUNKNOWN
- INVTRCGRP
- JIUNHNDINT
- JNLALLOCGROW
- JNLNOREPL
- JRTNULLFAIL
- JNLRECINCMPL
- LOCKSPACEINFO
- LOCKSPACEUSE
- MAXSEMGETRETRY
- MUINFOUNT6
- MUTRUNC1ATIME
- MUTRUNCBACKINPROG
- MUTRUNCERROR
- MUTRUNCFAIL

- MUTRUNCNOSPACE
- MUTRUNCNOTBG
- MUTRUNCNOV4
- MUTRUNCPERCENT
- MUTRUNCSSINPROG
- MUTRUNCSUCCESS
- NOLOCKMATCH
- NONASCII
- NORESYNCSUPPLONLY
- NORESYNCDATERONLY
- NOSUPPLSUPPL
- ORLBKCMPLT
- ORLBKFRZOVER
- ORLBKFRZPROG
- ORLBKNOSTP
- ORLBKNOV4BLK
- ORLBKSTART
- ORLBKTERMNTD
- PERMGENDIAG
- PERMGENFAIL
- RCVR2MANY
- RCVRMANYSTRMS
- REPL2OLD
- REPLINSTDBSTRM
- REPLINSTMISMTCH
- REPLUPGRADESEC
- RESOLVESEQNO
- RESOLVESEQSTRM
- RESUMESTRMNUM
- REUSEINSTNAME
- RLBKCONFIGBNDRY
- RLBKSTRMSEQ
- RNDWNSKIPCNT
- RSYNCSTRMSUPPLONLY

- RSYNCSTRMVAL
- SECNOTSUPPLEMENTARY
- STRMNUMMISMTCH1
- STRMNUMMISMTCH2
- STRMSEQMISMTCH
- SUPRCVRNEEDSSUPSRC
- SRVLCKWT2LNG
- STRMNUMIS
- TRIGMODREGNOTRW
- UPDSYNC2MTINS
- UPDSYNCINSTFILE
- ZDATEBADDATE
- ZDATEBADTIME
- ZTRIGNOTRW

The following messages were marked as deprecated (●).

- ACTLSTEXP
- FMLLSTPRESENT

The following messages were changed:

- DBFLCORRP
- JOBEXAMFAIL
- DBCDBNOCERTIFY
- LOCKSPACEFULL
- NOREPLCTDREG
- PRIMARYISROOT
- REPLINSTMISMTCH
- REPLUPGRADEPRI
- REPLUPGRADESEC
- SEMWT2LONG
- SHMREMOVED

		<ul style="list-style-type: none"> • RSYNCSTRMVAL • SECNOTSUPPLEMENTARY • STRMNUMMISMTCH1 • STRMNUMMISMTCH2 • STRMSEQMISMTCH • SUPRCVRNEEDSSUPSRC • SRVLCKWT2LNG • STRMNUMIS • TRIGMODREGNOTRW • UPDSYNC2MTINS • UPDSYNCINSTFILE • ZDATEBADDATE • ZDATEBADTIME • ZTRIGNOTRW <p>The following messages were marked as deprecated (●).</p> <ul style="list-style-type: none"> • ACTLSTEXP • FMLLSTPRESENT <p>The following messages were changed:</p> <ul style="list-style-type: none"> • DBFLCORRP • JOBEXAMFAIL • DBCDBNOCERTIFY • LOCKSPACEFULL • NOREPLCTDREG • PRIMARYISROOT • REPLINSTMISMTCH • REPLUPGRADEPRI • REPLUPGRADESEC • SEMWT2LONG • SHMREMOVED
Revision 1	13 January 2012	Improved the description of UNIMPLOP.
Revision V5.4-002B	26 December 2011	<ul style="list-style-type: none"> • Conversion to documentation revision history reflecting GT.M releases with revision history for each chapter. • Marked deprecated messages as ●.

Table of Contents

- About This Manual vi
- 1. Error Messages 1
 - A. Standardized Error Codes 240
 - B. Reference Implementation Error messages 242

About This Manual

The GT.M Message and Recovery Procedures Reference Manual describes the meaning of messages issued by GT.M components, including the M compiler, the run-time system, and utility programs. It serves OpenVMS and UNIX versions of GT.M.

This manual is organized alphabetically according to the message identifier. The meaning of each message is provided as well as suggestions for locating and addressing the cause of error-type messages.

Intended Audience

This manual is for programmers and system managers who use GT.M.

Purpose of the Manual

The GT.M Message and Recovery Procedures Reference Manual helps you understand and act on GT.M messages. This manual complements the other GT.M manuals.

How to Use This Manual

The GT.M Message and Recovery Procedures Reference Manual is intended to be used primarily to determine the nature of a message and interpret its meaning. Therefore, Messages are listed in alphabetical order, according to the mnemonic that precedes them. Cross references to additional information are provided in individual entries, as appropriate. The manual does not include a table of contents or index.

Conventions Used in This Manual

References to other GT.M documents are implemented as relative hypertext links to PDF files in the same directory. These links work correctly in a browser that uses a plug-in component to display PDF files. For example, for Firefox you could use MozPluggger to plug in the Evince PDF reader. The links also work correctly offline if you download all related GT.M documents to the same local directory.

GT.M messages are identified by a signature of the form GTM-s-abcdef where -s- is a severity indicator and abcdef is an identifier. The severity indicators are: -I- for informational messages, -W- for warnings, -E- for errors and -F- for events that cause a GT.M process to terminate abnormally. For more information on monitoring GT.M messages, refer to "Appendix B: Monitoring GT.M Messages" in the Administration and Operations Guide.

Each entry in this manual is presented in the following format as illustrated by the NOPRINCIO message.

- | |
|---|
| <ol style="list-style-type: none">1. NOTPRINCIO2. NOTPRINCIO, Output currently directed to device xxxx3. Run Time Warning: This message displays the current device when the process enters Direct Mode and the current device xxxx is not the principal device.4. Action: To redirect all I/O to the terminal, note the current device or save it in a temporary variable and USE \$P. If you decide to resume program execution, restore the current device with a USE command. |
|---|

where

1. Indicates the unique mnemonic preceding the error message and is the component by which the entry is alphabetized.
2. Indicates the mnemonic and the actual message that accompanies it.
3. Indicates the GT.M component that generates the message, its severity, and a short description of its implication(s).
4. Suggests action(s) to take when the message appears.

This manual can be used with GT.M on any of its supported platforms. However, because in some instances the suggested actions are more useful when platform-specific information is provided, the following conventions are used, as necessary.

UNIX: The term UNIX is used here in the general sense of all platforms for which GT.M uses a POSIX API. As of this date, this includes: AIX; HP-UX on IA64 and PA-RISC; GNU/Linux on IA64, x86 and x86_64; Solaris on SPARC; z/OS.

Platform Identifier: If an entire statement is either UNIX- or OpenVMS- specific, it is preceded by the phrase "In UNIX," or "In OpenVMS,". The following example illustrates this convention. Action: In OpenVMS, relink your image to include the missing routine. In UNIX, relink your external call descriptor image to include the missing routine.

If a phrase or word is either UNIX- or OpenVMS- specific, it is suffixed by "(UNIX)" or "(VMS)". The following examples illustrates this convention.

Run Time Error: Strings longer than 32,767 (UNIX) or 65,535 (OpenVMS) bytes have been used with alternative collation and only gtm_ac_xform and gtm_ac_xback are defined in the collation library.

Compile Time Error: This indicates that translation of the indicated environment variable (UNIX) or logical name (OpenVMS) failed. Examine the accompanying message to find out the failure type and the reason of the error.

Although the terms "host shell command" and "file-specification" have some platform-specification connotations, they are used in their most generic sense throughout this manual. The former describes commands that originate from the host operating system, rather than from GT.M. The latter may refer to a simple file name or a full directory path to that file.

Henceforth, the term "originating instance" is used where previously "primary instance" or "originating primary instance" were used, and the term "replicating instance" is used where previously "secondary instance" and "originating secondary instance" were used. Since it is easier to change documentation than it is to change software, and given our tradition of maintaining compatibility especially for existing programs and scripts, the former terms will remain in the software for a long time to come, even as they are supplemented with the new terms in the code, and replaced in the documentation.

We have changed the terminology because "initiating" and "replicating" describe roles that currently exist and will continue to exist as GT.M evolves, while "primary" and "secondary" are both more and less precise and will become less useful as we add capabilities to GT.M. An "initiating instance" is always the instance that first records a transaction (including non-TP mini-transactions). A "replicating instance" is always following the action of an "originating instance", previously called "root primary". We have called secondaries that replicate propagating primaries, but they are not originating instances.

Following Up on Suggested Actions

When the suggested action is to "Report the error to the group responsible for database integrity at your operation," you may also refer to the *Maintaining Database Integrity* chapter in the Administration and Operations Guide.

For information about utility-generated messages, refer to the chapter that describes that utility in the Administration and Operations Guide.

The OpenVMS edition of GT.M uses the OpenVMS standard message facility as well as other OpenVMS system components. Therefore, some errors issued by GT.M originate in OpenVMS and are described in this manual.

MUPIP LOAD Errors

If a MUPIP LOAD error occurs, ensure that the proper media is loaded and that the command input includes the proper file-specification.

If the input file is FORMAT=GO or ZWR, the database should contain the correct content to the point where the failure occurred and should be usable. You can edit and possibly correct the input file.

If the input file is FORMAT=BIN, the database is probably corrupt. Fix the database integrity issues and EXTRACT the file again.

For more information on LOAD and EXTRACT, refer to the *MUPIP* chapter in the Administration and Operations Guide.

Plug-in Errors

GT.M's plug-in architecture allows you to choose your preferred encryption software. Some plugin errors that you may encounter are as follows:

Database file <path> not found

Plugin error: The plugin is unable to find the specified database file.

Action: Verify that the database file exists, the corresponding entry in the master key file points to the database file, and appropriate authorizations exist in the directory path and the database file.

Encryption handle corrupted

Plugin error: The plugin detected an internal error.

Action: This error indicates that there is a communication error between GT.M and the gtmcrypt plug-in. Replace the process with undamaged one. Report the entire incident context to your GT.M support channel.

Encryption key file <path> not found

Plugin error: The plugin was not able to find the key file on the specified path.

Action: Verify that the master key file entry for this key file points to the correct path. Verify that the key file itself exists. Verify proper authorizations on directory path and file.

Encryption library has not been initialized

Plugin error: A gtmcrypt function was called before gtmcrypt_init().

Action: Call gtmcrypt_init() before calling any other encryption functions.

For more information on the plug-in errors and their fixes, see the documentation of your preferred encryption software.

Appendix B: Reference Implementation Error messages lists some errors that the GT.M team encountered while testing GT.M's plug-in architecture with GNU Privacy Guard, the widely available implementation of Pretty Good Privacy (see "PGP: Pretty Good Privacy" by Simson Garfinkel).

MUPIP INTEG Errors

Database errors reported by MUPIP INTEG differ in impact and severity. Some require an immediate action to prevent extending the damage, action on other less severe errors may be delayed.

The following table outlines the MUPIP INTEG error messages with their severity using the codes as listed below:

About This Manual

A	Access: Prevents database access
B	Benign: Presents no risk of additional damage and has little or no effect on database performance
D	Dangerous: Presents a high risk that continuing updates may cause significant additional damage
I	Index: If the block is an index block, continuing updates will be quite dangerous: treat as a D; if the block is a datablock, continuing updates can only cause limited additional damage
T	Transient: Usually cleared by an update to the database

MUPIP INTEG Error Messages			
Error Name	Message	Severity	Section*
DBBADKYNM	Bad Kay Name	I	K1
DBBADNSUB	Bad numeric subscript	I	K1
DBBADPNTR	Bad pointer value in directory	D	K4
DBBDBALLOC	Block doubly allocated	D	K3
DBBFSTAT	Block busy/free status unknown (local bitmap corrupted)	D	M1
DBBNPNTR	Bit map block number as pointer	D	K4
DBBPLMGT2K	Blocks per local map is greater than 2K	D	I3
DBBPLMLT512	Blocks per local map is less than 512	D	I3
DBBPLNOT512	Blocks per local map is not 512	D	I3
DBBSIZZRO	Block size equals zero	A	I3
DBCOMPTOOLRG	Record has too large compression count	I	O2
DBFGTBC	File size larger than block count would indicate	B	I4
DBFSTBC	File size smaller than block count would indicate	D	I4
DBFSTHEAD	File smaller than database header	A	I3
DBGTDDBMAX	Key larger than database maximum	I	K7
DBHEADINV	Header size not valid for database	A	I3
DBINCLVL	Block at incorrect level	D	O1
DBINCRVER	Incorrect version of GT.M database	A	I2
DBINVGBL	Invalid mixing of global names	D	K3
DBKEYGTIND	Key greater than index key	I	K2
DBKGTALLW	Key larger than maximum allowed length	I	K1
DBLOCMBINC	Local bit map incorrect	B	M1

Section*: Refers to the specified section in *Managing Database Integrity* chapter of Administration and Operations Guide. The section details a description along with the action item to be taken on encountering the error message.

About This Manual

MUPIP INTEG Error Messages			
Error Name	Message	Severity	Section*
DBLRCINVSZ	Last record of block has invalid size	I	K5
DBLTSIBL	Key less than sibling's index key	I	K2
DBLVLINC	Local map block level incorrect	B	M2
DBMAXKEYEXC	Maximum key size for database exceeds design maximum	D	I3
DBMAXRSEXBL	Maximum record size for database exceeds what the block size can support	D	I3
DBMBMINCFREZ	Master bit map incorrectly asserts this local map has free space	B	M1
DBMBPFLDIS	Master bit map shows this map full, in disagreement with both disk and INTEG results	B	M1
DBMBPFLDLBM	Master bit map shows this map full, agreeing with disk local map	B	M1
DBMBPFLINT	Master bit map shows this map full, agreeing with MUPIP INTEG	B	M1
DBMBPFRDLBM	Master bit map shows this map has space, agreeing with disk local map	B	M1
DBMBPFRINT	Master bit map shows this map has space, agreeing with MUPIP INTEG	B	M1
DBMBPIN CFL	Master bit map incorrectly marks this local map full	B	M1
DBMBSIZMN	Map block too small	B	M2
DBMBSIZMX	Map block too large	B	M2
DBMBTNSIZMX	Map block transaction number too large	T	I6
DBMRKBUSY	Block incorrectly marked busy	B	M1
DBMRKFREE	Block incorrectly marked free	D	M1
DBMXRSEXC MIN	Maximum record size for database is less than the design minimum	D	I3
DBNOTDB	File does not have a valid GDS file header	A	I3
DBNOTMLTP	Block size not a multiple of 512 bytes	A	I3
DBRBNLBMN	Root block number is a local bit map number	D	K4
DBRBNNEG	Root block number negative	D	K4
DBRBNTOOLRG	Root block number greater than last block number in file	D	K4
DBREADBM	Read error on bitmap	D	H7

Section*: Refers to the specified section in *Managing Database Integrity* chapter of Administration and Operations Guide. The section details a description along with the action item to be taken on encountering the error message.

About This Manual

MUPIP INTEG Error Messages			
Error Name	Message	Severity	Section*
DBRLEVLTON	Root level less than one	D	O1
DBRLEVTOOHI	Root level higher than maximum	D	O1
DBSVBNMIN	Start VBN smaller than possible	A	I3
DBSZGT64K	Block size is greater than 64K	A	I3
DBTNNEQ	Current tn and early tn are not equal	T	I6
DBTNTTOOLG	Block transaction number too large	T	I6
DBTTLBLK0	Total blocks equal zero	A	I4
DBUNDACCMT	Cannot determine access method; trying with BG	T	I6
<p>Section*: Refers to the specified section in <i>Managing Database Integrity</i> chapter of Administration and Operations Guide. The section details a description along with the action item to be taken on encountering the error message.</p>			

Chapter 1. Error Messages

ABNCOMPTINC

ABNCOMPTINC, Deviceparameter xxxx and deviceparameter yyyy are not compatible in the zzzz command

Compile Time Error: The command specifies incompatible deviceparameters (e.g., specifying both FIXED and VARIABLE).

Action: Refer to the Chapter 9 Input Output Processing in the Programmer's Guide and modify the list.

ACOMPTBINC

ACOMPTBINC, Deviceparameter xxxx is compatible with only yyyy in the command zzzz

Run Time Error: An OPEN, USE, or CLOSE command specifies a deviceparameter that does not apply to the command.

Action: Look for deviceparameters that should be on other I/O commands. For example, the deviceparameter DELETE is valid on CLOSE but produces this error if it is applied to the USE command.

ACTIVATEFAIL

ACTIVATEFAIL , Failed to activate passive source server for secondary instance name xxxx

MUPIP Error: This error is issued by a mupip replic -source -activate command when it tries to activate a passive source server and at the same time switch the instance from being a secondary (propagating primary) to a root primary. It is possible to switch an instance from being a secondary to a root primary by activating an already running passive source server on that instance at the same time specifying the rootprimary qualifier. But there should be no process other than the one passive source server running on that instance. If not the above error is issued.

Action: Shutdown all processes other than the passive source server that are running on that replication instance (source servers, receiver server, update process, GT.M processes etc.) and are accessing the journal pool and then reissue the mupip replic -source -activate command

ACTLSTEXP ●

Last used version: **V5.4-002B**

ACTLSTEXP, Actuallist expected

Run Time Error: This indicates that the DO command, without an actuallist of parameters, failed because its entryref argument specified a label that has a formallist of parameters.

Action: Review the interface between the DO and the subroutine. Add the actuallist to the DO, remove the formallist from the label, or identify another subroutine to invoke, as appropriate.

ACTLSTTOOLONG

ACTLSTTOOLONG, More actual parameters than formal parameters: xxxx

Compile/Run Time Error: This indicates that the label xxxx with a formallist; is invoked from within a routine with a longer actualist (during compile-time). At run-time, a similar error can occur when a longer actualist is supplied by an invocation from another routine.

Action: Review the interface between the DO command and the subroutine. Modify the actualist, formallist, and/or label, as appropriate.

ACTOFFSET

ACTOFFSET, Actualist not allowed with offset

Compile Time Error: This indicates that a DO command or extrinsic specified an actualist and an entryref that includes an offset.

Action: Look for an inappropriate offset.

ACTRANGE

ACTRANGE, Alternate Collating Type xxxx is out of range

Run Time Error: The alternate collation sequence type does not fall in the expected range of 0 to 255.

Action: Define a new collation sequence type that has a value between 0 and 255 inclusive. For more information, refer the "Internationalization" chapter of the Programmer's Guide.

ADDRTOOLONG

ADDRTOOLONG, Socket address xxxx of length aaaa is longer than the maximum permissible length bbbb

Run Time Error: This indicates that the address value supplied with the CONNECT or LISTEN deviceparameters exceeds the maximum acceptable length.

Action: Examine the value and shorten the length of the address value.

ALIASEXPECTED

ALIASEXPECTED, Alias or alias container variable expected in this context

Compile time or Run Time Error: This indicates the argument for a SET * or KILL * command used a non-alias local variable where the syntax requires an alias or alias container variable.

Action: Correct the code in or investigate the logic to determine why the local variable in question is not in the expected state.

AMBISYIPARAM

AMBISYIPARAM, Parameter xxxx is ambiguous to \$ZGETSYI()

Run Time Error: This indicates that the argument xxxx is ambiguous to \$ZGETSYI() because it does not have enough characters.

Action: Add enough characters to make the argument unambiguous.

ANCOMPTINC

ANCOMPTINC, Deviceparameter xxxx is not compatible with any other deviceparameters in the yyyy command

Compile Time Error: This indicates that the specified deviceparameter can only be used by itself.

Action: Remove conflicting deviceparameters from the command.

ARROWNTDSP

ARROWNTDSP, Unable to display ^----- due to length of source line

Compile Time Error: Displayed instead of the arrow indicating where compilation error occurred due to a long line source line.

Action: Refer to the source code. the line number and column number in the associated messages identify the position of the problem. Consider shortening the line, at least until the error is found and corrected.

ASSERT

ASSERT, Assert failed xxxx line yyyy

Run Time Error: An internal GT.M consistency check failed. This error occurs only in some special versions of GT.M software.

Action: Report the entire incident context to your GT.M support channel.

BACKUPCTRL

BACKUPCTRL, Control Y or control C encountered during backup, aborting backup

MUPIP Warning: This indicates that BACKUP terminated because of an operator <CTRL>-C or <CTRL>-Y.

Action: Do not rely on the result of this BACKUP. If appropriate, investigate whether any of the BACKUP output files are complete and therefore potentially useable.

BACKUPKILLIP

BACKUPKILLIP, Kill in progress indicator is set for file ffff, backup database could have incorrectly marked busy integrity errors

MUPIP Warning: This indicates that one or more active process are performing KILL cleanup in database file ffff. Generally, BACKUP can wait for this to finish in order to get a consistent copy of the database. However, this indicates it waited several minute and is proceeding The resulting backup almost surely contains blocks incorrectly marked busy.

Action: Wait and perform the BACKUP when there are no large KILL operations triggering extensive cleanup. If this is not desirable, fix the errors in the backup copy (reported by an INTEG NOMAP) with DSE MAP FREE. If there are many such blocks you can edit the INTEG output to create a script to drive the DSE operations. Alternatively, if you can get standalone access, to the database you may use DSE MAP RESTORE. Do not use MAP RESTORE on an active database.

BADACCMTHD

BADACCMTHD, Invalid access method was specified, file not created

MUPIP Warning: This indicates that CREATE encountered an invalid access method for the dynamic segment in the current Global Directory, which is defined by the logical name GTM\$GBLDIR / environment variable gtmgbldir.

Action: Use the Global Directory Editor (GDE) to verify the access method for the Global Directory. Look for the use of GT.M components with different version numbers.

BADCASECODE

BADCASECODE, xxxx is not a valid case conversion code.

Run Time Error: The two-argument form of \$ZCONVERT() reports this error if the case conversion specifier (second argument) is not one of the valid codes (U,u,L,l,T and t).

Action: Choose a valid case designation code.

BADCHAR

BADCHAR, XXX is not a valid character in the YYY encoding form.

Runtime Error: GT.M triggers this error when it encounters a byte sequence that is not legal according to the given character set of the current device.

Action: Either handle the illegal byte sequence or disable the triggering of BADCHAR error by VIEW "NOBADCHAR" command. If the error is from a READ or WRITE command, consider performing the I/O in M-mode. For more information, refer to Programmer's Guide.

BADCHSET

BADCHSET, xxxx is not a valid character mapping in this context.

Run Time Error: When GT.M recognizes that the expr in ICHSET=expr or OCHSET=expr is not one of the supported character set names ("M", "UTF-8", "UTF-16", "UTF-16LE" or "UTF-16BE"), it reports this error. Note that not all modes are supported under all conditions.

Action: Choose the proper designation for a supported character set.

BADDBVER

BADDBVER, Incorrect database version: xxxx

Run Time Error: This indicates that the database version is not compatible with the current GT.M version.

Action: Upgrade the database. For more information, refer to the release notes for the current GT.M version and any intervening versions back to the last prior version used.

BADGBLSECVER

BADGBLSECVER, Global section xxxx does not match the current database version

Run Time Error: In attempting to startup a database file, GT.M encountered a shared memory section containing a database version older than the current database version.

Action: Do not attempt to access the same database files simultaneously with different versions of GT.M. Perform a MUPIP RUNDOWN with the prior version of database existing in the shared memory section. If needed, contact your system administrator for help.

BADREGION

BADREGION, Region is not BG, MM, or CM

LKE Error: The current global directory attempted to map a region with an access method other than those listed.

Action: Use LKE only with database mapped to one of the listed database access methods.

BADTAG

BADTAG Unable to use file ffff (CCSID tttt) with CCSID uuuu

Run Time Error: This z/OS specific error indicates the device or file ffff had a tag tttt incompatible with the tag uuuu implied by chset associated with OPEN command.

Action: Change the [i/o]chset to match the device or file, or use iconv to convert the file to an appropriate character set, or possibly use chtag to [re]tag the file.

BADGTMNETMSG

BADGTMNETMSG, Invalid message sent to GT.CM server, type: xxxx

GT.CM Server Error: The GT.CM Server received an invalid message. Possible causes include an undetected network error, a message originating from a process that is inappropriately intruding on the GT.CM environment, or a protocol failure in a legitimate process.

Action: Retry the action that resulted in the error notification. If the problem persists, contact the group responsible for database operations on your network.

BADJPIPARAM

BADJPIPARAM, xxxx is not a legal parameter for \$ZGETJPI()

Run Time Error: This indicates that the argument xxxx is not a valid keyword for \$ZGETJPI().

Action: Refer to the Programmer's Guide for correct keyword usage.

BADLKIPARAM

BADLKIPARAM, xxxx is not a legal parameter for \$ZGETLKI()

Run Time Error: This indicates that the argument xxxx is not a valid keyword for \$ZGETLKI().

Action: Refer to the Programmer's Guide for correct keyword usage.

BADQUAL

BADQUAL, Unrecognized qualifier: xxxx

Run Time Error: This indicates that a SET of \$ZROUTINES specified xxxx, which is an unknown qualifier.

Action: Use an accepted qualifier: SRC= or NOSRC.

BADSRVRNETMSG

BADSRVRNETMSG, Invalid message received from GT.CM server

Run Time Error: This indicates that a GT.M process received an invalid message. Possible causes include an undetected network error, a message originating from a process that is inappropriately intruding on the GT.CM environment, or a protocol failure in a legitimate process.

Action: Retry the action that resulted in the notification error. If the problem persists, contact the group responsible for database operations on your network. Stop and restart the server to attempt to resolve the problem.

BADSYIPARAM

BADSYIPARAM, xxxx is not a legal parameter to \$ZGETSYI()

Run Time Error: This indicates that the argument xxxx for \$ZGETSYI() is not a valid keyword.

Action: Refer to the Programmer's Guide for correct keyword usage.

BADTRNPARAM

BADTRNPARAM, xxxx is not a legal parameter to \$ZTRNLNM

Run Time Error: This indicates that the argument xxxx for \$ZTRNLNM() is not a valid keyword.

Action: Refer to the Programmer's Guide for correct keyword usage.

BCKUPBUFLUSH

BCKUPBUFLUSH, Unable to flush buffer for online backup

MUPIP Error: This indicates that the online BACKUP was unable to flush the buffer data to disk. The most likely cause is that MUPIP does not have write access to the database file.

Action: Run the BACKUP from a process with write authorization, or wait until other processes have completed the buffer flush. MUPIP sets the repair flag if there is a serious problem. Run a MUPIP INTEG using the FA[ST] qualifier to reset this flag. Report to your System Administrator.

BEGINST

BEGINST, Beginning LOAD at record number: xxxx

MUPIP Information: This indicates that the LOAD command with the FORMAT=BINARY qualifier started with record number xxxx.

BEGSEQGTENDSEQ

BEGSEQGTENDSEQ, Journal file xxxx has beginning sequence number aaaa greater than end sequence number bbbb

MUPIP Error: This indicates that the beginning sequence number aaaa of the journal file xxxx is greater than end sequence number bbbb.

Action: Report the entire incident context with appropriate log messages to your GT.M support channel.

BFRQUALREQ

BFRQUALREQ, The [NO]BEFORE qualifier is required for this command

MUPIP Error: Any MUPIP SET command with JOURNAL=ON must specify either BEFORE_IMAGE or NOBEFORE_IMAGE journaling.

Action: Add the argument and select either BEFORE_IMAGE or NOBEFORE_IMAGE journaling for the database in question.

BIGNOACL

BIGNOACL, Existing file found when BIGRECORD specified with UDF format but no GT.M ACE, perhaps lost during COPY

Run Time Error: An existing file with an RMS record format of undefined (UDF) was opened by an OPEN command with a BIGRECORD parameter but no Access Control Entry containing information GT.M needs to access the file. The ACE may have been lost when copying such a file with the OpenVMS DCL COPY command.

Action: BIGRECORD format files should always be copied using the OpenVMS DCL BACKUP command which preserves the Access Control Entries on files. If the record format (VARIABLE or FIXED) and RECORDSIZE specified when creating the file are known, the ACE can be restored using the following OpenVMS DCL command: SET SECURITY <filename.ext> / ACL=(APPLICATION,SIZE=20,FLAGS=%X0A01,ACCESS=%X00F60001,DATA=%X424D5447,<format>,<recordsize>) Replacing “<filename.ext>” by the name of the file, “<format>” by “2” if the format is VARIABLE or “1” if FIXED, and “<recordsize>” by the RECORDSIZE originally specified.

BINHDR

BINHDR, gggg Date: dddd TIME: tttt Extract Region Characteristics rrrr Blk Size: xxxx Rec Size: yyyy Key Size: kkkk

MUPIP Information: This message displays header information for a binary format file. gggg is the global loaded. dddd is the date on which the region was extracted. tttt is the time when the region was extracted. rrrr is the region that contains the global. xxxx is the block size specified for the region. yyyy is the record size specified for the block. kkkk is the key size specified for the record.

BITMAPSBAD

BITMAPSBAD, Database bit maps are incorrect

Run Time Error: This indicates that a database operation encountered a corrupt bit map.

Action: GT.M uses bit maps in database files to determine whether a block is free or in use. Report this database structure error to the group responsible for database integrity at your operation.

BKUPTMPFILOPEN

BKUPTMPFILOPEN, Open of backup temporary file aaaa failed

Run Time Error: When an online backup is in progress, a GT.M process doing updates to the database is saving away the pre-update images of the blocks it updates in a special backup area used to make sure the backups are consistent. Periodically, these blocks need to be flushed out to a temporary file and are flushed by the process needing the space to put its own changed blocks. This means every running process needs to have R/W access to the temporary file created by the backup. If the process cannot open the temporary file, this error is written to the operator log, the backup is flagged as having encountered an error and the process proceeds. So this error is only backup related. It is NOT an error in the process itself which proceeds as if backup were not running.

Action: Determine cause of why process could not open temporary file, fix, and restart backup.

BKUPTMPFILWRITE

BKUPTMPFILWRITE, Write to backup temporary file aaaa failed

Run Time Error: When an online backup is in progress, a GT.M process doing updates to the database is saving away the pre-update images of the blocks it updates in a special backup area used to make sure the backups are consistent. Periodically, these blocks need to be flushed out to a temporary file and are flushed by the process needing the space to put its own changed blocks. This means every running process needs to have R/W access to the temporary file created by the backup. If the database write generates an error, the BKUPTFWFAIL error is written to the operator log, the backup is flagged as having encountered an error and the process proceeds. So this error is only backup related. It is NOT an error in the process itself which proceeds as if backup were not running.

Action: Determine cause of why the write failed, fix, and restart backup.

BLKCNT

BLKCNT, Last LOAD Block/RMS Record number: xxxx

MUPIP Information: This indicates that a LOAD command with the FORMAT=BINARY qualifier completed at record number xxxx. Indicates the last block or record number successfully processed by MUPIP LOAD. This information may be useful for auditing the LOAD, or if a restart is required after a LOAD is stopped prematurely.

BLKCNTEDITFAIL

BLKCNTEDITFAIL, MUPIP recover or rollback failed to correct the block count field in the file header for file xxxx

GDE/DSE Information: This indicates that rollback should correct the block count field in the file header. This message is an informational message issued by Recovery/Rollback process and no action is required. This error is benign.

BLKSIZ512

BLKSIZ512, Block size xxxx rounds to yyyy

GDE/DSE Information: This indicates that an ADD, CHANGE, or TEMPLATE command defined the BLOCKSIZE qualifier equal to xxxx, which is not divisible by 512. GDE adjusted the block size to yyyy, which is the next largest multiple of 512.

Action: If yyyy is not acceptable, modify the BLOCKSIZE qualifier value so that it is divisible by 512.

BLKTOODEEP

BLKTOODEEP, Block level too deep

Compile Time Error: This indicates that the line starts with too many level-indicator delimiters (.) for the level of nesting associated with the argumentless DO on the previous line.

Action: Remove inappropriate level indicator(s).

BLKWRITERR

BLKWRITERR, Unable to queue disk write for block XXXX. Will keep trying.

Run Time Information: This indicates that the disk is offline or not working because of a hardware or software problem in the disk subsystem.

Action: Check the disk subsystem operation.

BOMMISMATCH

BOMMISMATCH, XXX Byte Order Marker found when YYY character set specified.

Run Time Error: A Byte Order Marker (BOM) for character set XXX was found at the beginning of a file specified as containing data in character set YYY.

Action: Specify the proper character set when opening the file. For UTF-16 data, specifying CHSET="UTF-16" will use the BOM to determine whether the data is Little Endian or Big Endian. If no BOM is found, GT.M assumes Big Endian.

BOVTMGTEOVTM

BOVTMGTEOVTM, Journal file xxxx has beginning timestamp aaaa greater than end timestamp bbbb

MUPIP Error: This indicates that the beginning time stamp aaaa of the journal file xxxx is greater than the ending timestamp bbbb. This could be due to something that changed the system time while journaling was going on.

Action: Changing system time during GT.M Run-time is not allowed. Contact your GT.M support channel for further assistance.

BOVTNGTEOVTN

BOVTNGTEOVTN, Journal file xxxx has beginning transaction yyyy which is greater than end transaction zzzz

MUPIP Error: This indicates that MUPIP JOURNAL command has found that journal file xxxx has beginning transaction yyyy which is greater than end transaction zzzz.

Action: Report the error with appropriate log messages to your GT.M support channel.

BREAK

BREAK, Break instruction encountered

Run Time Information: This indicates that GT.M encountered a BREAK command within a routine and entered Direct Mode.

Action: All commands entered at the Direct Mode prompt are compiled and executed as they are entered. To continue program execution, enter the ZCONTINUE command. This message can be suppressed using VIEW BREAKMSG value.

BREAKDEA

BREAKDEA, Break instruction encountered during Device error action

Run Time Information: This indicates that GT.M encountered a BREAK command within a device EXCEPTION string and entered Direct Mode.

Action: GT.M activates EXCEPTION strings for deviceparameters when a device reports an exception condition. All commands entered at the Direct Mode prompt are compiled and executed as they are entered. To continue program execution, enter the ZCONTINUE command. It is important to ensure the EXCEPTIONS in production code are thoroughly debugged.

BREAKZBA

BREAKZBA, Break instruction encountered during ZBREAK action

Run Time Information: This indicates that GT.M encountered a BREAK command within a ZBREAK action string and entered Direct Mode. The ZBREAK command sets or clears breakpoints during debugging. All commands entered at the Direct Mode prompt are compiled and executed as they are entered.

Action: To continue program execution, enter the ZCONTINUE command.

BREAKZST

BREAKZST, Break instruction encountered during ZSTEP action

Run Time Information: This indicates that GT.M encountered a BREAK command within a ZSTEP action string and entered Direct Mode.

Action: The ZSTEP command causes GT.M to proceed to the beginning of the next line of M code that matches the characteristic specified by the ZSTEP argument. GT.M compiles and executes all commands entered at the Direct Mode prompt as they are entered. To continue program execution, enter the ZCONTINUE command.

BTFAIL

BTFAIL, The database block table is corrupt; error type xxxx

Run Time Error: This indicates that a database operation failed because the tables of recently used blocks are damaged. GT.M uses the block tables to control and optimize database traffic.

Action: Report this database cache error to the group responsible for database integrity at your operation.

BOOLSIDEFFECT

BOOLSIDEFFECT, Extrinsic (\$\$), External call (\$&) or \$INCREMENT() with potential side effects in Boolean expression

Compile Time Warning: This optional message, accompanied by a line and column pointing to the issue, indicates a Boolean expression that contains a side effect in a term other than its first. By default, GT.M may skip evaluating such terms.

Action: Revise the code to your standards and use the VIEW (arguments [NO]FULL_BOOLEAN or FULLBOOL_WARN) command and / or the environment variable (gtm_boolean) to select the appropriate setting for GT.M handling of this construct.

BUFFLUFAILED

BUFFLUFAILED, Errors flushing buffers from uuuu for database file dddd

MUPIP Error: MUPIP or DSE (uuuu) could not flush the buffers for database file dddd completely. In the case of MUPIP, this typically means that some process is not releasing the critical section. In the case of DSE, this typically means there is some error in the global buffer cache which needs to be fixed.

Action: In the case of MUPIP, wait approximately 20 seconds and retry. In the case of DSE, try DSE CACHE RECOVER to fix the cache. If the error persists, report it to the group responsible for database integrity at your operation as soon as possible.

BUFOWNERSTUCK

BUFOWNERSTUCK, PID xxxx waiting for PID yyyy to finish disk read of block zzzz. Been waiting for aaaa minutes.

Run Time Warning: Poor response time from the I/O subsystem could cause this error, or if the process performing the disk-read was suspended.

Action: If the disk-reading process is found to be in a process-suspended state, un-suspend it. If it is not suspended, examine the I/O subsystem performance characteristics for behavior problems. Report the full error message, along with the GT.M operator log messages during the specific timeframe, to your GT.M support channel.

BUFRDTIMEOUT

BUFRDTIMEOUT, Pid xxxx timed out waiting for buffered read by process yyyy to complete in database file zzzz

Run Time Information: This indicates that a process requiring buffer transfer began but did not complete. The system cancelled the process.

Action: The failed process must be run again. Other errors may appear in the operators log with this one to indicate why the process failed.

BUFSIZIS

BUFSIZIS, Journal Buffer size is xxxx

GDE Information: This message reports xxxx as the size of the journal buffer.

Action: Review the accompanying message(s) for additional information.

BUFTOOSMALL

BUFTOOSMALL, But block size xxxx requires buffer size yyyy

GDE Information: This indicates that an ADD, CHANGE, or TEMPLATE command specified an xxxx argument definition for the BUFFER_SIZE qualifier, which is incompatible with the definition of the BLOCKSIZE. yyyy specifies the minimum buffer size that can support this block size.

Action: Modify the block size and/or buffer size so they are compatible. Review the accompanying message(s) for the buffer size.

CALLERID

CALLERID, Routine xxxx called from yyyy

Run Time Information: This message provides the error location. Typically this error displays with other errors.

Action: Review the accompanying message(s) for additional information. Include this information if reporting the error to your GT.M support channel.

CALLINAFTERXIT

CALLINAFTERXIT, After a gtm_exit, a process can never create a valid GT.M context

Runtime Error: Once a call-in has done a call to gtm_exit(), a process can no longer do GT.M call-ins

Action: Either move or remove the inappropriate call-ins or move the gtm_exit call to a later point.

CEBIGSKIP

CEBIGSKIP, Compiler escape user routine skip count is too large

Compile/Run Time Error: This indicates that the skip count exceeded the maximum line length.

Action: Verify the user-supplied escape-handling routine(s).

CENOINDIR

CENOINDIR, Indirection type information not available for compiler escape feature

Compile/Run Time Error: This indicates that GT.M does not currently support this feature.

CETOOLONG

CETOOLONG, Compiler escape substitution exceeds maximum line size

Compile/Run Time Error: This indicates that the length of the substitution string exceeds the maximum line length.

Action: Determine whether the input source file can be modified to have less source on the line in question. If this is not possible, modify the substitution mechanism design or implementation.

CETOOMANY

CETOOMANY, Too many compiler escape substitutions in a single statement

Compile/Run Time Error: This indicates that the program being compiled contained more nested substitutions than allowed; that is 1024.

Action: Reduce the number of substitutions to less than 1024.

CEUSRERROR

CEUSRERROR, Compiler escape user routine returned error code xxxx

Compile/Run Time Error: This indicates that the compiler encountered the error specified by error code xxxx.

Action: Use the error code as the function argument of \$ZMESSAGE() to determine the text associated with the error and the appropriate corrective action.

CHNGTPRSLVTM

CHNGTPRSLVTM, Mupip will change tp_resolve_time from xxxx to yyyy because expected EPOCH or EOF record was not found in Journal File zzzz.

MUPIP Information: At startup, backward recovery/rollback internally computes a time called the tp_resolve_time which is until when backward processing will be performed across journal files of all regions. During backward processing it is possible in very rare cases that recovery does not see an EPOCH record or an EOF record as the last record in the journal file of those regions that had not been updated for quite a long time. In such cases recovery reduces the tp_resolve_time further by taking into account the timestamp of the last journal record. This effectively causes further backward processing but is necessary for a clean recovery. A CHNGTPRSLVTM message is printed whenever such journal files are encountered by backward recovery.

Action: None necessary

CIDIRECTIVE

CIDIRECTIVE, Invalid directive parameter passing. Expected I, O or IO.

Syntax Error: This indicates that a missing directive or syntactically invalid directive found for the parameter, pointed to by the previous messages EXTSRCLIN and EXTSRCLOC.

Action: One of the directives I, O or IO should be specified for the parameter in the entry displayed.

CIENTNAME

CIENTNAME, No label reference found for this entry in call-in table

Call In/Syntax Error: This indicates that a label reference to the M routine is missing or syntactically invalid for an entry in the call-in table (specified by GTMCI environment variable)

Action: Correct the syntax errors in the call-in table entry, at the location pointed to by the two previous messages: EXTSRCLIN and EXTSRCLOC, displaying the line and the column number respectively.

Make sure a valid M label reference is bound to the C call-name specified for this entry.

CIMAXLEVELS

CIMAXLEVELS, Too many nested Call-ins. Nested resources exhausted at level !UL.>/error/fao=1!/ansi=0

Run Time/Call-in Error: This indicates that GT.M runs out of its internal condition handlers stack due to too many levels of nested call-ins.

Action: Ensure that the call-in application is not nested more than the limit (xxxx) that GT.M supports. The number of nested call-ins can be reduced by not using call-ins, wherever possible, from external call functions.

CIMAXPARAM

CIMAXPARAM, Exceeded maximum number of parameters in the call-in table entry. An M routine cannot accept more than 32 parameters.

Run Time/Call-in Error: This indicates that the call-in table specified by \$GTMCI contains more than 32 parameters. Since an M formallist can only accept up-to 32 parameters, user cannot pass more than 32 arguments to gtm_ci(), excluding <c-call-name> and <ret-type>.

Action: Reduce the number of parameters to be less than 32, in the call-in table as well in the M routine.

CINOENTRY

CINOENTRY, No entry specified for xxxx in the call-in table

Run Time Error: This indicates that the call-name invoked by the C program does not have a corresponding entry in the call-in table specified by GTMCI environment variable.

Action: Add an entry to the call-in table for the call-name. Refer to the External Calls chapter in Programmer's Guide.

CIPARTYPE

CIPARTYPE, Invalid type specification for O/IO directive - expected pointer type

Call In/Syntax Error: This indicates that non-pointer types specified for the parameters to be passed by output-only (O) and input-output (IO) convention.

Action: Make sure one of the valid pointer types is specified for O and IO parameters. Refer to the External Calls chapter in Programmer's Guide.

CIRCALLNAME

CIRCALLNAME, Call-in routine name expected but not found

Call In/Syntax Error: This indicates that a call-name, which is to be bound to an M routine, is either missing or syntactically invalid for an entry in the call-in table file.

Action: Make sure a valid call-name is specified in the call-in table entry, at the location pointed to by the two previous messages: EXTSRCLIN and EXTSRCLOC, displaying the line and the column number respectively.

CIRPARAMNAME

CIRPARAMNAME, Invalid parameter specification for call-in table

Call In/Syntax Error: This indicates that a syntax error was found in parameter specification in the call-in table.

Action: Correct the syntax errors and make sure the parameters are correctly specified. Refer to the External Calls chapter in Programmer's Guide.

CIRTNTYP

CIRTNTYP, Invalid return type

Call In/Syntax Error: This indicates that the return type specified in the call-in entry is either missing or found invalid.

Action: Correct the return type syntax errors in the call-in table entry, at the location pointed to by the two previous messages: EXTSRCLIN and EXTSRCLOC, displaying the line and the column number respectively. Make sure a valid return type is specified for this entry.

CITABENV

CITABENV, Environment variable for call-in table xxxx not set

Call In/Run Time Error: This indicates that the environment variable GTMCI is not defined when an external C routine is about to call an M routine through GT.M call-in mechanism

Action: Check if GTMCI is defined to a valid file path to a call-in table. The call-in table file should contain a list of entries, each entry describing the parameter types, their passing convention of each M routine and its binding to a C routine.

CITABOPN

CITABOPN, Unable to open call-in table: xxxx

Call In/Run Time Error: This indicates that the call-in table defined by the environment variable GTMCI could not be opened.

Action: Check if the file path specified by GTMCI is correct and has at least read permissions for the user running GT.M. Check for secondary message(s) accompanying this error.

CITPNESTED

CITPNESTED, Call-ins can not be used inside a TP transaction

Run Time/Call-in Error: This indicates that a nested call-in (`gtm_ci()`) was invoked from an external call function that was called within a TSTART/TCOMMIT fence.

Action: GT.M currently does not handle TP support across multiple call-in invocations. Make sure all external call C functions, that invoke call-in functionality are not fenced within a TSTART/TCOMMIT boundar

CIUNTYPE

CIUNTYPE, Unknown parameter type encountered

Call In/Syntax Error: This indicates that missing or invalid parameter type specified for the entry displayed by the previous messages EXTSRCLIN and EXTSRCLOC.

Action: Make sure one of the valid parameter types is specified for the parameter in the entry displayed. Refer to the External Calls chapter in Programmer's Guide.

CLIERR

CLIERR, xxxx

Run Time Error: This indicates that an invalid command has been entered. xxxx provides further detail to the invalid command entered.

Action: Review the error and enter valid command.

CLOSEFAIL

CLOSEFAIL, Error while closing file descriptor dddd

Run Time Error: GT.M records this error in the syslog whenever it attempts to close an open file descriptor dddd and the close returns with an error. After recording this error, the GT.M process resumes normal operation.

Action: Report the above error message along with the accompanying GTM-I-CALLERID message to your GT.M support channel, as it may be a symptom of out-of-design operation.

CLSTCONFLICT

CLSTCONFLICT, Cluster conflict opening database file xxxx; could not secure access. Already open on node yyyy.

Run Time Error: This indicates that the process attempted to access non-clustered database xxxx, which was opened by node yyyy.

Action: Review the accompanying message(s) for additional information. Move the process to the appropriate node or use GDE to change the mapping in the Global Directory.

This error message can also occur after a database is improperly shut down. Perform a MUIPIP RUNDOWN with a FILE or REGION qualifier.

CMD

CMD, Command expected but not found

Compile Time Error: This indicates that GT.M encountered something other than a command- the next valid syntax element. This error can occur when there is an invalid character in the middle of a variable name or keyword, such as in the M line S X=Y_\$B or in the command W "this is a tab <TAB>".

Action: Verify the line syntax. Replace the line if it contains invisible (non-graphic) characters because diagnosing the line syntax may prove time consuming.

CMEXCDASTLM

CMEXCDASTLM, Exceeded AST limit. Cannot open database.

GT.CM Server Error: This indicates that the GT.CM server exceeded its quota of asynchronous system traps (ASTs).

Action: Increase the ASTLIM for the GT.CM server by modifying GTCMSTART.COM, shut down and restart the GT.CM server. Review SYSGEN factors and user authorizations for AST limits.

CMICHECK

CMICHECK, Internal CMI error. Report to GT.M Support.

GT.CM Server Error: This indicates that the GT.CM and DECNET cannot communicate properly. See GTMCHECK.

Action: Report the entire incident context to your GT.M support channel.

CMINTQUE

CMINTQUE, Interlock failure accessing GT.CM server queue

GT.CM Server Error: This indicates that interlock cannot move data onto the queue interlock bit.

Action: Accompanying messages should indicate the hardware or software problem, which caused the interlock failure.

CMSYSSRV

CMSYSSRV, Error doing system service, status:

GT.CM Server Error: This indicates that the GT.CM server could not successfully perform some system service. This message is followed by a secondary error message that describes the nature of the failure.

Action: Retry the action that resulted in the error. If the problem persists, contact the group responsible for database operations on your network.

CNOTONSYS

CNOTONSYS, command is not supported by this operating system

Compile Time Error: This indicates that the operating system does not support the command.

Action: Check the operating system documentation for a supported command.

COLLARGLONG

COLLARGLONG, Collation sequence nnn does not contain routines for long strings

Run Time Error: Strings longer than 32,767 (UNIX) or 65,535 (OpenVMS) bytes have been used with alternative collation and only gtm_ac_xform and gtm_ac_xback are defined in the collation library.

Define gtm_ac_xfrom_1 and gtm_ac_xback_1 routines in the collation library.

COLLATIONUNDEF

COLLATIONUNDEF, Collation type xxxx is not defined

Run Time Error: This indicates that an attempt was made to reference a collation sequence that is not available.

Action: Ensure that the environment variable GTM_COLLATE_n in UNIX and logical name gtm_collate_n in OpenVMS is properly defined, where n is the identification number of the failing collation type. Ensure that the file referencing GTM_COLLATE_n / gtm_collate_n is available to the process. Use host shell commands to verify its location and protections. Examine the executable (and/or the objects that comprise it) to determine whether it has the proper entry points. For more information, refer to the "Internationalization" chapter of the Programmer's Guide.

COLLDATAEXISTS

COLLDATAEXISTS, Collation type cannot be changed while data exists

Run Time Error: This indicates that an attempt was made to change the local collation type while local variables exist.

Action: KILL or NEW the local variables before you change the local collation type.

COLLFNMISSING

COLLFNMISSING, Routine xxx is not found for collation sequence nnn

Run Time Error: Required transformation back routine is missing in collation library.

Action: If gtm_ac_xfrom_1 is defined, also define gtm_ac_xback_1. Or if gtm_ac_xform is defined, also define gtm_ac_xback in the collation library.

COLLTYPVERSION

COLLTYPVERSION, Collation type xxxx, version yyyy mismatch

Run Time Error: This indicates that the user image with collation type xxxx does not accept the version of collation for an existing global. yyyy is the version associated with the global.

Action: Review the implementation history of the current collation algorithm. Modify or replace the image, if appropriate. If the version should change, temporarily RESTORE the older matching module, unLOAD and KILL any globals using the older algorithms, RESTORE the new algorithms, and reLOAD the global.

COLON

COLON, Colon (:) expected in this context

Compile Time Error: This indicates that GT.M did not encounter a colon where expected.

Action: Look for a \$SELECT() function that does not have a colon separating the conditional expression from its corresponding value expression. Also, look for a ZGOTO that is missing a colon between the level and an entry reference.

COMMA

COMMA, comma expected in this context

Compile Time Error: This indicates that GT.M did not encounter a comma where expected.

Action: Look for a missing argument or comma in a function that requires multiple arguments.

COMMAORRPAREXP

COMMAORRPAREXP, Comma or right parenthesis expected but not found

Compile Time Error: This indicates that GT.M did not encounter a comma or right parenthesis where expected.

Action: Look for a list of improperly formatted subscripts, arguments, or parameters.

COMMENT

COMMENT, Comment line. Placed zbreak at next executable line.

Run Time Information: This indicates that a ZBREAK specified a line that had no active code. Therefore, GT.M set the ZBREAK at the next line containing source code.

COMMITWAITPID

COMMITWAITPID, Pid *www* waited *ttt* minute(s) for pid *pppp* to finish commits in database file *dddd*

Run Time Warning: This warning message in the operator log indicates the total amount of time that the process *www* waited for another process *pppp* to finish database transaction commit. If the `$gtm_proctuckexec` mechanism is enabled, this message invokes it. If a process waits for more than one process to finish database transaction commits, it issues this message for each one it encounters.

Action: If the process *pppp* is still running, get a C-stack trace of the process (using a debugger) and report to your GT.M support channel with system log and operator log information.

COMMITWAITSTUCK

COMMITWAITSTUCK, Pid *www* timed out after waiting *ttt* minute(s) for *nnnn* concurrent GT.M process(es) to finish commits in database file *dddd*

Run Time Error: This error message indicates that a process could not finish database transaction commit and timed out waiting for other concurrent processes to finish.

Action: Check the operator log for accompanying COMMITWAITPID messages. Every concurrent GT.M processes reported with the COMMITWAITSTUCK messages would have accompanying COMMITWAITPID message(s). If so, review those messages. If not, report to your GT.M support channel with system log and operator log information.

COMPILEQUALS

COMPILEQUALS, Error in compiler qualifiers: *xxxx*

Error Messages

Compile/Run Time Error: This indicates that a run-time compilation specified an invalid qualifier (xxxx). Qualifiers for a run-time compilation can be specified with ZLINK or by setting \$ZCOMPILE to a qualifier string that GT.M uses for auto-ZLINKs with no qualifiers.

Action: Review the qualifiers in the ZLINK sub-argument or those being SET into \$ZCOMPILE.

COREINPROGRESS

COREINPROGRESS, Previous core attempt failed; core generation bypassed.

Run Time Error: This indicates that the process, which failed, was unable to create a memory dump file and tried to create another one.

Action: Report the entire incident context to your GT.M support channel for further analysis.

CORRUPT

CORRUPT, Corrupt input in Blk #xxxx, Key #yyyy; resuming with next global block

MUPIP Warning: This indicates that LOAD encountered bad data in the input file.

Action: Refer to the topic MUPIP LOAD Errors in "About This Manual" section.

CPBEYALLOC

CPBEYALLOC, Attempt to copy beyond the allocated buffer

DSE Time Error: This indicates that DSE tried to add a record that did not fit into the block size and/or the balanced tree structure.

Action: Check the block size. Move or create the record in a different location.

CRITRESET

CRITRESET, The critical section crash count for region xxxx has been incremented

Run Time Fatal Error: This indicates that the critical section for region xxxx was reset by the DSE command CRITICAL with the qualifiers INIT and RESET, while this process was accessing that database.

Action: Wait until the DSE repair operations are complete before retrying or restarting the process.

CRITSEMFAIL

CRITSEMFAIL, Error with semaphores for region xxxx

Run Time Error: This indicates that GT.M encountered a missing or damaged semaphore. This typically indicates that an agent external to GT.M has deleted or modified the semaphores GT.M uses to manage database and LOCK interactions.

Action: Investigate the state of the GT.M semaphores and all previous actions that might have damaged them.

CRYPTDLNOOPEN

CRYPTDLNOOPEN, Error loading encryption library. xxxx

Run Time Error: GT.M failed to load the gtmcrypt plug-in or one of its related libraries.

Action: Refer to the accompanying detail (xxxx) and verify that the gtmcrypt plug-in and related libraries are properly installed and that \$LD_LIBRARY_PATH and \$LIBPATH are properly set.

CRYPTHASHGENFAILED

CRYPTHASHGENFAILED, Error generating encryption hash. xxxx

Run Time Error: gtmcrypt plug-in reports there is problem with the hash function.

Action: Examine the message (xxxx) from the plug-in and take the appropriate action.

CRYPTINIT

CRYPTINIT, Error initializing encryption library. xxxx

Run Time Error: The gtmcrypt plug-in reports it is unable to initialize one or more of its related libraries.

Action: Examine the detailed message (xxxx) from the plug-in and take appropriate action.

CRYPTJNLWRONGHASH

CRYPTJNLWRONGHASH, Encryption key hash mismatch between journal file jjjj and corresponding database file dddd

Run Time Error: gtmcrypt plug-in reports the hash of the key in the header of database file dddd does not match the hash stored in the header of journal file jjjj. This is most likely caused by inappropriate operator action such as replacing the current journal file with an older journal file.

Action: Correct the error that caused the incorrect journal file to be pointed to by the database file. If the correct journal file has been inadvertently deleted, create new journal files with the -noprevjnl switch. Take a backup as soon as possible thereafter. Depending on your situation, you may need to refresh secondary instances.

CRYPTKEYFETCHFAILED

CRYPTKEYFETCHFAILED, Cannot obtain encryption key for ffff. xxxx

Run Time Error: gtmcrypt plug-in reports it was unable to obtain an encryption key for file ffff.

Action: Examine the message (xxxx) from the plug-in and take the needed action: for example, verify encryption key for this file is pointed to by the database key file, verify proper permissions on the directory path and the file, and so on. Also, make sure that there is an appropriate maximum process limit because obtaining an encryption key may fork other processes.

CRYPTKEYFETCHFAILEDNF

CRYPTKEYFETCHFAILEDNF, Cannot obtain encryption key. xxxx

Run Time Error: gtmcrypt plug-in reports it was unable to obtain an encryption key based upon matching the hash of an encryption key.

Action: Examine the message (xxxx) from the plug-in and take the needed action: for example, verify encryption keys for all database files are pointed to by the database key file. For extracts and backups, verify all the keys from the databases that provided records are in the database key file.

CRYPTNOMM

CRYPTNOMM, is an encrypted database. Cannot support MM access method.

MUPIP or GDE error: This error is triggered by an attempt to mark an MM database as encrypted with GDE or to switch an encrypted database from BG to MM with MUPIP SET. The MM access method is not supported for encrypted databases.

Action: Use the BG access method for encrypted files.

CRYPTNOPSWDINTP

CRYPTNOPSWDINTP, Cannot prompt for password inside a TP transaction.

Run-time error: This error occurs if the process used an external call to set the gtm_passwd environment variable to null string after startup and then accessed an encrypted database file for the first time within a TP transaction.

Action: If possible, set the gtm_passwd environment variable to the obfuscated password. Otherwise, revise the logic to touch a global mapped to each encrypted database, for example, \$DATA(myglobal), to ensure the prompting happens before entering any TP transaction.

CRYPTNOV4

CRYPTNOV4, ffff is an encrypted database. Cannot downgrade(to V4) with Encryption option enabled.>/error/fao=2!/ansi=0

MUPIP error: An attempt to downgrade ffff which is an encrypted database to the V4 (GT.M version 4) format failed because the V4 format does not support encrypted database files.

Action: Use the database in the current format. If a V4 format is required, extract the data in unencrypted ZWRite format with MUPIP EXTRACT and load it into a newly created V4 database.

CRYPTOPFAILED

CRYPTOPFAILED, Encrypt/Decrypt operation failed. xxxx

Run Time Error: gtmcrypt plug-in reports there is problem with encryption or decryption.

Action: Examine the message (xxxx) from the plug-in and take appropriate action.

CTLMNEMAXLEN

CTLMNEMAXLEN, The maximum length of a control mnemonic has been exceeded

Run Time Error: This indicates that GT.M encountered a controlmnemonic that exceeds the supported maximum length.

Action: Modify the control mnemonic so that it does not exceed the permitted length.

CTLMNEXPECTED

CTLMNEXPECTED, Control mnemonic is expected in this context

Run Time Error: This indicates that GT.M requires a control mnemonic in this context.

Action: Modify the spelling of the control mnemonic. Refer to the Input Output Processing chapter in the GT.M Programmer's Manual.

CTRAP

CTRAP, Character trap \$C(xxxx) encountered

Run Time Error: This indicates that the current device encountered character xxxx in its input stream while xxxx was defined as an exception by a CTRAP deviceparameter.

Action: Determine why this character was defined as an error or why an EXCEPTION string was not defined to address it.

CTRLC

CTRLC, CTRL_C encountered

Run Time Information: This indicates that the principal device encountered a <CTRL>-C in its input stream and GT.M put the process into Direct Mode.

Action: GT.M compiles and executes all commands entered at the Direct Mode prompt as they are entered. To continue program execution, enter the ZCONTINUE command. Response to <CTRL>-C is controlled with the [NO]CENABLE and CTRAP = deviceparameters.

CTRLY

CTRLY, User interrupt encountered

Run Time Information: This indicates that the principal device encountered a <CTRL>-Y in its input stream, and GT.M gave control to the OpenVMS CLI. The normal CLI is DCL.

Action: You can resume operation with a CONTINUE command if your actions do not invoke other images. For more information on commands that invoke images, refer to the OpenVMS DCL Dictionary.

CURRSOCKOFR

CURRSOCKOFR, Current socket of index xxxx is out of range. There are only yyyy sockets.

Run Time Error: This indicates that an OPEN, USE, READ or WRITE attempted to select a socket outside the range of available sockets.

Action: Review the socket management logic and revise to use only available sockets.

DBADDRALIGN

DBADDRALIGN, Database file *xxxx*, element location *aaaa*: blk = *bbbb*: [*yyyy*] control *cccc* was unaligned relative to base *dddd* and element size *eeee*

Run Time Information: This indicates that a control structure in the database cache is damaged.

Action: None needed. GT.M fixes this error as part of cache recovery, which follows the cache verification. If this message shows up frequently or is reproducible, contact your GT.M support channel.

DBADDRANGE

DBADDRANGE, Database file *xxxx*, element location *yyyy*: control *zzzz* was outside *aaaa* range *bbbb* to *cccc*

Run Time Information: This indicates that a process was abnormally terminated. Database control structures may be damaged.

Action: GT.M often fixes this error unless there is a serious problem causing this error. If there is a serious problem, the accompanying messages identify the cause.

DBADDRANGE8

DBADDRANGE8, Database file *xxxx*, element location *yyyy*: control *zzzz* was outside *aaaa* range *bbbb* to *cccc*.

Run time Error: This message is the same as a DBADDRANGE message except that *bbb8* and *ccc8* are 8-byte quantities (as opposed to 4-byte quantities in DBADDRANGE).

Action: GT.M often fixes this error unless there is a serious problem causing this error. If there is a serious problem, the accompanying messages identify the cause.

DBBADFREEBLKCTR

DBBADFREEBLKCTR, Database *xxxx* free blocks counter in file header: *yyyy* is incorrect; should be *zzzz*. Auto-corrected.

Run Time Information: This indicates that the free-blocks counter in the database fileheader is incorrect. The correct value is indicated in the message and the free-blocks counter is reset. This correction only occurs when GT.M performs a database file extension.

DBBADKYNM

DBBADKYNM, *xxxx* is an invalid key name

MUPIP Error: This is a MUPIP INTEG error. Refer to the topic MUPIP INTEG Errors in About This Manual section of this manual.

DBBADNSUB

DBBADNSUB, *xxxx* Bad numeric subscript

MUPIP Error: This is a MUPIP INTEG error. Refer to the topic MUPIP INTEG Errors in About This Manual section of this manual.

DBBADPNTR

DBBADPNTR, xxxx Bad pointer value in directory

MUPIP Error: This is a MUPIP INTEG error. Refer to the topic MUPIP INTEG Errors in About This Manual section of this manual.

DBBADUPGRDSTATE

DBBADUPGRDSTATE, Correcting conflicting values for fields describing database version upgrade state in the file header for region rrrr (ffff) - make fresh backups with new journal files immediately.

Run Time Warning: This warning message in the operator log indicates region rrrr (file ffff) had an out-of-design combination of database upgrade conditions, which may have caused defective journal files and -online BACKUPS. GT.M automatically corrects this condition, but you should investigate the possible causes for such file header damage.

Action: Make fresh backups with new journal files immediately.

DBBDBALLOC

DBBDBALLOC, xxxx Block doubly allocated

MUPIP Error: This is a MUPIP INTEG error. Refer to the topic MUPIP INTEG Errors in About This Manual section of this manual.

DBBFSTAT

DBBFSTAT, xxxx Block busy/free status unknown (local bitmap corrupted)

MUPIP Error: This is a MUPIP INTEG error. Refer to the topic MUPIP INTEG Errors in About This Manual section of this manual.

DBBLEVMN

DBBLEVMN, xxxx Block level less than zero

Run Time Information: This indicates that a database operation failed. The level specified for block xxxx is less than zero.

Action: Report this database structure error to the group responsible for database integrity at your operation.

DBBLEVMX

DBBLEVMX, xxxx Block level higher than maximum

Run Time Information: This indicates that a database operation failed. The level specified for block xxxx exceeds the maximum allowed during a block certification or a DSE integrity check.

Action: Report this database structure error to the group responsible for database integrity at your operation.

DBBMBARE

DBBMBARE, xxxx Bit map does not protect itself

Run Time Information: This indicates that a bitmap error was encountered during a block certification or a DSE integrity check.

Action: Report this database structure error to the group responsible for database integrity at your operation.

DBBMINV

DBBMINV, xxxx Bit map contains an invalid pattern

Run Time Information: This indicates that a bitmap error was encountered during a block certification or a DSE integrity check.

Action: Report this database structure error to the group responsible for database integrity at your operation.

DBBMLCORRUPT

DBBMLCORRUPT, Database xxxx: Bitmap blk yyyy is corrupt (Size = aaaa, lev1 = bbbb, tn = cccc: Dbtn = dddd): Database integrity errors likely

Run Time Error: This indicates that a local bitmap block was found corrupted in the global buffers. In the event that the message is followed by a GTMASSERT in the operator log, a dump-file/core may also be produced.

Action: MUPIP RUNDOWN the indicated database and check for integrity errors, check for bitmap related errors and fix them before resuming operations on the database. Report to your GT.M support channel with the operator log information and dump-cores/files, if any.

DBBMMSTR

DBBMMSTR, xxxx Bit map does not match master map

Run Time Information: This indicates that a bitmap error was encountered during a block certification or a DSE integrity check.

Action: Report this database structure error to the group responsible for database integrity at your operation.

DBBMSIZE

DBBMSIZE, xxxx Bit map has incorrect size

Run Time Information: This indicates that a bitmap error was encountered during a block certification or a DSE integrity check.

Action: Report this database structure error to the group responsible for database integrity at your operation.

DBBNPNTR

DBBNPNTR, Bit map block number as pointer

MUPIP Error: This is a MUPIP INTEG error. Refer to the topic MUPIP INTEG Errors in About This Manual section of this manual.

DBBPLMGT2K

DBBPLMGT2K, Blocks per local map is greater than 2k

MUPIP Error: This is a MUPIP INTEG error. Refer to the topic MUPIP INTEG Errors in About This Manual section of this manual.

DBBPLMLT512

DBBPLMLT512, Blocks per local map is less than 512

MUPIP Error: This is a MUPIP INTEG error. Refer to the topic MUPIP INTEG Errors in About This Manual section of this manual.

DBBPLNOT512

DBBPLNOT512, Blocks per local map is not 512

MUPIP Error: This is a MUPIP INTEG error. Refer to the topic MUPIP INTEG Errors in About This Manual section of this manual.

DBBSIZMN

DBBSIZMN, xxxx Block too small

Run Time Warning: This indicates that during block certification or DSE/MUPIP integrity check the size of block xxxx was found to be less than the minimum allowed.

Action: Report this database structure error to the group responsible for database integrity at your operation.

DBBSIZMX

DBBSIZMX, xxxx Block larger than file block size

Run Time Warning: This indicates that during block certification or DSE/MUPIP integrity check the size of block xxxx was found to exceed the block size for the database region.

Action: Report this database structure error to the group responsible for database integrity at your operation.

DBBSIZZRO

DBBSIZZRO, Block size equals zero

MUPIP Error: This is a MUPIP INTEG error. Refer to the topic MUPIP INTEG Errors in About This Manual section of this manual.

DBBTUFIXED

DBBTUFIXED, The blocks-to-upgrade file-header field has been changed to the correct value

MUPIP Information: MUPIP INTEG has corrected the blocks-to-upgrade field.

Action: Report this to the group responsible for database integrity at your operation.

DBBTUWRNG

DBBTUWRNG, The blocks-to-upgrade file-header field is incorrect. Expected xxxx, found yyyy

MUPIP INTEG Error: The "Blocks to Upgrade" counter was found to be incorrect by MUPIP INTEG (this is only checked for non-FAST integ).

Action: If there are no other integrity errors, MUPIP INTEG will repair the counter. If there are other integrity errors, fix those errors first, then rerun MUPIP INTEG which will repair the counter if it is still found to be in error. Although this error is not indicative of any specific kind of database damage it does represent an out-of-design condition (except following a system crash in which before image journaling was not in use) that your GT.M support channel would like to know about.

DBCBAADFILE

DBCBAADFILE, Source file xxx does not appear to have been generated by DBCERTIFY SCAN - rerun SCAN or specify correct file

DBCERTIFY/V5CBSU Error: V5CBSU and DBCERTIFY CERTIFY require the output file from DBCERTIFY SCAN. The file which was specified is not in the correct format.

Action: Specify the file created by DBCERTIFY SCAN. Rerun DBCERTIFY SCAN if needed.

DBCCERR

DBCCERR, Interlock instruction failure in critical mechanism for region xxxx

Run Time Error: This indicates that an interlocked operation for the specified region failed.

Action: Report this database concurrency error to the group responsible for database integrity at your operation.

DBCCMDFAIL

DBCCMDFAIL, Executed command failed with return code xxxx yyyy which executed yyyy yyyy

DBCERTIFY Error: During processing, the DBCERTIFY attempts to execute certain DSE and/or MUPIP commands in temporary command scripts that DBCERTIFY creates. The specified command failed to execute.

Action: The action to take depends on the code returned by the attempt and if any associated messages were created on either the console or the operator log. Some common causes of problems could be that \$gtm_dist (UNIX) or GTM\$DIST (OpenVMS) are not properly pointing to the current GT.M V4 version or DBCERTIFY has no access or access to the wrong global directory for which it is executing commands.

DBCDBCERTIFIED

DBCDBCERTIFIED, Database xxx has been certified for use with xxxx

DBCERTIFY Message: DBCERTIFY CERTIFY has successfully completed and marked the database as certified for use by the specified GT.M version.

Action: Either keep running GT.M V4 version or proceed immediately to GT.M V5 MUPIP UPGRADE at user's discretion.

DBCDBNOCERTIFY

DBCDBNOCERTIFY, Database xxxx HAS NOT been certified due to the preceeding errors - rerun DBCERTIFY SCAN

Mupip Error: MUPIP UPGRADE for V5 triggers this error if it finds the DBCERTIFY CERTIFY command has not run to completion on database xxx.

Action: Complete the scan phase of DBCERTIFY by executing the DBCERTIFY SCAN command.

DBCREC2BIGINBLK

DBCREC2BIGINBLK, A Record in block bbbb has a length greater than the maximum uuuuu in database dddd.

Mupip Error: MUPIP UPGRADE for V5 triggers this error when the size of a record in block bbbb exceeds the maximum record size of uuuu in database region dddd.

Action: -

DBCINTEGERR

DBCINTEGERR, Encountered integrity error in database xxxx

DBCERTIFY Error: DBCERTIFY discovered what appears to be an integrity error while processing the specified database. This error is accompanied by a secondary message giving an explanation of what the error is.

Action: Run a MUPIP INTEG (not FAST integ) on the database in question; fix damage, then re-run the phase reporting the error. If the integrity error persists, contact your GT.M support channel.

DBCKILLIP

DBCKILLIP, Cannot proceed with kill-in-progress indicator set for database xxx

DBCERTIFY Error: DBCERTIFY discovered that the kill in progress indicator was on for the specified database. DBCERTIFY will not process a database with this indicator on.

Action: Run a MUPIP INTEG (FAST integ is OK) on the database in question; correct errors, then re-run the phase reporting the error. If the error persists, contact your GT.M support channel.

DBCLNUPINFO

DBCLNUPINFO, Database file xxxx / yyyy

Run Time Information: When a process that holds the critical sectionlock on one or more databases gets abnormally terminated, it dumps information pertaining to its current state into the global sections for each of the concerned databases. The next process that references the concerned database notices the previous abnormal termination and uses the dumped information to update the global buffers and takes the database to a safe and consistent state. During this transition, the process displays a subset of the dumped information in the operator log to be used for debugging purposes by your GT.M support channel, in case database integrity errors are experienced later.

Action: The message text describes the cause of this error. Report any database structure errors to the group responsible for database integrity at your operation.

DBCMODBLK2BIG

DBCMODBLK2BIG, Block 0xaaa has been modified since DCERTIFY SCAN but is still too large or has an earlier TN than in DCERTIFY SCAN - Rerun scan

DBCERTIFY Error: DBCERTIFY reports this error when the block it is processing has a different TN than it did in the scan phase yet the block is still too large.

Action: This condition indicates that something has been done to the database since the scan phase was run - either it was restored from an earlier backup or the reserved bytes value was (even temporarily) reduced. DBCERTIFY SCAN must be rerun.

DBCMPBAD

DBCMPBAD, xxxx yyyy Compression count not maximal

Run Time Warning: This indicates that a database operation failed because block xxxx contains a record at offset yyyy with a compression count that is too low.

Action: Report this database structure error to the group responsible for database integrity at your operation.

DBCMPMX

DBCMPMX, xxxx yyyy Record compression count is too large

Run Time Warning: This indicates that during block certification or DSE/MUPIP integrity check, the block xxxx was found to contain a record at offset yyyy that exceeds the compression count.

Action: Report this database structure error to the group responsible for database integrity at your operation.

DBCOMMITCLNUP

DBCOMMITCLNUP, Pid dddd [hhhh] handled error (code = eeee) during commit of xxxx transaction in database file yyyy

Information Message: This message is output to the operator log and indicates that there was an error in the midst of committing a xxxx (TP or non-TP) transaction that involved the database file yyyy, but the process (pid = dddd in decimal and hhhh in hexadecimal) handled the error and completed the commit. If non-zero, the error code eeee is what triggered the error in the first place. If zero, accompanying syslog messages will contain information on the cause.

Action: In most cases the commit will be successfully completed. But in very rare cases, there might be errors that prevent the transaction from being successfully completed. To determine if there was any error, examine the following operator log messages. If there are more than 3 (three) DBCLNUPINFO messages for the same database file from the same process-id, then that particular database is suspect and an integrity check of that database needs to be done at the earliest. In addition, contact your GT.M support channel with the operator log messages.

DBCMPNZRO

DBCMPNZRO, xxxx yyyy First record of block has nonzero compression count

Run Time Warning: This indicates that during block certification or DSE/MUPIP integrity check the first record at offset yyyy of block xxxx was found to have a nonzero compression count.

Action: Report this database structure error to the group responsible for database integrity at your operation.

DBCNOEXTND

DBCNOEXTND, Unable to extend database xxx

DBCERTIFY Error: DBCERTIFY attempted to use MUPIP EXTEND to extend the database but the attempt failed.

Action: Examine the accompanying messages from the MUPIP EXTEND attempt to see why the extend failed. Some common causes for this are that \$gtm_dist on UNIX or GTM\$DIST on OpenVMS did not properly point to the currently installed V4 distribution, or there was insufficient disk space to perform the expansion.

DBCNOFINISH

DBCNOFINISH, DBCERTIFY unable to finish all requested actions

DBCERTIFY Error: This indicates DBCERTIFY encountered an error, which prevented the requested action from completing. The action has partially completed.

Action: Review the accompanying message(s) for additional information to identify the cause.

DBCNOTSAMEDB

DBCNOTSAMEDB, Database has been moved or restored since DBCERTIFY SCAN - Rerun scan

DBCERTIFY Error: DBCERTIFY has noted that the unique database identifiers for the database have changed since DBCERTIFY SCAN was run.

Action: The database is required to have not been moved around or restored or recovered since DBCERTIFY SCAN was run. DBCERTIFY SCAN must be rerun.

DBCNTRLERR

DBCNTRLERR, Database file xxxx: control error suspected but not found

Run Time Error: This indicates that GT.M detected the possibility of damage to database cache structures and performed a cache verification and rebuild, but found no evidence of damage.

Action: Verify that there are no locked or runaway processes. Check disk loads for evidence of resource constraints.

DBCOMPTOOLRG

DBCOMPTOOLRG, xxxx Record has too large compression count

MUPIP Error: This is a MUPIP INTEG error. Refer to the topic MUPIP INTEG Errors in About This Manual section of this manual.

DBCREC2BIG

DBCREC2BIG, Record with key xxx is length yyy in block 0xaaa is greater than the maximum length yyy in database xxx

DBCERTIFY Error: DBCERTIFY has identified a record with the given key in the given block with a length that exceeds the maximum length allowed in the given database.

Action: This is typically due to the user reducing the maximum record length to meet the DBCERTIFY requirements but not verifying that no records exist that exceeds that length. The solution is to either delete or otherwise restructure the record or to MUPIP extract/load into a database with a larger blocksize.

DBCREINCOMP

DBCREINCOMP, xxxx Header indicates database file creation was interrupted before completion

Run Time/MUPIP Error: This is either a MUPIP Integ error, or this indicates that a database operation tried to activate a database file that was improperly initialized.

Action: Delete the damaged file and use MUPIP CREATE to recreate the database. Refer to 'MUPIP INTEG Error Messages' table in the Chapter 11-Maintaining Database Integrity, chapter of the Administration and Operations Guide of GT.M.

DBCRERR

DBCRERR, Database file xxxx, cr location yyyy blk = zzzz error: aaaa was bbbb, expecting cccc -- called from module xxx at line yyy

Run Time Error: This usually indicates that a process was abnormally terminated and left database control structures in an inconsistent state in shared memory.

Action: GT.M often fixes this error unless there is a more serious problem causing this error. If there is a more serious problem, accompanying messages identify the cause.

DBCRERR8

DBCRERR8, Database file xxxx, or location yyyy blk = zzzz error: aaaa was bbbb, expecting cccc -- called from module yyy at line xxx

Run time Error: This message is the same as a DBCRERR message except that bbbb and cccc are 8-byte quantities (as opposed to 4-byte quantities in DBCRERR). See Error description for message DBCRERR.

Action: GT.M often fixes this error unless there is a more serious problem causing this error. If there is a more serious problem, accompanying messages identify the cause.

DBCSCNOTCMPLT

DBCSCNOTCMPLT, Specified DBCERTIFY SCAN output file is not complete - Rerun scan

DBCERTIFY Error: DBCERTIFY CERTIFY has noted that the header of the scan phase output is not filled in indicating that the scan phase did not complete normally.

Action: Rerun DBCERTIFY SCAN to produce a complete output file for the certify phase to process.

DBDANGER

DBDANGER, Process pppp killed while committing update for database file xxxx. Possibility of damage to block yyyy.

Run Time Warning: This message is issued when a recovery of the database global buffer cache structures needs to be performed. It might discover that the cache recovery was necessary because of a GT.M process being killed (kill-9) while in the process of committing a change to the database. The cache recovery routine issues this message while proceeding with the recovery.

Action: This is a warning type message indicating possible database corruption due to process kills (kill-9). A database integrity check is recommended. Make sure that kill-9 or STOP/ID is not used to stop any GT.M processes.

DBDIRTSUBSC

DBDIRTSUBSC, <xxxx Directory tree block contains non name-level entries>

Run Time/DSE Information: This indicates that the specified database block has an internal structural damage since it contains subscripts global variable names even though this block is part of the directory tree.

Action: Report this database structure error to the group responsible for database integrity at your operation.

DBDSRDFMTCHNG

DBDSRDFMTCHNG, Database file xxx, Desired DB Format set to yyy by zzz with pid ppp [0xppp] at transaction number [0xttt].

MUPIP Information: The desired database block format has been changed to version yyy for database file xxx by the zzz command with process number ppp at transaction number ttt.

DBENDIAN

DBENDIAN, Database file xxxx is aaaa endian on a gggg endian system

MUPIP/Run Time Error: This indicates that the database file being opened is in the wrong endian format for the current system. This usually means that the file was copied from another system with the opposite endian format.

Action: To use the database file on the current system, change the endian format using the MUPIP ENDIANCVT command.

DBFGTBC

DBFGTBC, xxxx File size larger than block count would indicate

MUPIP Error: This is a MUPIP INTEG error. Refer to the topic MUPIP INTEG Errors in About This Manual section of this manual.

DBFHEDERR ●

Last used version: **V5.3-001A**

DBFHEADERR, Database file xxxx: control problem: yyyy was zzzz expecting aaaa

Run Time Error: This indicates that database cache recovery was triggered due to some abnormal event, and the recovery routine detected damage to the control structures in the database global buffer cache.

Action: The system automatically attempts to correct the problem. If this error continues to occur, attempt MUPIP RUNDOWN and if that fails too, restore database from backup and replay journal files. Report this error to your GT.M support channel if necessary.

DBFHEADERR4

DBFHEADERR4: Database file ffff: control problem: aaaa was xxxx expecting yyyy

Run Time Error: This indicates that database cache recovery was triggered due to some abnormal event, and the recovery routine detected damage to the control structures in the database global buffer cache.

Action: The system automatically attempts to correct the problem. If this error continues to occur, attempt MUPIP RUNDOWN and if that fails too, restore database from backup and replay journal files. Report the entire incident context to your GT.M support channel if necessary.

DBFHEADERR8

DBFHEADERR8: Database file ffff: control problem: aaaa was xxxx expecting yyyy

Run Time Error: This indicates that database cache recovery was triggered due to some abnormal event, and the recovery routine detected damage to the control structures in the database global buffer cache.

Action: The system automatically attempts to correct the problem. If this error continues to occur, attempt MUPIP RUNDOWN and if that fails too, restore database from backup and replay journal files. Report the entire incident context to your GT.M support channel if necessary.

DBFILERR

DBFILERR, Error with database file. xxxx.

Run Time Error: This indicates that an I/O operation on the database file encountered an error.

Action: Review the accompanying message(s) for a detailed error status.

DBFILEXT

DBFILEXT, Database file xxxx extended from yyyy blocks to zzzz blocks at transaction aaaa

Run Time/MUPIP Information: This operator log message indicates that the specified database file extended as described by the message.

DBFILOPERR

DBFILOPERR, Error doing database I/O to region xxxx

Run Time Error: This indicates that the database manager portion of the run-time system encountered an error when it attempted to open, read, write, or close a database file.

Action: Report this error to the group responsible for database integrity at your operation. Review the accompanying message(s) for additional information and analyze the system error log.

DBFLCORRP

DBFLCORRP, xxxx Header indicates database file is corrupt

Run Time/MUPIP Error: This indicates that a database operation tried to activate database file xxxx, which was previously marked as damaged.

Action: If ROLLBACK (either -NOONLINE or -ONLINE) terminates abnormally (say because of a kill -9), it leaves the database in a potentially inconsistent state indicated by the FILE corrupt field in the database file header. When an ROLLBACK terminates leaving this field set, all other processes receive DBFLCORRP errors any time they attempt to interact with the database. The best way to clear DBFLCORRP is by running another ROLLBACK. MUPIP SET -FILE -PARTIAL_RECOV_BYPASS and DSE CHANGE -FILE -CORRUPT=FALSE -NOCRIT can also clear this condition but these commands do not ensure that the database has consistent state so you should always run MUPIP INTEG after executing these commands.

DBFRZRESETFL

DBFRZRESETFL, Freeze release failed on database file xxxx

MUPIP Error: This indicates that MUPIP failed to release the freeze on database file xxxx.

Action: Review the accompanying message(s) for additional information. Analyze DSE DUMP /FILE /ALL output.

DBFRZRESETSUC

DBFRZRESETSUC, Unfreeze successfully done on database file xxxx.

MUPIP Information: This indicates that RECOVER encountered a database file that had a state of FREEZE, and released the FREEZE.

DBFSTBC

DBFSTBC, xxxx File size smaller than block count would indicate

MUPIP Error: This is a MUPIP INTEG error. Refer to the topic MUPIP INTEG Errors in About This Manual section of this manual.

DBFSTHEAD

DBFSTHEAD, xxxx File smaller than database header

MUPIP Error: This is a MUPIP INTEG error. Refer to the topic MUPIP INTEG Errors in About This Manual section of this manual.

DBFSYNCERR

DBFSYNCERR, Error synchronizing database file xxxx to disk

Run Time Error: While using before-image journaling, the database is hardened to disk every time GT.M writes an epoch-record. If this operation returns an error, then the DBFSYNCERR error is issued to the user accompanied by system information about the cause of the system service error.

Action: Trouble shoot the file system, on which the database file resides, for issues related to FSYNC(). Report the entire incident context to your GT.M support channel along with any GT.M operator log messages within the same time frame.

DBGTDDBMAX

DBGTDDBMAX, xxxx Key larger than database maximum

MUPIP Error: This is a MUPIP INTEG error. Refer to the topic MUPIP INTEG Errors in About This Manual section of this manual.

DBHEADINV

DBHEADINV, xxxx Header size not valid for database

MUPIP Error: This is a MUPIP INTEG error. Refer to the topic MUPIP INTEG Errors in About This Manual section of this manual.

DBIDMISMATCH

DBIDMISMATCH, Database file xxxx ID (region yyyy) does not match file ID in shared memory (ID=zzzz). Ensure region is properly rundown.

Run Time Error: When a GT.M process attaches to a database and finds the corresponding shared memory structures initialized already, it performs integrity checks on the shared memory contents to ensure that they correspond back to the database file. When the shared memory copy of the database file ID does not match with the actual file ID of the database, the above error is issued.

Action: Perform a MUPIP RUNDOWN on that region. If it fails with the same DBIDMISMATCH error, then the shared memory contents are corrupt. Consult your GT.M support channel before proceeding further.

DBINCLVL

DBINCLVL, xxxx Block at incorrect level

MUPIP Error: This is a MUPIP INTEG error. Refer to the topic MUPIP INTEG Errors in About This Manual section of this manual.

DBINCRVER

DBINCRVER, xxxx Incorrect version of GT.M database

MUPIP Error: This is a MUPIP INTEG error. Refer to the topic MUPIP INTEG Errors in About This Manual section of this manual.

DBINVGBL

DBINVGBL, xxxx Invalid mixing of global names

MUPIP Error: This is a MUPIP INTEG error. Refer to the topic MUPIP INTEG Errors in About This Manual section of this manual.

DBJNLNOTMATCH

DBJNLNOTMATCH, Database xxxx points to journal file name yyyy but the journal file points to database file zzzz

MUPIP Error: This indicates that there is a mismatch in the name of the database file xxxx and the name zzzz, saved in the journal file header yyyy.

Action: Contact your GT.M support channel if the cause of the error cannot be diagnosed. If appropriate change the database file name in the journal file using the command MUPIP SET /jnlfile /dbfilename=xxxx yyyy.

DBKEYGTIND

DBKEYGTIND, xxxx Key greater than index key

MUPIP Error: This is a MUPIP INTEG error. Refer to the topic MUPIP INTEG Errors in About This Manual section of this manual.

DBKEYMN

DBKEYMN, xxxx Key too short

Run Time Warning: This indicates that a block certification or DSE/MUPIP integrity check failed on block xxxx since it contains a record at offset yyyy with a key that does not meet the minimum size requirement.

Action: Report this database structure error to the group responsible for database integrity at your operation.

DBKEYMX

DBKEYMX, xxxx Key too long

Run Time Warning: This indicates that a block certification or DSE/MUPIP integrity check failed on block xxxx since it contains a record at offset yyyy whose key exceeds the allowable size.

Action: Report this database structure error to the group responsible for database integrity at your operation.

DBKEYORD

DBKEYORD, xxxx Keys out of order

Run Time/MUPIP/DSE Warning: This indicates that a DSE/MUPIP INTEG command determined that the block contains a record at offset yyyy whose key is not in proper M collating sequence. This error is also reported at run-time if block certification fails on a particular block. Block certification is active if GDSCERT is enabled by a VIEW command.

Action: Report this database structure error to the group responsible for database integrity at your operation.

DBKGTALLW

DBKGTALLW, xxxx Key larger than maximum allowed length

MUPIP Error: This is a MUPIP INTEG error. Refer to the topic MUPIP INTEG Errors in About This Manual section of this manual.

DBLOCMBINC

DBLOCMBINC, xxxx Local bit map incorrect

MUPIP Error: This is a MUPIP INTEG error. Refer to the topic MUPIP INTEG Errors in About This Manual section of this manual.

DBLRCINVSZ

DBLRCINVSZ, xxxx Last record of block has invalid size

MUPIP Error: This is a MUPIP INTEG error. Refer to the topic MUPIP INTEG Errors in About This Manual section of this manual.

DBLTSIBL

DBLTSIBL, xxxx Keys less than sibling's index key

MUPIP Error: This is a MUPIP INTEG error. Refer to the topic MUPIP INTEG Errors in About This Manual section of this manual.

DBLVLINC

DBLVLINC, xxxx Local bitmap block level incorrect

MUPIP Error: This is a MUPIP INTEG error. Refer to the topic MUPIP INTEG Errors in About This Manual section of this manual.

DBMAXKEYEXC

DBMAXKEYEXC, xxxx Maximum key size for database exceeds design maximum

MUPIP Error: This is a MUPIP INTEG error. Refer to the topic MUPIP INTEG Errors in About This Manual section of this manual.

DBMAXNRSUBS

DBMAXNRSUBS, XXXX Maximum number of subscripts exceeded.

Compile Time Error: The subscripted variable required more than 31 subscripts.

Action: Modify the routine to observe this limit on subscripts in a single variable.

DBMAXREC2BIG

DBMAXREC2BIG, Maximum record size (xxx) is too large for this block size (yyy) - Maximum is zzz

DBCERTIFY/MUPIP Error: DBCERTIFY and MUPIP UPGRADE report this error when the maximum record size is too close to the database blocksize and does not allow room for the expanded V5 block header.

Action: Reduce the maximum record size or mupip extract/load into a database with a larger blocksize. Note that if the maximum record size is reduced with DSE, it is possible that records that exceed the reduced size still exist in the database which is now an integrity error. DBCERTIFY SCAN will find these blocks and report on them if they exist.

DBMAXRSEXBL

DBMAXRSEXBL, xxxx Maximum record size for database exceeds what the block size can support

MUPIP Error: This is a MUPIP INTEG error. Refer to the topic MUPIP INTEG Errors in About This Manual section of this manual.

DBMBMINCFRE

DBMBMINCFRE, xxxx Master bit map incorrectly asserts this local map has free space

MUPIP Error: This is a MUPIP INTEG error. Refer to the topic MUPIP INTEG Errors in About This Manual section of this manual.

DBMBPFLDIS

DBMBPFLDIS, xxxx Master bit map shows this map full, in disagreement with both disk and INTEG results

MUPIP Error: This is a MUPIP INTEG error. Refer to the topic MUPIP INTEG Errors in About This Manual section of this manual.

DBMBPFLDLBM

DBMBPFLDLBM, xxxx Master bit map shows this map full, agreeing with disk local map

MUPIP Error: This is a MUPIP INTEG error. Refer to the topic MUPIP INTEG Errors in About This Manual section of this manual.

DBMBPFLINT

DBMBPFLINT, xxxx Master bit map shows this map full, agreeing with MUPIP INTEG

MUPIP Error: This is a MUPIP INTEG error. Refer to the topic MUPIP INTEG Errors in About This Manual section of this manual.

DBMBPFRDLBM

DBMBPFRDLBM, xxxx Master bit map shows this map has space, agreeing with disk local map

MUPIP Error: This is a MUPIP INTEG error. Refer to the topic MUPIP INTEG Errors in About This Manual section of this manual.

DBMBPFRINT

DBMBPFRINT, xxxx Master bit map shows this map has space, agreeing with MUPIP INTEG

MUPIP Error: This is a MUPIP INTEG error. Refer to the topic MUPIP INTEG Errors in About This Manual section of this manual.

DBMBPINCFL

DBMBPINCFL, xxxx Master bit map incorrectly marks this local map full

MUPIP Error: This is a MUPIP INTEG error. Refer to the topic MUPIP INTEG Errors in About This Manual section of this manual.

DBMBSIZMN

DBMBSIZMN, xxxx Map block too small

MUPIP Error: This is a MUPIP INTEG error. Refer to the topic MUPIP INTEG Errors in About This Manual section of this manual.

DBMBSIZMX

DBMBSIZMX, xxxx Map block too large

MUPIP Error: This is a MUPIP INTEG error. Refer to the topic MUPIP INTEG Errors in About This Manual section of this manual.

DBMBTNSIZMX

DBMBTNSIZMX, xxxx Map block transaction number too large

MUPIP Error: This is a MUPIP INTEG error. Refer to the topic MUPIP INTEG Errors in About This Manual section of this manual.

DBMINRESBYTES

DBMINRESBYTES, Minimum RESERVED BYTES value required for certification/upgrade is xxx - Currently is yyy

DBCERTIFY/MUPIP Error: DBCERTIFY and MUPIP UPGRADE report this error when the reserved bytes field of the database file header (as shown by DSE DUMP -FILEHEADER) is not at a sufficient value for the GT.M V5 upgrade.

Action: Increase the reserved bytes value with either MUPIP or DSE so that the value is at least 8 bytes for UNIX and 9 bytes for OpenVMS. Note that the reserved bytes value is reduced by the above amounts by MUPIP UPGRADE.

DBMISALIGN

DBMISALIGN, Database file xxxx has yyyy blocks which does not match alignment rules. Reconstruct the database from a backup or extend it by at least zzzz blocks.

MUPIP Error: This is an auxiliary message, and is preceded by a primary message.

Action: Follow the primary message description and action as specified in this manual.

DBMRKBUSY

DBMRKBUSY, xxxx Block incorrectly marked busy

MUPIP Error: This is a MUPIP INTEG error. Refer to the topic MUPIP INTEG Errors in About This Manual section of this manual.

DBMRKFREE

DBMRKFREE, xxxx Block incorrectly marked free

MUPIP Error: This is a MUPIP INTEG error. Refer to the topic MUPIP INTEG Errors in About This Manual section of this manual.

DBMXRSEXCMIN

DBMXRSEXCMIN, xxxx Maximum record size for database exceeds what the block size can support

MUPIP Error: This is a MUPIP INTEG error. Refer to the topic MUPIP INTEG Errors in About This Manual section of this manual.

DBNAMEMISMATCH

DBNAMEMISMATCH, Database file xxxx (region (yyyy) referenced by shared memory (ID=zzzz) is not accessible. Ensure region is properly rundown

Run Time Warning: When a GT.M process attaches to a database and finds the corresponding shared memory structures already initialized, it performs integrity checks on the shared memory contents to ensure that they correspond back to the database file. When the shared memory points to a database file name that is not valid, this error is issued.

Action: This error means that the shared memory contents are corrupt; consult your GT.M support channel before proceeding further.

DBCDBNOCERTIFY

DBCDBNOCERTIFY, Database xxxx HAS NOT been certified due to the preceding errors - rerun DBCERTIFY SCAN

Mupip Error: MUPIP UPGRADE for V5 triggers this error if it finds the DBCERTIFY CERTIFY command has not run to completion on database xxx.

Action: Complete the scan phase of DBCERTIFY by executing the DBCERTIFY SCAN command.

DBNOCRE

DBNOCRE, Not all specified databases, or their associated journal files were created

MUPIP Warning: This indicates that MUPIP CREATE failed a task in creating the new database files.

Action: See accompanying messages for more detailed information on the failure.

DBNOREGION

DBNOREGION, None of the database regions accessible

MUPIP/DSE Error: MUPIP INTEG or DSE can report this error. This indicates that none of the database files specified in the Global Directory could be opened (or they do not exist).

Action: Ensure the proper assignment for the environment variable, gtmglbdir/logical name GTM\$GBLDIR. Verify that the database file(s) specified in the Global Directory exist and that their protection allows access. Also, refer to the 'MUPIP INTEG Error Messages' table in the Chapter 11-Maintaining Database Integrity, chapter of the Administration and Operations Guide of GT.M.

DBNOTDB

DBNOTDB, xxxx File does not have a valid GDS file header

MUPIP Error: This is a MUPIP INTEG error. Refer to the topic MUPIP INTEG Errors in About This Manual section of this manual.

DBNOTGDS

DBNOTGDS, xxxx - Unrecognized database file format

Run Time Error: This indicates that a database operation attempted to activate file xxxx, which is not a GDS file.

Action: Use GDE to ensure that the files in the Global Directory are properly named. It is likely that something other than GT.M or its utilities wrote to a database file or created a file with a name that coincides with one specified in the current Global Directory.

DBNOTMLTP

DBNOTMLTP, xxxx Block size not a multiple of 512 bytes

MUPIP Error: This is a MUPIP INTEG error. Refer to the topic MUPIP INTEG Errors in About This Manual section of this manual.

DBOPNERR

DBOPNERR, Error opening database file xxxx

Run Time Error: This indicates that a database operation tried to open the database file xxxx, which was inaccessible.

Action: Use GDE to ensure that the files in the Global Directory are properly named. Use the host shell command to ensure that the files exist and have proper security. Review the accompanying message(s) that indicate the reason for file-open failure.

DBPREMATEOF

DBPREMATEOF, Premature end of file with database file xxxx

Run Time Error: This indicates that the size of the database file is less than the size of the minimum required database file header. The file may not be a valid GT.M file.

Action: Investigate whether the file was properly created (with MUPIP) or inappropriately truncated. Also check whether the global directory points to a valid database.

DBPRIVERR

DBPRIVERR, No privilege for attempted update operation for file: xxxx

Run Time Error: This indicates that the process did not have write access to database file xxxx.

Action: Disable application access to the function that resulted in the error or have the security manager grant write access to the appropriate user. Under some circumstances, security considerations may require moving some globals to other regions.

DBPTRMAP

DBPTRMAP, xxxx Block pointer is a bit map block number

Run Time Error: This indicates that the block certification facility encountered a block pointer to a bitmap location (in the index block).

Action: Report this error to the group responsible for database integrity at your operation.

DBPTRMX

DBPTRMX, xxxx Block pointer larger than file maximum

Run Time Warning: This indicates that a database operation failed because the block contains a record at offset yyyy whose block pointer points beyond the end of the file.

Action: Report this database structure error to the group responsible for database integrity at your operation.

DBPTRNOTPOS

DBPTRNOTPOS, xxxx Block pointer negative

Run Time Warning: This indicates that xxxx block contains a record yyyy, which nests an invalid index pointer, and so failed block certification.

Action: Report this database structure error to the group responsible for database integrity at your operation.

DBQUELINK

DBQUELINK, Database file xxxx, element location yyyy: blk = zzzz: control aaaa queue problem: was bbbb, expecting cccc

Run Time Error: This indicates that database cache recovery was triggered due to some abnormal event and the recovery routine detected damage to an internal GT.M queue control structure in the database global buffer cache.

Action: The system automatically attempts to correct the problem. If this error continues to occur, attempt MUPIP RUNDOWN and if that fails too, restore database from backup and replay the journal files. Report the error to your GT.M support channel if necessary.

DBRBNLBMN

DBRBNLBMN, xxxx Root block number is a local bit map number

MUPIP Error: This is a MUPIP INTEG error. Refer to the topic MUPIP INTEG Errors in About This Manual section of this manual.

DBRBNNEG

DBRBNNEG, xxxx Root block number negative

MUPIP Error: This is a MUPIP INTEG error. Refer to the topic MUPIP INTEG Errors in About This Manual section of this manual.

DBRBNTOOLRG

DBRBNTOOLRG, xxxx Root block number greater than the last block number in file

MUPIP Error: This is a MUPIP INTEG error. Refer to the topic MUPIP INTEG Errors in About This Manual section of this manual.

DBRDERR

DBRDERR, Cannot read database file xxxx after opening

Run Time Error: This indicates that a database operation attempted reading the file xxxx without having the read access.

Action: Use the host shell commands to verify the file access and adjust it, if appropriate.

DBRDONLY

DBRDONLY, Database file xxxx read only

Run Time Error: This indicates that a database operation tried to write to a read-only file or database.

Action: Verify the read and write privileges for the database and adjust it, if appropriate.

DBREADBM

DBREADBM, xxxx Read error on bitmap

MUPIP Error: This is a MUPIP INTEG error. Refer to the topic MUPIP INTEG Errors in About This Manual section of this manual.

DBREMOTE

DBREMOTE, Database region xxxx is remote; perform maintenance on the server node

DSE, LKE, or MUPIP Error: This indicates that a database maintenance operation was attempted on region xxxx. This node does not maintain the region directly; instead, it uses GT.CM to access the node as a client. This error is also reported by ^%GBLDEF if the target global is mapped to another node and served by GT.CM.

Action: Perform database maintenance on the server node.

DBRLEVLSTONE

DBRLEVLSTONE, xxxx Root level less than one

MUPIP Error: This is a MUPIP INTEG error. Refer to the topic MUPIP INTEG Errors in About This Manual section of this manual.

DBRLEVTOOHI

DBRLEVTOOHI, xxxx Root level higher than maximum

MUPIP Error: This is a MUPIP INTEG error. Refer to the topic MUPIP INTEG Errors in About This Manual section of this manual.

DBRNDWN

DBRNDWN, Error during global database rundown for region xxxx. Please notify those responsible for proper database operation.

Run Time Error: This indicates that a process encountered a problem attempting to rundown the database file xxxx.

Action: Refer to the associated messages for more information.

DBRNDWNWRN

DBRNDWNWRN, Global database xxxx not rundown successfully by PID yyyy [zzzz]. Global section was not removed.

Run Time Error: When the last process attached to a GT.M database shared memory segment or global section detaches from the same, it normally removes the segment/section from the system. In case of an error while flushing the contents from the segment/section to the database file on disk, this removal is not done and this error is issued.

Action: Attempt a MUPIP RUNDOWN on that region. In case of an error, attempt corrective action corresponding to the displayed error.

DBROLLEDBACK

DBROLLEDBACK. Concurrent ONLINE ROLLBACK detected on one or more regions. The current operation is no longer valid>/error

Run Time Error: This indicates a non-TP mini-transaction attempted to interact with the database and found that a concurrent online rollback had taken the database to a state earlier than the one at the end of the last mini-transaction by this process unless there has been an intervening TP transaction.

Action: Application dependent - this error indicates a discontinuity in the database state that may cause inconsistent application data.

DBROOTBURN

DBROOTBURN, xxxx Root block has data level

Run Time/DSE Information: This indicates that the specified block has a block certification error or a DSE integrity error.

Action: Report this database structure error to the group responsible for database integrity at your operation.

DBRSIZMN

DBRSIZMN, xxxx Record too small

Run Time Warning: This indicates that a DSE/MUPIP INTEG command failed because block xxxx contains a record at offset yyyy that does not meet the minimum size requirement.

Action: Report this database structure error to the group responsible for database integrity at your operation.

DBRSIZMX

DBRSIZMX, xxxx Record too large

Run Time Warning: This indicates that a DSE/MUPIP INTEG command failed because block xxxx contains a record at offset yyyy that exceeds the maximum record size for the database region.

Action: Report this database structure error to the group responsible for database integrity at your operation.

DBSHMNAMEDIFF

DBSHMNAMEDIFF, Database file ffff points to shared memory mmmm which points to a different database file

Runtime Error: Database access gives this error, if the database is copied or moved without properly closing it. This error indicates that database ffff and shared memory mmmmm do not correspond to each other.

Action: Perform MUPIP RUNDOWN on the database.

DBSTARCMP

DBSTARCMP xxxx Star record has nonzero compression count

Run Time Warning: This indicates that a block certification or DSE integrity check failed on xxxx. Block xxxx contains a record at offset yyyy that should be a star key; however, it has a non-zero compression count.

Action: Report this database structure error to the group responsible for database integrity at your operation.

DBSTARSIZ

DBSTARSIZ, xxxx Star record has wrong size

Run Time Warning: This indicates that a block certification or DSE integrity check failed on xxxx. Block xxxx contains a record at offset yyyy whose star key does not have the proper size.

Action: Report this database structure error to the group responsible for database integrity at your operation.

DBSVBNMIN

DBSVBNMIN, xxxx Start VBN smaller than possible

MUPIP Error: This is a MUPIP INTEG error. Refer to the topic MUPIP INTEG Errors in About This Manual section of this manual.

DBSZGT64K

DBSZGT64K, xxxx Block size is greater than 64k

MUPIP Error: This is a MUPIP INTEG error. Refer to the topic MUPIP INTEG Errors in About This Manual section of this manual.

DBTN

DBTN, Block TN is xxxx

MUPIP Information: This is an auxiliary message and is preceded by a primary message.

Action: Follow the primary message description and action as specified in this manual.

DBTNLTCTN

DBTNLTCTN, Transaction numbers greater than the current transaction were found

MUPIP Information: This is an auxiliary message and is preceded by a primary message. It accompanies DBTNTOOLG.

Action: Follow the primary message description and action as specified in this manual.

DBTNNEQ

DBTNNEQ, xxxx Current tn and early tn are not equal

MUPIP Error: This is a MUPIP INTEG error. Refer to the topic MUPIP INTEG Errors in About This Manual section of this manual.

DBTNRESET

DBTNRESET, Cannot reset transaction number for this region

Run Time Error: This message is an auxiliary message to the DBRDONLY message.

Action: Follow the primary message description and action as specified in this manual.

DBTNRESETINC

DBTNRESETINC, WARNING: tn_reset for database is incomplete due to integrity errors

Run Time Warning: Automatic resetting of transaction number has not been done due to other errors.

Action: Clean up the other errors and then run integ again.

DBTNTOOLG

DBTNTOOLG, xxxx Block transaction number too large

MUPIP Error: This is a MUPIP INTEG error. Refer to the topic MUPIP INTEG Errors in About This Manual section of this manual.

DBTOTBLK

DBTOTBLK, Total blocks should be xxxx, is yyyy

MUPIP Information: This is an auxiliary message, and is preceded by a primary message.

Action: Follow the primary message description and action as specified in this manual.

DBTTLBLK0

DBTTLBLK0, xxxx Total blocks equal zero

MUPIP Error: This is a MUPIP INTEG error. Refer to the topic MUPIP INTEG Errors in About This Manual section of this manual.

DBUNDACCMT

DBUNDACCMT, xxxx Cannot determine access method; trying with BG

MUPIP Error: This is a MUPIP INTEG error. Refer to the topic MUPIP INTEG Errors in About This Manual section of this manual.

DBVERPERFWARN1

DBVERPERFWARN1, Performance warning: Database aaaa is running in compatibility mode which degrades performance. Run MUPIP REORG UPGRADE for best overall performance.

Run Time Warning: This is a warning that the database is currently in compatibility (downgrade) mode. This mode causes all modified GDS blocks to be reformatted (to the downgraded database format) before they are flushed to the database file on disk. This is a very large performance hit.

Action: As the message indicates, run MUPIP REORG UPGRADE as soon as possible to move away from compatibility mode. This command can be run without taking the database offline. Once that completes successfully, the database is fully upgraded and there is no reformatting overhead anymore while flushing modified blocks to disk.

DBVERPERFWARN2

DBVERPERFWARN2, Performance warning: Database aaaa is not fully upgraded. Run MUPIP REORG UPGRADE for best overall performance.

Run Time Warning: This is a performance warning message that indicates the database is not yet fully upgraded i.e. there are still blocks in the database file that need to be upgraded. Staying in this mode causes some inefficiencies which include (but are not limited to) reading blocks from disk.

Action: As the message indicates, run MUPIP REORG UPGRADE at the earliest. This command can be run without taking the database offline. Once that completes successfully, the database file is fully upgraded.

DBWCVERIFYEND

DBWCVERIFYEND, Database file xxxx, write cache verification finished by pid pppp [aaaa] at transaction number yyyy

Run Time Information: This indicates that process-id pppp (aaaa in hexadecimal) has completed verification of the database cache for the database file xxxx.

DBWCVERIFYSTART

DBWCVERIFYSTART, Database file xxxx, write cache verification started by pid pppp [aaaa] at transaction number bbbb

Run Time Information: This indicates that process-id pppp (aaaa in hexadecimal), has started a verification of the database cache for the database file xxxx.

DCNINPROG

DCNINPROG, Attempt to initiate operation while disconnect was in progress

GT.CM Server Error: This indicates that the GT.CM tried to link while disconnecting or experiencing network problems.

Action: Review network error logs.

DELIMSIZNA

DELIMSIZNA, Delimiter size is not appropriate

GTM compile error or runtime error: A socket related IO command (OPEN or USE) triggers this error if the delimiter string exceeds its maximum length.

Action: Use a delimiter string with an appropriate length.

DELIMWIDTH

DELIMWIDTH, Delimiter length xxxx exceeds device width yyyy

Run Time Information: This indicates that the length of the first delimiter string specified in the DELIMITER deviceparameter exceeds the WIDTH of the socket device being OPEND (or USEd).

Action: Modify the first delimiter string to have a length of at most the WIDTH of the socket device.

DEVICEREADONLY

DEVICEREADONLY, Cannot write to a read-only device

Run Time Error: The application made an attempt to WRITE to a read-only device.

Action: Review code and context to see if the WRITE was intended for another device, and if so add the appropriate USE. If the WRITE is intended for this device, change the device OPEN to permit the WRITE.

DEVICEWRITEONLY

DEVICEWRITEONLY, Cannot read from a write-only device

Run-time error: The application made an attempt to READ from a device in a WRITEONLY state, typically due to the OPEN command specifications.

Action: Check for logic errors and revise the code.

DEVNOTIMP

DEVNOTIMP, XXXX device not implemented on in this environment

Run Time Error: This indicates that the device support is not available in the currently running version of GT.M.

Action: Refer to the GT.M documentation. Contact your GT.M support channel for information about the support available for this type of device on your platform.

DEVOPENFAIL

DEVOPENFAIL, Error opening xxxx

Run Time Error: This indicates that a GT.M process encountered an error while opening the device xxxx. A supplementary TEXT message and a system message provide more details about the cause of the error.

Action: Verify the device exists on the system where the OPEN is being attempted.

DEVPARINAP

DEVPARINAP, Device parameter inappropriate to this command

Compile Time Error: This indicates that an OPEN, USE, or CLOSE command specifies a deviceparameter that does not apply to the command.

Action: Look for deviceparameters that should be on other I/O commands. For example, the deviceparameter "DELETE" is valid on CLOSE but produces this error if it is applied to the USE command.

DEVARMNEG

DEVARMNEG, Deviceparameter must be a positive value

Run Time Error: This indicates that the argument to the deviceparameter had a negative value where only positive values are appropriate.

Action: Modify the argument to provide a positive value.

DEVARMROT

DEVARMROT, The protection specification is invalid

Compile Time Error: This indicates that an OPEN, USE, or CLOSE command specified a protection deviceparameter with an improperly formatted argument.

Action: Modify the protection mask.

DEVARMTOOBIG

DEVARMTOOBIG, String deviceparameter exceeds 255 character limit

Run/Compile Time Error: This indicates that an OPEN, USE, or CLOSE command specified a deviceparameter that equated to a string expression whose evaluated length exceeds 255 characters.

Action: Verify the program logic and modify it to use shorter deviceparameter strings.

DEVARMUNK

DEVARMUNK, Deviceparameter unknown

Compile Time Error: This indicates that an OPEN, USE, or CLOSE command specified an unrecognized keyword instead of an expected deviceparameter.

Action: Modify the deviceparameter in question.

DEVARMVALREQ

DEVARMVALREQ, A value is required for this device parameter

Compile Time Error: This indicates that an OPEN, USE, or CLOSE command specified a valid deviceparameter that requires a value; however, one was not provided.

Action: Ensure that deviceparameters have values where required. For example, the deviceparameter WRAP is valid but must include a value for the wrap length.

DIRONLY

DIRONLY, Directories only are allowed in file specs: xxxx

Run Time Error: This indicates that a SET of \$ZROUTINES specified a SRC qualifier with an argument element xxxx that was not a valid directory specification.

Action: Look for missing parenthesis or brackets.

DISTPATHMAX

DISTPATHMAX, \$gtm_dist path is greater than maximum (xxxx)

Run Time Error: This indicates that the path specified by the gtm_dist environment variable has exceeded the indicated maximum limit of 1024 bytes.

Action: Move the directory or use a link to shorten the path.

DIVZERO

DIVZERO, Attempt to divide by zero

Run Time Error: This indicates that a divide or module operator had a zero for its divisor operand.

Action: Modify the routine to protect against zero division.

DLCKAVOIDANCE

DLCKAVOIDANCE, Possible deadlock detected: Database pppp: Dbtn qqqq: t_tries rrrr: dollar_trestart ssss: now_crit tttt: TP transaction restarted

Run Time Error: This indicates that GT.Ms deadlock avoidance algorithm got triggered and aborted a possible deadlock.

Action: Report the error to your GT.M support channel with complete operator log information.

DLLCHSETM

DLLCHSETM, Routine XXX in library YYY was compiled with CHSET=M which is different from \$ZCHSET. Recompile with CHSET=UTF-8 and re-link.

Runtime Error: This error is triggered when a UTF-8 mode process attempts to execute a shared library's routine that was compiled in M-mode.

Action: Recompile and relink the routine using UTF-8-mode settings or switch to M mode.

DLLCHSETUTF8

DLLCHSETUTF8, Routine XXX in library YYY was compiled with CHSET=UTF-8 which is different from \$ZCHSET. Recompile with CHSET=M and re-link.

Runtime Error: This error is triggered when an M mode process attempts to execute a shared library's routine that was compiled in UTF-8 mode.

Action: Recompile and relink the routine using M-mode settings or switch to UTF-8 mode.

DLLNOCLOSE

DLLNOCLOSE, Failed to unload external dynamic library

Run Time Error: This indicates that the process encountered a problem attempting to unload a dynamically linked library.

Action: Refer to the associated messages for more information.

DLLNOOPEN

DLLNOOPEN, Failed to load external dynamic library xxxx

Run Time Error: This indicates that the process encountered a problem attempting to load a dynamically linked library.

Action: Refer to the associated messages for more information.

DLLNORTN

DLLNORTN, Failed to look up the location of the symbol xxxx

Run Time Error: This indicates that the process was unable to find the routine it needed in the dynamically linked library.

Action: Ensure that the environment variable for dynamic library path is defined and correctly locates the shared library file, as well as any other dependent shared libraries. Also ensure that the symbol xxxx is defined in one of the libraries.

DLLVERSION

DLLVERSION, Routine aaaa in library bbbb was compiled with an incompatible version of GT.M. Recompile with the current version and re-link.

Run Time Error: This indicates that the routine aaaa that was loaded out of the shared library bbbb was compiled with a version of GT.M that is not compatible with the current version of GT.M.

Action: Recompile the M routine aaaa and re-link (recreate) the shared library. The linker options for creating a shared library are platform dependant. Refer to GT.M Programmer's Manual for details.

DLRCILLEGAL

DLRCILLEGAL, Illegal \$CHAR() value xxxx

MUPIP Error: This indicates that MUPIP LOAD with the qualifier FORMAT=GO or ZWR encountered an invalid Unicode code point xxxx for \$CHAR() in its input stream.

Action: Edit or recreate the input file so the value falls within the valid range of Unicode code points.

DLRCTOOBIG

DLRCTOOBIG, xxxx value cannot be greater than 255

MUPIP Error: This indicates that MUPIP LOAD with the qualifier FORMAT=GO encountered xxxx in its input stream. xxxx was in the \$CHAR() format used for non-graphic characters but it exceeded the maximum acceptable value of 255.

Action: Refer to the topic MUPIP LOAD Errors in About This Manual section of this manual. Edit the input file so the value falls within the range of 0-255.

DLRCUNXEOR

DLRCUNXEOR, xxxx unexpected end of record in \$CHAR()/\$ZCHAR() subscript

MUPIP Error: This indicates that MUPIP LOAD with the qualifier FORMAT=GO encountered xxxx in its input stream. xxxx was in the \$CHAR() format used for non-graphic characters but the \$C() format did not complete properly.

Action: Refer to the topic MUPIP LOAD Errors in About This Manual section of this manual. Edit the input file to remove a spurious "\$" or fix a \$CHAR() representation.

DSEBLKRDFAIL

DSEBLKRDFAIL, Failed attempt to read block

DSE Error: This indicates that DSE could not read the block from the database file. This error may also be caused by attempts to reference blocks outside of the database. It is very likely that the database may have integrity errors or that the process-private data structures are corrupted.

Action: Make sure that the referenced block is less than the total blocks. If not, report this database cache error to the group responsible for database integrity at your operation.

DSEFAIL

DSEFAIL, DSE failed. Failure code: xxxx.

DSE Error: This indicates that DSE could not complete a database operation. xxxx contains failure codes for the four attempts. It is very likely that the database may have integrity errors or that the process-private data structures are corrupted.

Action: Report this database structure error to the group responsible for database integrity at your operation.

DSEINVLCLUSFN

DSEINVLCLUSFN, Specified function is invalid for clustered databases

DSE Error: This indicates that the DSE command (WCINIT or ALL used with the qualifiers RENEW or WCINIT) requested a cache reinitialization on a clustered database.

Action: This operation is managed automatically by the Cluster Control Program (CCP) on the first node to open the database file. Once the database is in use, all nodes must drop access in order to reinitialize the database cache.

DSENOFINISH

DSENOFINISH, DSE unable to finish all requested actions

DSE Error: This indicates that DSE was not able to complete the actions it was directed to perform.

Action: Refer to the associated message(s) for more information.

DSEONLYBGMM

DSEONLYBGMM FLU, xxxx is supported only for BG/MM access methods

DSE Warning: This indicates that the current region has an access method that is neither Buffered Globals nor Memory Mapped. DSE can not flush such a region.

Action: Before starting DSE, make sure that the Global Directory contains regions that have an access method of Buffered Globals or Memory Mapped.

DSEWCINITCON

DSEWCINITCON, No action taken, enter YES at CONFIRMATION prompt to initialize global buffers

DSE Warning: This indicates that DSE did not yet perform the operation that was initiated by the WCINIT command or the ALL command with either the WCINIT or RENEW qualifier because the operator did not confirm it.

Action: To perform these operations, enter YES at the CONFIRMATION prompt to verify your intention to perform this potentially disruptive operation. The DSE WCINIT command reinitializes shared memory structures for the current region.

DSEWCREINIT

DSEWCREINIT, Database cache reinitialized by DSE for region rrrr

DSE Information: This indicates a DSE operator action to rebuild the database cache for region rrrr.

Action: None required.

DSKSPACEFLOW

DSKSPACEFLOW, Disk space for file xxxx nearing maximum size. YYYY blocks available.

Run Time Warning: This indicates that disk space for the specified file system, on which the database / journal files are located, is almost full.

Action: Review and make disk space.

DUPTN

DUPTN, Duplicate transaction found [TN = xxxx] at offset aaaa in journal file yyyy

MUPIP Warning: This indicates that two different transactions have the same transaction number.

Action: Report the entire incident context to your GT.M support channel.

DUPTOKEN

DUPTOKEN, Token xxxx is duplicate in the journal file yyyy for database zzzz

MUPIP Error: This indicates that two transactions (TP or ZTP) have the same token (xxxx), in the specified journal file yyyy, violating the uniqueness of the ID that distinguishes transactions from one another. The result is that both transactions are considered broken and reported in the broken transactions extract file.

Action: Report the entire incident context to your GT.M support channel.

DVIKEYBAD

DVIKEYBAD, \$ZGETDVI("xxxx","yyyy") contains an illegal keyword

Run Time Error: This indicates that a \$ZGETDVI function encountered an invalid keyword. xxxx is the device. yyyy is the keyword.

Action: Verify the spelling of the keyword.

DYNUPGRDFAIL

DYNUPGRDFAIL, Unable to dynamically upgrade block 0xaaa in database yyy due to lack of free space in block

Runtime Error: There was not enough free space in the block to convert it in place to the current format during normal database access. This indicates that the DBCERTIFY database certification procedure was not properly carried out.

Action: Either mark the block free (making appropriate index changes) or downgrade the database and re-run DBCERTIFY (both phases).

DZTRIGINTRIG

DZTRIGINTRIG, \$ZTRIGGER() is not allowed inside trigger context. Trigger name: nnnn

Run Time Error: This message indicates an attempt to use the \$ZTRIGGER() function, which potentially modifies triggers, while executing code within the context of some trigger.

Action: Rework the code to modify or examine triggers so that it falls outside of trigger execution.

DZWRNOALIAS

DZWRNOALIAS, \$ZWRTAC cannot be aliased.

Compile Time Error: This indicates the argument for a SET * command attempted to assign a \$ZWRTAC* pseudo-variable as a alias.

Action: Correct the code in question - the \$ZWRTAC* is only useful in restoring context from ZSHOW or ZWRITE output and has very narrow capabilities.

DZWRNOPAREN

DZWRNOPAREN, \$ZWRTACxxx is not allowed inside a parenthesized SET target

Compile Time Error: This indicates the argument for a SET command attempted to assign a \$ZWRTAC* pseudo-variable within a parenthesized list of left-hand arguments.

Action: Correct the code in question - the \$ZWRAC* is only useful in restoring context from ZSHOW or ZWRITE output and has very narrow capabilities.

ECLOSTMID

ECLOSTMID, \$ECODE overflow, the first and last ecodes are retained, but some intervening ecodes have been lost

Run Time Warning: If the \$ECODE exceeds the maximum string length, references to it return only the codes for the earliest and latest errors separated by the code for ECLOSTMID, which indicates that suppression of intervening error codes has occurred to accommodate string length restrictions.

Action: Consider whether it would be appropriate to introduce code to SET \$ECODE= . This error is encountered either when the \$ETRAP error handling is recursing (and probably defective), or while using \$ZTRAP error handling that was coded prior to the introduction of \$ECODE.

ENDIANCVT

ENDIANCVT, Converted database file xxxx from yyyy endian to zzzz endian on a wwww endian system

MUPIP Information: When MUPIP ENDIANCVT has successfully completed, it displays the conversion information. This information includes the database file, its previous endian format, the new endian format, and the endian format that is native to the current system.

EORNOTFND

EORNOTFND, xxxx End of record not found

MUPIP Error: This indicates that LOAD encountered a database reference record containing an open parenthesis without a close parenthesis. xxxx is the record.

Action: Refer to the topic MUPIP LOAD Errors in About This Manual section of this manual.

EPOCHTNHI

EPOCHTNHI, At the EPOCH record at offset xxxx of yyyy transaction number [0xaaaa] is higher than database transaction number [0xbbbb]

MUPIP Error: This indicates that during turnaround point from where mupip applies logical records, backward recover found that epoch's transaction number is greater than the current database transaction number.

Action: Contact your system administrator and if necessary report the entire incident context to your GT.M support channel.

EQUAL

EQUAL, Equal sign expected but not found

Compile Time Error: This indicates that a SET or FOR specified the left side of an assignment statement but not the expected equal (=) sign.

Action: Look for a missing equal sign.

ERRCALL

ERRCALL, Error called from xxxx line yyyy

Run Time Error: This message provides additional diagnostic information related to accompanying messages.

Action: Review accompanying messages for additional information about the cause of this error. If necessary, report the entire incident context to your GT.M support channel.

ERRORSUMMARY

ERRORSUMMARY, Errors occurred during compilation

Compile Time Error: This indicates that GT.M encountered one or more errors during compilation.

Action: Review the individual error messages for further information. The compilation may have produced code that is usable as long as the execution path does not encounter the error(s).

ERRWETRAP

ERRWETRAP, Error while processing \$ETRAP

Run Time Error: This indicates that \$ETRAP contained invalid M code or caused a run time error.

Action: Check the \$ETRAP variable. To get more information about the errors- SET \$ZTRAP to empty string, and a temporary variable to the contents of \$ETRAP. Then, SET \$ETRAP to "BREAK" and XECUTE the temporary variable. It is best to keep the source code in \$ETRAP simple if \$ZTRAP is an empty string, since ETRAP specifies a string that GT.M invokes upon encountering an exception condition.

ERRWEXC

ERRWEXC, Error while processing exception string

Run Time Error: This indicates that an exception string contained invalid M code or caused a run-time error.

Action: Review the exception string and \$ZTRAP. To get more information about the errors, SET a temporary variable to the contents of the exception string. Then, SET the exception string to "BREAK" and XECUTE the temporary variable.

ERRWIOEXC

ERRWIOEXC, Error while processing I/O exception string

Run Time Error: This indicates that a device EXCEPTION string contained invalid M code or caused a run-time error.

Action: Review the exception string. To get more information about the errors, SET a temporary variable to the contents of the exception string and XECUTE the temporary variable. The EXCEPTION deviceparameter on an OPEN, USE, or CLOSE command defines an error handler for an I/O device.

ERRWZBRK

ERRWZBRK, Error while processing ZBREAK action string

Run Time Error: This indicates that a ZBREAK action contained invalid M code or caused a run-time error.

Action: Review the ZBREAK action string. To get more information about the errors, SET a temporary variable to the contents of the ZBREAK action string and XECUTE the variable. ZBREAK sets temporary break and trace points.

ERRWZINTR

ERRWZINTR, Error while processing \$ZINTERRUPT

Run Time Error: This indicates that a job interrupt had been signaled but that there was an error compiling the \$ZINTERRUPT string. This message is sent to the operator log facility and if at the direct mode prompt or executing a direct mode command to the user console.

Action: Correct the \$ZINTERRUPT to contain valid GT.M commands.

ERRWZTRAP

ERRWZTRAP, Error while processing \$ZTRAP

Run Time Error: This indicates that \$ZTRAP contained invalid M code or caused a run-time error.

Action: Verify the \$ZTRAP variable. To get more information about the errors, SET a temporary variable to the contents of \$ZTRAP. Then, SET \$ZTRAP to "BREAK" and XECUTE the variable. Make sure the source code in \$ZTRAP is simple because ZTRAP specifies a string that GT.M invokes when it encounters an exception condition.

EVENTLOGERR

EVENTLOGERR, Error in event logging subsystem

Run Time Error: This indicates that the user is unable to access the event logging shared library or an event logging routine within the shared library.

Action: Review accompanying messages for additional information.

EXCLUDEREORG

EXCLUDEREORG, Global: xxxx is present in the EXCLUDE option. REORG will skip the global.

MUPIP Warning: This indicates that MUPIP did not reorg the specified global because it was also mentioned in the EXCLUDE qualifier.

Action: Take out the global name from the EXCLUDE option and do not specify a global name in both the SELECT and EXCLUDE options.

EXECOM

EXECOM, Executing command file xxxx

GDE or DSE Information: This indicates that an @ command activated command file xxxx.

EXPR

EXPR, Expression expected but not found

Compile Time Error: This indicates that GT.M did not encounter a valid expression when it expected one.

Action: Look for missing expressions or extra delimiters, such as a space, comma, or colon.

EXTGBLDEL

EXTGBLDEL, Invalid delimiter for extended global syntax

Run Time Error: This indicates that the global reference started with a vertical bar (|) or left-bracket ([), indicating that it includes an environment specification (Global Directory). However, the environment specification did not terminate with either a vertical bar (|) or right-bracket (]), respectively.

Action: Insert the appropriate trailing delimiter for the environment specification or remove the environment specification.

EXTRACTCTRL

EXTRACTCTRL, User interrupt encountered during extract, halting

MUPIP Warning: This indicates that EXTRACT encountered either a <CTRL>-Y or two <CTRL>-C in quick succession during the course of its operation and aborted prior to normal completion.

Action: If the results of the EXTRACT are needed, reactivate it.

EXTRACTFILERR

EXTRACTFILERR, Error with extract file xxxx

MUPIP Error: This indicates that EXTRACT encountered an error when opening its output file: xxxx.

Action: Review the accompanying message(s) for additional information.

EXTRCLOSEERR

EXTRCLOSEERR, Error closing extract file xxxx

MUPIP Error: OpenVMS or RMS error closing the specified xtract output file, during the MUPIP EXTRACT.

Action: Refer to the accompanying message(s) for details and take appropriate action.

EXTRFAIL

EXTRFAIL, Extract failed for the global gggg. MUPIP INTEG should be run.

MUPIP Error: A MUPIP EXTRACT operation on the global gggg failed because of database consistency issues.

Run the MUPIP INTEG command to identify the database consistency issues. See [verify_database_integrity_11.html](#) for more information.

EXTRFMT

EXTRFMT, Extract error: bad format type. Must be ZWR, GO, or BINARY.

MUPIP Error: This indicates that EXTRACT could not create the sequential output file because of an invalid format specified.

Action: Change the format to one of the three supported formats.

EXTRIOERR

EXTRIOERR, Error writing extract file xxxx

MUPIP Error: OpenVMS or RMS errors writing to the specified extract output file, during the MUPIP EXTRACT.

Action: Refer to the accompanying message(s) for details and take appropriate action.

EXTSRCLIN

EXTSRCLIN, xxxx yyyy

Run Time Error: This indicates that there is an error in the external call table. The message indicates the line where GT.M found the error.

Action: Review the line listed in the message and correct the error.

EXTSRCLOC

EXTSRCLOC, At column xxxx, line yyyy, source module zzzz

Run Time Error: This indicates that there is an error in the external call table. The message indicates the line and the location within that line where the error is located.

Action: Review the listed line and location and correct the error.

FCHARMAXARGS

FCHARMAXARGS, Argument count of \$CHAR() function exceeded the maximum of 255

Compile Time Error: This indicates that a \$CHAR() function specified an argument that was not in the range of 0 to 255. This error is also reported by services that attempt to format data using \$CHAR() format.

Action: Look for large or negative \$CHAR() arguments and ensure that all the arguments contain valid ASCII codes.

FCNSVNEXPECTED

FCNSVNEXPECTED, Function or special variable expected in this context

Compile Time Error: This indicates that GT.M encountered a dollar sign in an expression that was not followed by a valid function or special variable name.

Action: Look for misspelled function and special variable names or a missing \$ in an extrinsic.

FILECREATE

FILECREATE, AAAA file xxxx created

MUPIP Information: This indicates that a file xxxx was created due to AAAA, where AAAA is lost transaction, broken transaction, or Journal Extract file.

Action: Look for the xxxx file for further relevant data.

FILEDEL

FILEDEL, File xxxx successfully deleted

MUPIP/Run Time Information: This indicates that GT.M or MUPIP has successfully deleted a file. This message is issued when a journal file in an inconsistent state found and deleted by the run-time system or MUPIP SET. This message is also issued by MUPIP JOURNAL RECOVER/ROLLBACK command when it deletes journal files that were created by a previously interrupted RECOVER/ROLLBACK command and are no longer necessary.

FILEDELFAIL

FILEDELFAIL, Deletion of file xxxx failed

Run Time/MUPIP warning: This indicates that GT.M or MUPIP failed to remove the specified journal file xxxx.

Action: Review the accompanying message(s) for additional information.

FILEEXISTS

FILEEXISTS, File xxxx already exists

MUPIP Error: This indicates that MUPIP discovered a file with the filename xxxx already existing and did not overwrite it while executing the specified command(s).

Action: Rename the already existing file xxxx and reissue the MUPIP command(s), or modify the MUPIP command to name (explicitly/implicitly) a file different from xxxx.

FILEIDGBLSEC

FILEIDGBLSEC, File ID in global section does not match with the database file

Run Time Error: When a GT.M process attaches to a database and finds the corresponding shared memory structures initialized already, it performs integrity checks on the shared memory contents to ensure that they correspond back to the database file. When the shared memory copy of the database file ID does not match with the actual file ID of the database, this error is issued.

Action: Perform a MUPIP RUNDOWN on that region. If it fails with the same FILEIDGBLSEC error, then the shared memory contents are corrupt. Consult your GT.M support channel before proceeding further.

FILEIDMATCH

FILEIDMATCH, Saved File ID does not match the current ID - the file appears to have been moved

Run Time Error: This indicates that the journal file identified by a database in turn identifies itself as belonging to another database. Since a journal file must have a one-to-one relationship with a database, the process cannot do updates on this region until the problem is resolved.

Action: Create a new journal file, make a backup if appropriate and resume work.

FILENAMETOOLONG

FILENAMETOOLONG, File name too long

Run Time/MUPIP Information: This indicates that GT.M or MUPIP has encountered a file name exceeding the maximum permissible length. In OpenVMS all logical names are expanded and the expanded name must be within the maximum file name length permitted by GT.M.

FILENOTCREATE

FILENOTCREATE, AAAA file xxxx not created

MUPIP Information: This indicates that the file xxxx was not created due to AAAA; where AAAA is lost transaction, or broken transaction, or Journal Extract file.

Action: Review accompanying messages for any further information. If there are no accompanying messages, it indicates that MUPIP did not find any lost or broken transactions for the corresponding extract file to be created.

FILENOTFND

FILENOTFND, File xxxx not found

Compile/Run Time Error: This indicates that GT.M could not locate the specified source file xxxx.

Action: Look for a misspelling of the file-specification or improper preparation of the environment. If xxxx is a source file, it could have been moved or modified since the object in the image was compiled. Use ZLINK to make the object match the source.

FILEPARSE

FILEPARSE, Error parsing file specification: xxxx

Run Time Error: This indicates a problem with the specification of file xxxx, or the path to it. If the file is a source or object file, a ZLINK command or \$ZROUTINES-related action encountered the error.

Action: Look for, and correct any typographical errors in the file-specification.

FILERENAME

FILERENAME, File xxxx is renamed to yyyy

Run Time Information: This indicates that an existing file xxxx has been renamed to yyyy so that a new file created with the original name does not overwrite the existing one. GT.M renames files during an automatic journal switch in case no explicit journal file name is specified, in which case the message is sent to the operator log. The utilities (MUPIP, GT.CM) rename files while opening log files or journal extract files and they send the message to the terminal. GT.M or utilities rename files only if the new file name specified already exists.

FILTERBADCONV

FILTERBADCONV, Bad conversion of transaction xxxx by filter

Run Time Error: This error is logged to the replication server log file. This indicates that the output of the user-supplied external replication filter for the transaction with journal sequence number xxxx is incorrect.

Action: Fix the filter. Restart the replication server with the fixed filter.

FILTERCOMM

FILTERCOMM, Error communicating transaction xxxx with the filter

Run Time Error: This error is logged to the replication server log file. This indicates that the replication server encountered an error writing transaction with journal sequence number xxxx to the user supplied external filter's input. The accompanying system error message gives more details.

Action: Stop the filter and restart the replication server with the filter. Report the entire incident context to your GT.M support channel.

FILTERNOTALIVE

FILTERNOTALIVE, Replication server detected that the filter is not alive while attempting to send transaction xxxx

Run Time Error: This error is logged to the replication server log file. This indicates that the replication server detected abnormal termination of the user supplied external filter while attempting to write transaction with journal sequence number xxxx to the filter's input.

Action: Determine the cause of the filter's abnormal termination. Fix the filter and restart the replication server with the fixed filter.

FMLLSTMISSING

FMLLSTMISSING, The formal list is absent from a label called with an actual list: xxxx

Run/Compile Time Error: This indicates that a DO attempted to transfer control with an actualist to a label xxxx that has no formallist.

Action: Look at the interface between the DO and the subroutine. Modify the actualist, formallist, and/or label as appropriate.

FMLLSTPRESENT ●

Last used version: **V5.4-002B**

FMLLSTPRESENT, The actual list is absent from a call to a label with a formal list: xxxx

Run/Compile Time Error: This indicates that a DO attempted to transfer control with no actuallist to a label xxxx that has a formallist.

Action: Look at the interface between the DO and the subroutine. Modify the actuallist, formallist, and/or label as appropriate.

FNARGINC

FNARGINC, Format specifiers to \$FNUMBER are incompatible: "xxxx"

Run Time Error: This indicates that a \$FNUMBER function specified a format containing codes xxxx, which is incompatible code.

Action: Look for the character code "P" or "p" with any character other than the code ",".

FNNAMENEG

FNNAMENEG, Depth argument to \$NAME cannot be negative

Run Time Error: This indicates that GT.M encountered a \$NAME() reference with the optional integer expression that is set to a negative number.

Action: Modify the routine to ensure that \$NAME() arguments are never negative.

FNOTONSYS

FNOTONSYS, Function or special variable is not supported by this operating system

Compile Time Error: This indicates that GT.M encountered a function or special variable it cannot process on the current operating system.

Action: Some functions are not appropriate to all operating environments. Contact your GT.M support channel if you have questions about how to accomplish a particular task.

FNUMARG

FNUMARG, \$FNUMBER format specifier xxxx contains an illegal character: yyyy

Run Time Error: This indicates that an \$FNUMBER function specified a format xxxx that contains invalid codes.

Action: Ensure that the format specifier in a \$FNUMBER function is a sequence of the code characters "P", "p", "T", "t", "+", "-", and ",".

FORCEDHALT

FORCEDHALT, Image HALTed by MUPIP STOP

Run Time Warning: This indicates that a GT.M process recognized the receipt of a MUPIP STOP command and is terminating. This command stops GT.M processes in an orderly fashion.

Action: Determine who initiated the MUPIP STOP and why they did so. Restart the process, if appropriate.

FORCTRLINDX •

Last used version: **V5.4-002A**

FORCTRLINDX, Using a FOR with an indexed control variable that's assigned values calculated in extrinsics is not currently safe

Compile/Run Time: GT.M currently has trouble managing the interaction between a subscripted FOR control variable and expressions for its maintenance that have side effects that might change the array holding the control variable. Such elements include extrinsics (\$\$), external calls (\$&) and \$INCREMENT(). To avoid possible process context corruption, the compiler issues this error when it encounters the threatening circumstances. Note that the error can also appear at run-time without the accompanying indications of line and column if the construct appears in an XECUTE, the control variable is specified with indirection or the compiler warning is ignored.

Action: Chose an unsubscripted FOR control variable and / or evaluate the side-effect element(s) before setting up the FOR.

FOROFLOW

FOROFLOW, FOR commands nested too deeply

Compile Time Error: This indicates that a single line contained more than 32 FOR statements.

Action: Rework the routine so FORs are not so deeply nested in a single line.

FREEMEMORY

FREEMEMORY, Error occurred freeing memory

Run Time Error: Indicates an internal problem with storage management. The error is usually accompanied by a secondary message, which lists the reason towards the request failure.

Action: Report the entire incident context to your GT.M support channel.

FREEZE

FREEZE, Region: xxxx is already frozen

MUPIP Information: MUPIP FREEZE or DSE CHANGE commands generate this error if the region is already frozen.

Action: If the existing FREEZE is appropriate, no further action is necessary. If you decide to remove the prior FREEZE, issue a FREEZE command with the OFF qualifier.

FREEZECTRL

FREEZECTRL, Control Y or control C encountered during attempt to freeze the database. Aborting freeze.

MUPIP Information: This indicates that the user aborted the MUPIP FREEZE or MUPIP BACKUP/NOONLINE command; while the command was attempting to freeze the already frozen database.

FREEZEERR

FREEZEERR, Error while trying to ffff region rrrr

MUPIP Error: This indicates an unsuccessful database freeze or unfreeze (ffff) operation on region rrrr.

Action: Look for accompanying text that explains the cause of the error and take appropriate action.

FSEXP

FSEXP, File specification expected but not found

Run Time Error: This indicates that a \$ZROUTINES-related action did not specify a valid file-specification.

Action: Look for missing brackets.

FSYNCTIMOUT

FSYNCTIMOUT, Timed out on fsync for journal file xxxx

Run Time Error: This indicates that the process has been unable to get the journal flush lock on the journal file for nearly two minutes and has timed out. It is very likely that another process is holding the journal flush lock, wanting to do an fsync() on the journal file and has not released it so long, which would suggest an issue with disk subsystem response times.

Action: Check the disk subsystem for a software or hardware problem.

FTOKERR

FTOKERR, Error getting ftok of the file xxxx

Run Time Error: This indicates that GT.M failed to take ftok of the database.

Action: Review the accompanying message(s) to identify the cause of the failure.

GBLEXPECTED

GBLEXPECTED, Global variable reference expected in this context

Run Time Error or MUPIP: This message indicates an attempt to apply a trigger definition to something other than a global variable.

Action: Review and correct the trigger definition.

GBLMODFAIL

GBLMODFAIL, Global variable Conflict Test failed. Failure code: xxxx.

Run Time Error: This indicates that a \$ZQGBLMOD function encountered an integrity error while restoring from a failover. It is very likely that the database may have integrity errors or that the process-private data structures are corrupted.

Action: Report this error to the group responsible for database integrity at your operation.

GBLNAME

GBLNAME, Either an identifier or a left parenthesis is expected after a ^ in this context

Compile Time Error: This indicates that GT.M encountered a circumflex in a valid location for a global variable name; however, the circumflex was not followed by a variable name or a left parenthesis.

Action: Look for unwanted circumflexes in expressions. Ensure that global variable names are valid.

GBLNOEXIST

GBLNOEXIST, Global xxxx no longer exists

MUPIP Information: The specified global variable does not exist in the database. This indicates that the global variable xxxx was present when MUPIP reorg started, but was killed when reorg was working on it.

GBLOFLOW

GBLOFLOW, Database segment is full

Run Time Error: This indicates that an error was encountered while extending the database file.

Action: Examine the accompanying message(s) for the cause of the error. If the error is due to lack of enough free space on the disk to fit the size of a database file, try performing a KILL of some nodes in the database to get free blocks in the existing allocated space (you may need to KILL several subscripted nodes before you can KILL a name node).

GBLSECNOTGDS

GBLSECNOTGDS, Global section xxxx is not a GT.M global section

Run Time Error: This indicates that when attempting to startup a database file, GT.M encountered an existing global section whose contents it did not recognize.

Action: Investigate whether you have a global section name conflict between GT.M and some other application. GT.M uses GT \$ as a prefix for all global section names that it creates. Make sure no other application in the system is using the same naming convention; it is very likely the global section contents are damaged. If necessary, report the entire incident context to your GT.M support channel.

GDCREATE

GDCREATE, Creating global directory File xxxx

GDE Information: This indicates that an EXIT command caused GDE to write a new Global Directory into file xxxx.

GDECHECK

GDECHECK, Internal GDE consistency check

GDE Fatal Error: This indicates that an internal consistency check failed.

Action: Look in the user's current working directory for a GDEDUMP.DMP context file that your GT.M support channel can examine to help determine the cause of the error. If necessary, report the entire incident context to your GT.M support channel.

GDINVALID

GDINVALID, Unrecognized Global Directory file format: xxx, expected label: yyy, found: zzz

Run Time Error: This indicates that a version of the global directory file xxx does not match with the version expected by GT.M. The file might have been created by an incompatible GT.M version..

Action: Compare the labels yyy and zzz. If the global directory was created by an earlier GT.M version, upgrade the file by loading and then saving the file using the GDE of the new GT.M version.

GDNOTSET

GDNOTSET, Global Directory not changed because the current GD cannot be written

GDE Information: This indicates that GDE could not complete a SETGD command because it could not verify the current Global Directory. This prevented a write of the current information.

Action: Either modify the current Global Directory or abandon it by adding the QUIT qualifier to the SETGD command.

GDREADERR

GDREADERR, Error reading Global Directory: xxxx

GDE Information: This indicates that GDE encountered an error when it attempted to read the existing Global Directory in file xxxx.

Action: Review the accompanying message(s) for additional information.

GDUNKNFMT

GDUNKNFMT, xxxx is not formatted as a global directory

GDE Information: This indicates that GDE could not load the specified file xxxx because it is not a valid Global Directory file. GDE aborts the load after it issues this message.

Action: Verify that the file is valid and look for typographical errors. Something other than GT.M or its utilities may have written to the Global Directory file or created a file with a name that coincides with the one specified by GTM\$GBLDIR / gtmgbldir.

GDUPDATE

GDUPDATE, Updating Global Directory File xxxx

GDE Information: This indicates that an EXIT or SETGD command caused GDE to write a new version of the existing Global Directory in file xxxx.

GDUSEDEFS

GDUSEDEFS, Using defaults for Global Directory xxxx

GDE Information: This indicates that GDE did not find an existing Global Directory using the logical name GTM\$GBLDIR / gtmgbldir. As a result, it is starting the session with default values.

GETCWD

GETCWD, Error getting current working directory for file xxxx

Run Time/MUPIP Error: This indicates that GT.M could not get the full path of current working directory.

Action: Review up the accompanying message(s) for additional information.

GETSOCKNAMERR

GETSOCKNAMERR, Getting the socket name failed from getsockname(): (errno==aaaa) xxxx

Run Time Error: This indicates getsockname() system call, which retrieves the locally bound address of the specified socket, failed.

Action: Review the accompanying messages and error code.

GETSOCKOPTERR

GETSOCKOPTERR, Getting the socket attribute xxxx failed: (errno == yyyy) zzzz

Run Time Error: This indicates that an attempt to determine a socket's attributes failed.

Action: Consider the OPEN or USE deviceparameters, and the error code.

GOQPREC

GOQPREC, Numeric precision in key error: Blk #xxxx, Key #yyyy. Record not loaded.

MUPIP Information: This indicates that GT.M was unable to precisely represent a key in the GOQ input file to a MUPIP LOAD.

Action: Examine the key on the source system, modify it, and repeat the process, or manually enter the modified record into GT.M.

GTMASSTERT

GTMASSTERT, xxxx - assert failed yyyy line zzzz

Compile/Run Time Fatal Error: This indicates that a design assumption failed at the location specified.

Action: Preserve the core (dump) files and report the entire incident context to your GT.M support channel for further analysis. If appropriate, verify database integrity by using the -FAST qualifier.

GTMASSERT2

GTMASSERT2, GT.M eeee - Assert failed LLLL for expression (eeee)

Compile/Run Time Fatal Error: This indicates a design assumption failed at the location LLLL because the expression eeee was FALSE.

Action: Preserve the core (dump) files and report the entire incident context to your GT.M support channel for further analysis. If appropriate, verify database integrity by using the -FAST qualifier.

GTMCHECK

GTMCHECK, Internal GT.M error. Report to GT.M Support.

Compile/Run Time Fatal Error: This indicates that a design assumption failed at the location specified.

Action: Preserve the core (dump) files and report the entire incident context to your GT.M support channel for further analysis. If appropriate, verify database integrity by using the -FAST qualifier.

GTMDISTUNDEF

GTMDISTUNDEF, Environmental variable \$gtm_dist is not defined

Run Time/MUPIP/LKE/DSE Error: This indicates that the environment variable gtm_dist, is not defined for all processes attempting to use (a particular version of) GT.M.

Action: Define the environment variable.

GTMDUMPFAIL

GTMDUMPFAIL, Could not create DUMP FILE

Run Time Error: This indicates that an unanticipated error caused GT.M to attempt to create a context file GTMDUMP.DMP in the SYS\$LOGIN directory of the process. GT.M could not create the file.

Action: Verify that privileges are appropriately assigned and that adequate disk space exists (usually less than 100 host blocks).

GTMERREXIT

GTMERREXIT, GTM image has exited with errors

Runtime error: Seen when GT.M on VMS is exiting due to a FATAL error (previously displayed) but wishes to exit with error status and not generate a dump file (if SET PROC/DUMP were in effect).

Action: See previous FATAL error.

GTMSECSHR

GTMSECSHR, xxxx Error during GTMSECSHR operation

Run Time Error: This indicates that user privileges do not allow access to GTMSECSHR; or GTMSECSHR was not properly installed.

Action: Verify that SECSHR is properly installed and review user privileges in the SECSHR log file.

GTMSECSHRCHDIRF

GTMSECSHRCHDIRF, gtmsecshr unable to chdir to its temporary directory (dddd)

GTMSECSHR process error: GTMSECSHR is not able to change directory to its temporary directory, dddd.

GTMSECSHR Error: The UNIX gtmsecshr process, which assists other process with cross-user signaling and similar things, uses a temporary directory determined by the operating system defined temporary directory (typically /tmp or /var/tmp) when it needs to save a core file, but it was unable to find that directory.

Action: Verify that the environment provides the desired dddd, that dddd exists and that it is a directory.

 the permissions on dddd should not matter (as long as it is a directory) since GTMSECSHR runs as root.

GTMSECSHRTMPPATH

GTMSECSHRTMPPATH, gtmsecshr path is pppp

Information Message: GT.M displays this message when different users of an instance of GT.M connect using a socket or a semaphore and when gtmsecshr is started and it detects an existing gtmsecshr. pppp indicates the gtm_tmp path set in the clients. Gtmsecshr inherits the path from the first GT.M process that uses its services.

Action: If different clients of the same instance of GT.M are using different gtmsecshr paths, then set a common value for the environment variable gtm_tmp for all users of an instance of GT.M, then stop and restart the processes that were using incorrect paths. If gtmsecshr itself has the incorrect path, all processes that are using that incorrect path must be stopped first - then stop gtmsecshr with a kill command.

GTMSECSHRDEFLOG

GTMSECSHRDEFLOG, \$gtm_log is either undefined or not defined to an absolute path, so gtm_log is set the default xxxx

Run Time Information: This indicates that GTMSECSHR has selected the default log file described in the message because the environment variable was not defined or had an unsuitable definition.

Action: When specifying the log file, be sure to select a full path and define the environment variable gtm_log properly.

GTMSECSHRFORKF

GTMSECSHRFORKF, GTMSECSHR server unable to fork off a child process

Run Time Error: This indicates that a GTMSECSHR was unsuccessful in starting because it was unable to create an independent process to run as a daemon.

Action: Refer to the associated message(s) for more information.

GTMSECSHRLOGF

GTMSECSHRLOGF, XXXX - YYYY; Error while creating GTMSECSHR log file

Run Time Warning: This indicates that the [UNIX] GTMSECSHR daemon was not able to create its log file.

Action: Check the accompanying message(s) for additional information. Check gtm_log environment variable.

GTMSECSHRLOGSWH

GTMSECSHRLOGSWH, Error switching GTMSECSHR log file

Run Time Error: This indicates that an operator attempt to start a new log file failed.

Action: Refer to the associated message(s) for more information.

GTMSECSHROPCMP

GTMSECSHROPCMP, GTMSECSHR operation may be compromised

Run Time Error: This indicates that GTMSECSHR could not acquire the privileges required to assist more than a restricted set of processes.

Action: If this is the proper mode of operation, ignore the warning. Normally GTMSECSHR should be a setuid executable owned by root.

GTMSECSHRPERM

GTMSECSHRPERM, The GTMSECSHR module in \$gtm_dist does not have the correct permission and UID

Run Time Warning: This indicates that a client did not start a GTMSECSHR because the executable was not owned by root and did not have setuid permission.

Action: Arrange to provide the GTMSECSHR executable with the proper characteristics.

GTMSECSHRRECVF

GTMSECSHRRECVF, GTMSECSHR receive on server socket failed

Run Time Error: This indicates that a receive operation failed in GTMSECSHR.

Action: Refer to the associated message(s) for more information.

GTMSECSHRSCSEL

GTMSECSHRSCSEL, GTMSECSHR select on socket failed

Run Time Error: This indicates that a select operation failed in GTMSECSHR.

Action: Refer to the associated message(s) for more information.

GTMSECSHRSEND

GTMSECSHRSEND, GTMSECSHR send on server socket failed

Run Time Error: This indicates that a socket operation failed in a GTMSECSHR.

Action: Refer to the associated message(s) for more information.

GTMSECSHRSGID

GTMSECSHRSGID, GTMSECSHR server setGID to root failed

Run Time Error: This indicates that the setgid operation failed during GTMSECSHR startup.

Action: Refer to the associated message(s) for more information.

GTMSECSHRSHUTDN

GTMSECSHRSHUTDN, GTMSECSHR process has received a shutdown request. Shutting down.

Run Time Information: This indicates that the GTMSECSHR daemon has shutdown.

GTMSECSHRSOCKET

GTMSECSHRSOCKET, xxxx - yyyy; Error initializing GTMSECSHR socket

Run Time Error: This indicates that a GT.M process or GTMSECSHR with PID yyyy was unable to open a socket for communication with either the server or client.

Action: Refer to the associated message(s) for more information.

GTMSECSHRSRVF

GTMSECSHRSRVF, xxxx - yyyy; Attempt to service request failed

Run Time Error: This indicates that a GT.M process with PID yyyy was unable to communicate with GTMSECSHR.

Action: This message is displayed when a process is requesting service and no GTMSECSHR process exists. One will be automatically started and the communication re-tried. This is an infrequent event: once the GTMSECSHR process is started, it remains for over an hour after processing the last request. It would also be unusual to see this message multiple times in a short period (minutes) for the same process. This message is also displayed when there may be a more serious problem (see the action section of an associated message GTMSECSHRTMPPATH).

GTMSECSHRSRVFIL

GTMSECSHRSRVFIL, xxxx: yyyy; Attempt to service request failed. Client ID: zzzz, mesg type: aaaa, file: bbbb

Run Time Warning: This indicates that GTMSECSHR failed to delete a UNIX domain socket file.

Action: Review accompanying message(s) for information on why GTMSECSHR could not delete the file.

GTMSECSHRSRVFID

GTMSECSHRSRVFID, xxxx: yyyy - Attempt to service request failed. Client ID: zzzz, mesg ID: aaaa, mesg code: bbbb.

Run Time Warning: This indicates that the GTMSECSHR was unable to complete the request of GT.M client.

Action: Examine the information in the message to see whether the message is appropriate to the environment; examine the environment and correct any inappropriate set up (such as the privileges available for GTMSECSHR).

GTMSECSHRSSIDF

GTMSECSHRSSIDF, GTMSECSHR server setSID failed

Run Time Error: This indicates that the setsid operation failed during GTMSECSHR startup.

Action: Refer to the associated message(s) for more information.

GTMSECSHRSTART

GTMSECSHRSTART, xxxx - yyyy; GTMSECSHR failed to startup

Run Time Warning: This indicates that GTMSECSHR startup failed.

Action: Refer to the associated message(s) for more information.

GTMSECSHRSUIDF

GTMSECSHRSUIDF, GTMSECSHR server setUID to root failed

Run Time Error: This indicates that the setuid operation failed during GTMSECSHR startup.

Action: Refer to the associated message(s) for more information.

GTMSECSHRTMOUT

GTMSECSHRTMOUT, GTMSECSHR exiting due to idle timeout

Run Time Information: This indicates that the GTMSECSHR had been idle long enough to time out and terminate.

Action: No action is required, another GTMSECSHR is started when it is needed.

GVDATAFAIL

GVDATAFAIL, Global variable \$DATA function failed. Failure code: xxxx

Run Time Error: This indicates that a \$DATA function encountered a database problem. xxxx contains the failure codes for the four attempts. It is very likely that the database may have integrity errors or that the process-private data structures are corrupted.

Action: Report this database error to the group responsible for database integrity at your operation.

GVDATAGETFAIL

GVDATAGETFAIL, Global variable DATAGET sub-operation (in KILL function) failed. Failure code: cccc.

Run Time Trigger Error: The target node for a KILL operation could not present its state to the trigger logic due to a database problem. cccc contains the failure codes for the failed attempts. The database may have integrity errors or the process-private data structures may be corrupted.

Action: Report this database error to the group responsible for database integrity at your operation.

GVFAILCORE

GVFAILCORE, A core file is being generated for later analysis if necessary

Run Time Error: This is an operator log-only message, which indicates that a core (dump) is being generated for the immediately preceding xxxxFAIL error message.

Action: Report this database error to the group responsible for database integrity at your operation. If the cause of the xxxxFAIL message is not otherwise known (for example, database damage due to recent system crash), the produced core will contain information that your GT.M support channel can use to determine the source of failure (UNIX only).

GVGETFAIL

GVGETFAIL, Global variable retrieval failed. Failure code: xxxx.

Run Time Error: This indicates that a database lookup of a global variable encountered an error. xxxx contains the failure codes for the four attempts. It is very likely that the database may have integrity errors or that the process-private data structures are corrupted.

Action: Report this database error to the group responsible for database integrity at your operation.

GVINCRFAIL

GVINCRFAIL, <Global variable \$INCR failed. Failure code: xxxx>

Run Time Error: This indicates that a \$INCREMENT command encountered a database problem when it attempted to update a global variable. xxxx contains the failure codes for the four attempts. It is very likely that the database may have structural damage or that the process-private data structures are corrupted.

Action: Report this database error to the group responsible for database integrity at your operation.

GVINCRISOLATION

GVINCRISOLATION, <\$INCREMENT cannot be performed on global xxxx as it has NOISOLATION turned ON>

Run Time Error: Global xxxx has NOISOLATION turned ON (through a VIEW "NOISOLATION" command). \$INCREMENT is currently not supported for such globals.

Action: Change the application either to turn OFF NOISOLATION on the global or not use \$INCREMENT on it.

GVINVALID

GVINVALID, xxxx Invalid global name

MUPIP Error: This indicates that LOAD encountered invalid global name xxxx in the input stream.

Action: Refer to the topic MUPIP LOAD Errors in About This Manual section of this manual.

GVIS

GVIS, Global variable: xxxx

Run Time Information: This message identifies a global variable.

Action: Refer to the accompanying message(s) for more information.

GVKILLFAIL

GVKILLFAIL, Global variable kill failed. Failure code: xxxx.

Run Time Error: This indicates that a KILL of a global variable encountered a database problem. xxxx contains the failure codes for the four attempts. It is very likely that the database may have integrity errors or that the process-private data structures are corrupted.

Action: Report this database error to the group responsible for database integrity at your operation.

GVNAKED

GVNAKED, Illegal naked global reference

Run Time Error: This indicates that the naked indicator was referenced before any named global reference or after an event that left it undefined.

Action: Review naked indicator references and correct them, if necessary. For example, the naked indicator cannot be the first global symbol referenced.

GVNAKEDEXTNM

GVNAKEDEXTNM, Cannot reference different Global Directory in a naked reference

Compile Time Error: This indicates that a global variable reference used the environment syntax but did not specify a name.

Action: Verify that the environment specifies a full global name.

GVNEXTARG

GVNEXTARG, Argument to global variable \$NEXT must be subscripted

Compile Time Error: This indicates that an attempt was made to use an un-subscripted global or local variable as the argument for a \$NEXT() function. In contrast to \$ORDER(), which can operate on un-subscripted names, \$NEXT() requires subscripted names.

Action: Use \$ORDER() or revise the code.

GVORDERFAIL

GVORDERFAIL, Global variable \$ORDER or \$NEXT function failed. Failure code: xxxx.

Run Time Error: This indicates that a \$ORDER or \$NEXT function encountered a database problem. xxxx contains the failure codes for the four attempts. It is very likely that the database may have integrity errors or that the process-private data structures are corrupted.

Action: Report this database error to the group responsible for database integrity at your operation.

GVPUTFAIL

GVPUTFAIL, Global variable put failed. Failure code: xxxx.

Run Time Error: This indicates that a SET command encountered a database problem when it attempted to update a global variable. xxxx contains the failure codes for the four attempts. It is very likely that the database may have integrity errors or that the process-private data structures are corrupted.

Action: Report this database error to the group responsible for database integrity at your operation.

GVQUERYFAIL

GVQUERYFAIL, Global variable \$QUERY function failed. Failure code: xxxx.

Run Time Error: This indicates that a \$QUERY function failed. xxxx contains the failure codes for the four attempts. It is very likely that the database may have integrity errors or that the process-private data structures are corrupted.

Action: Report this database error to the group responsible for database integrity at your operation.

GVQUERYGETFAIL

GVQUERYGETFAIL, Global variable QUERY and GET failed. Failure code: xxxx.

Run Time Error: This indicates that database query and Lookup in the same atomic transaction encountered a problem. xxxx contains the failure codes for the four attempts.

Action: Contact the system administrator and if needed report to your GT.M support channel.

GVREPLERR

GVREPLERR, Error replicating global in region xxxx

Run Time Error: This indicates that the database system successfully updated a global node on the primary copy of the database but it encountered an error making the same update in a replicated copy.

Action: Examine any secondary error. Investigate whether the problem is with the disk where the copy is located or with the communications system to the secondary copy, if it is remote. Correct the problem and resynchronize the copies.

GVRUNDOWN

GVRUNDOWN, Error during global database rundown

Run Time Error: This indicates that the process encountered an error when it attempted to RUNDOWN all database files as part of image termination.

Action: Report this database error to the group responsible for database integrity at your operation.

GVSUBOFLOW

GVSUBOFLOW, Maximum combined length of global subscripts exceeded

Run Time Error: This indicates that a global variable reference specified a total subscript length that exceeds the maximum length specified in the Global Directory for the current region.

Action: Ensure that the subscript is correct. Then, use DSE DUMP and the FILEHEADER qualifier to examine the key size for the region.

GVUNDEF

GVUNDEF, Global variable undefined: xxxx

Run Time Error: This indicates that the program attempted to evaluate an undefined global variable.

Action: Review the program flow and the preparation of the environment.

GVZPREVFAIL

GVZPREVFAIL, Global variable \$ZPREVIOUS function failed. Failure code: xxxx

Run Time Error: This indicates that a \$ZPREVIOUS function encountered a database problem. xxxx contains the failure codes for the four attempts. It is very likely that the database may have integrity errors or that the process-private data structures are corrupted.

Action: Report this database error to the group responsible for database integrity at your operation.

GVZTRIGFAIL

GVZTRIGFAIL, ZTRIGGER of a global variable failed. Failure code: cccc.

Run Time Error: A ZTRIGGER command failed because of problems in the database. cccc is a list of four codes indicating the reason for the failure on each of the attempts to commit the ZTRIGGER action.

Action: Report this database error to the group responsible for database integrity at your operation.

HTEXPFAIL ●

Last used version: **V5.3-001A**

HTEXPFAIL, Hash table expansion failed for lack of memory

Run Time/MUPIP Error: The hash table, an internally expanding data structure maintained by GT.M, has exceeded its maximum capacity. In GT.M, each unique local variable name uses up some hash table space. In MUPIP, it is backward recovery (or rollback) that might encounter this error. Here each TP transaction that is encountered in the backward processing phase of recovery uses up some hash table space. In either case, it is more likely that a process will run out of virtual memory much before it reaches the maximum hashtable capacity.

Action: Increase process memory quotas to increase available process virtual memory. Reduce the number of unique local variable names referenced by the GT.M process. For MUPIP backward recovery/rollback, reduce the number of TP transactions encountered in the backward processing phase by using a later timestamp in the SINCE_TIME qualifier or higher RESYNC_SEQNO for rollback.

HTOFLOW

HTOFLOW, Hash table overflow, local or region name space exceeded

Run Time/MUPIP Error: This indicates that the hash table contains too many local names or region names.

Action: Reduce the number of local and region name entries in the table.

HTSHRINKFAIL

HTSHRINKFAIL, Hash table compaction failed to allocate new smaller table due to lack of memory

Runtime Error: GT.M found an internal hash table over-allocated but was unable to reduce its size because the process memory was too large to allocate a new smaller table; GT.M must allocate the new table before it can release the old table because it must copy the contents out of the "too-large" table into the smaller one. After this warning, the process continues running with the larger table.

Action: Investigate whether the process size can be reduced, or the available memory increased.

ICUERROR

ICUERROR, ICU returned status ssss which is either unrecognized or inconsistent with the operating context

Run-time error: The open-source ICU module which GT.M uses for some Unicode processing return an error code ssss that GT.M did not recognize as valid for the current context.

Action: Consult the ICU documentation and / or refresh the ICU library with a known correct version.

ICUSYMNOTFOUND

ICUSYMNOTFOUND, Symbol xxxxx not found in ICU libraries. ICU needs to be built with symbol-renaming disabled or gtm_icu_version environment variable needs to be specified

Runtime Error: ICU version installed on the machine is built with symbol renaming and gtm_icu_version has not been defined

Action: Build ICU without symbol renaming or set gtm_icu_version environment variable to point to an appropriate ICU version.

ICUVERLT36

Type 1 - \$gtm_icu_version is aaa.bbb. ICU version greater than or equal to 3.6 should be used

Type 2 - libicuio has version aaa.bbb. ICU version greater than or equal to 3.6 should be used

Run time error: This message indicates an attempt to use an ICU version less than 3.6 with GT.M or utilities like MUIP or DSE.

Action: Upgrade ICU version to at least 3.6

IFBADPARAM

IFBADPARAM, External Interface Bad Parameter

Run Time Error: This indicates that an external routine could not access a GT.M database library routine because it had an invalid parameter in its call argument list. The GT.M database library routines allow an external routine to access a GT.M database.

Action: Look for and correct any typographical errors in the call format for the GT.M library routine.

IFNOTINIT

IFNOTINIT, External Interface must first call GTM\$INIT or M routine

Run Time Error: This indicates that an external routine could not access a GT.M database library routine because it did not call the GTM\$INIT library routine first. The GTM\$INIT library routine initializes the GT.M run-time environment.

Action: Call GTM\$INIT or a GT.M M routine before calling any other database access library routine.

IGNBMPMRKFREE

IGNBMPMRKFREE, Ignoring bitmap free-up operation for region rrrr (dddd) due to concurrent ONLINE ROLLBACK

Run Time Information: A multi-node KILL bit map cleanup operation detected a concurrent online rollback in region rrrr mapped to database file dddd, and abandoned the cleanup, possibly leaving incorrectly marked busy errors.

Action: If there are incorrectly marked busy errors, match them with this cause and clean them up using DSE.

ILLCHAR

ILLCHAR, xxxx is not a legal character in this context

GDE Information: This indicates that GDE encountered the invalid character xxxx in its command input stream. This character should never appear in the context in which it was found.

Action: Review and re-enter a valid command sequence.

ILLESOCKBFSIZE

ILLESOCKBFSIZE, The specified socket buffer size is xxxx, which is either 0 or too big

Run Time Error: This indicates that the OPEN command specified an inappropriate buffer size.

Action: Revise the command.

IMAGENAME

IMAGENAME, The executing module name should be xxxx instead of yyyy

Run Time Error: This indicates that the executable invoked should have been named xxxx instead of its current name yyyy.

Action: Revisit the GT.M installation.

INDEXTRACHARS

INDEXTRACHARS, Indirection string contains extra trailing characters

Compile Time Error: This indicates that an indirection string ends with a syntactically incorrect sequence.

Action: Look for extra trailing characters in the indirection string.

INDMAXNEST

INDMAXNEST, Maximum nesting of indirection expressions exceeded

Run Time Error: This indicates that indirection nesting required more space than GT.M provides. The space needed by GT.M is a function of the number, type and complexity of the indirect expressions being nested.

Action: Review the program logic and reduce the amount of nested indirection in the routine.

INDRMAXLEN

INDRMAXLEN, Maximum length xxxxx of an indirection argument was exceeded

Run Time Error: This indicates that an indirection or XECUTE used a value that exceeded the maximum length for a source code element.

Action: Review the the code to shorten the length of the XECUTE or indirection string.

INPINTEG

INPINTEG, Input integrity error -- aborting load

GDE Fatal Error: This indicates that GDE is aborting the session because integrity errors prevented it from loading the specified Global Directory. GDE usually displays this message with other error messages. GDE aborts the load after issuing this message.

Action: Review the accompanying message(s) for additional information. Verify whether the command specified the intended file. Something other than GT.M and its utilities probably wrote to a Global Directory file or created a file with a name identical to the one specified by GTM\$GBLDIR / gtmgbldir.

INSFFBCNT

INSFFBCNT, Insufficient byte count quota left for requested operation

Run Time Error: This indicates that an OPEN or JOB command could not establish a mailbox because it would exceed the process OpenVMS BYTLM.

Action: Review the I/O flow of the program and make adjustments to minimize concurrently open buffered I/O devices, or talk to your system manager about increasing the user BYTLM.

INSNOTJOINED

INSNOTJOINED, Replicating Instance RRRR is not a member of the same Group as Instance IIII

MUPIP Error: Issued by a Receiver Server or a MUPIP JOURNAL -ROLLBACK -FETCHRESYNC on instance RRRR attempted to connect with Instance IIII, and found they are members of different Groups. Only Supplementary Instances started with -UPDOK can accept updates from a different Group.

Action: Analyze the move and if appropriate, reinitialize the instance that is moving from one Group to another.

INSROLECHANGE

INSROLECHANGE, Supplementary Instance SSSS and non-Supplementary Instance IIII belong to the same Group

MUPIP Error: Issued by a Receiver Server or a MUPIP JOURNAL -ROLLBACK -FETCHRESYNC on Supplementary Instance SSSS attempted to connect to non-Supplementary Instance IIII, but found they have the same Group identification. Because supplementary and non-Supplementary Instances cannot belong to the same Group, one of these instances must have changed roles without appropriate re-initialization.

Action: Either reinitialize the instance that is changing roles or revert the inappropriate role change.

INSUNKNOWN

INSUNKNOWN, Supplementary Instance SSSS has no instance definition for non-Supplementary Instance IIII

MUPIP Error: Issued by a Receiver Server or a MUPIP JOURNAL -ROLLBACK -FETCHRESYNC on Supplementary Instance SSSS, started with -UPDOK, attempted to connect to non-Supplementary Instance IIII, but found it has no matching instance information.

Action: Take a backup of the database and replication instance file from a current instance on the non-Supplementary Group, load the backup data on the Supplementary Instance and start the Receiver Server on the supplementary instance using -UPDATERESYNC=<instbak.repl> where instbak.repl is the backup of the replication instance file taken along with the database backup.

INVTRCGRP

INVTRCGRP, Invalid trace group specified in \$gtm_trace_groups: gggg

Run Time Error: The process startup environment attempted to activate a diagnostic tracing facility but specified a group name of gggg and there is currently no such group.

Action: Check with your GT.M support channel for the currently available group names.

INTEGERRS

INTEGERRS, Database integrity errors

MUPIP Error: This indicates that INTEG encountered one or more errors in the database file.

Action: Review the accompanying errors for more information, and report this database error to the group responsible for database integrity at your operation.

INVACCMETHOD

INVACCMETHOD, Invalid access method

MUPIP Error: This indicates that the user specified an invalid access method in a MUPIP SET command.

Action: This command can only use Memory Map (MM) or Buffered Globals (BG) access methods.

INVADDRSPEC

INVADDRSPEC, Invalid IP address specification

Run Time Error: This indicates that there is no valid response at the IP address specified.

Action: Verify the IP address.

INVALIDRIP

INVALIDRIP, Invalid read-in-progress field in Cache Record. Resetting and continuing. Region: xxxx.

Run Time Error: This indicates that the read-in-progress field corresponding to a particular global buffer had an invalid value. The read-in-progress field usually indicates whether this global buffer is currently being read into from disk or not, and hence takes on two values only. Whenever the field takes on any value outside of these two, GT.M detects the situation and corrects it in addition to logging this incident in the operator log.

Action: Report the operator log message with any other relevant information to your GT.M support channel.

INVBITLEN

INVBITLEN, Invalid size of the bit string

Run Time Error: This indicates that an attempt was made to create a bit string of size less than 1 or more than 253,952 bits.

Action: Modify the code so it adheres to the permitted range.

INVBITPOS

INVBITPOS, Invalid position in the bit string

Compile/Run Time Error: This indicates that GT.M encountered a bit position argument to a \$ZBITGET or \$ZBITSET function that exceeded the length of the bit string, or was less than one (1).

Action: Use \$ZBITLEN() to modify the code so the bit reference falls within the allocated length of the bit string.

INVBITSTR

INVBITSTR, Invalid bit string

Run Time Error: This indicates that an attempt was made to use a bit string with a leading character that was not within the permitted range of values.

Action: Determine the source for the first character of the bit string and modify the method of its creation to limit possible values to the valid range of 0 to 7.

INVCMD

INVCMD, Invalid command keyword encountered

Compile Time Error: This indicates that the program attempted to use an invalid keyword where a command was expected.

Action: Look for typographical errors or improper command abbreviations.

INVCTLMNE

INVCTLMNE, Invalid control mnemonics

Run Time Error: This indicates that the system found a hardware incompatibility.

Action: Verify the TCP settings and check hardware manuals for correct settings and compatibility.

INVDLRCVAL

INVDLRCVAL, Invalid \$CHAR() value.

Runtime Error: The \$CHAR() function triggers this error if its arguments contains an invalid code-point. According to the Unicode Standard version 5.0, invalid code-points include the following sets:

1. The "too big" code-points (those greater than the maximum U+10FFFF).
2. The "surrogate" code-points (in the range [U+D800, U+DFFF]) which are reserved for UTF-16 encoding.
3. The "non-character" code-points that are always guaranteed to be not assigned to any valid characters. This set consists of [U+FDD0, U+FDEF] and all U+nFFFE and U+nFFFF (for each n from 0x0 to 0x10).

Action: Specify argument in the range of valid Unicode code-points.

INVECODEVAL

INVECODEVAL, Invalid value for \$ECODE (xxxx).

Run Time Error: This indicates that an attempt was made to assign \$ECODE an invalid value. Such an action modifies \$ECODE to have a valid value indicating this error, which triggers an error trap.

Action: Revise the SET \$ECODE value complies with the required specification of the error codes of the format ,Mnnn,Zxxx,Uxxx,. The error codes in the form of ,Mnnn,, ,Zxxx, and ,Uxxx, represent ANSI standard codes, implementation-specific codes and end-user defined codes respectively.

INVEORLIM

INVEORLIM, Invalid ERROR_LIMIT qualifier value. Must be at least zero

MUPIP Error: This indicates that the value assigned to the ERROR_LIMIT qualifier is negative (less than zero).

Action: Assign a value greater than zero (0) for ERROR_LIMIT qualifier.

INVFCN

INVFCN, Invalid function name

Compile Time Error: This indicates that an expression contained a string of the form "\$name(...)", but "name" was not a valid function name.

Action: Look for typographical errors, improper function name abbreviation, or a missing \$ in an extrinsic.

INVGBLDIR

INVGBLDIR, Invalid Global Directory spec: xxxx. Continuing with yyyy.

GDE Information: This indicates that the Global Directory xxxx specified by GTM\$GBLDIR / gtmgbldir or by SETGD and the qualifier FILE= is not a valid file-specification.

Action: When this error occurs, GDE uses the default specification of the current process default directory. Continue with the default and rename the result after leaving GDE, or change the specification with a SETGD command and the FILE= qualifier.

INVGLOBALQUAL

INVGLOBALQUAL, Error in GLOBAL qualifier : Parse error at offset xxxx in yyyy

MUPIP Error: This indicates a syntax error in GLOBAL qualifier value yyyy at offset xxxx.

Action: Specify correct value for GLOBAL.

INVTMEXIT

INVTMEXIT, Inappropriate invocation of gtm_exit. gtm_exit cannot be invoked from external calls.

Run Time/Call-in Error: This indicates that the call-in shut-down function gtm_exit() has been called from an external call C function. Since the GT.M run-time system must be operational even after the external call function returns, gtm_exit() is meant to be called only once during a process lifetime, and only from the base C/C++ program when GT.M functions are no longer required by the program.

Action: Remove all invocations of gtm_exit() from external call functions.

INVICUVER ●

Last used version: **V5.3-004**

INVICUVER, xxx not found in the ICU libraries. ICU version 3.6.x must be used.

Runtime Error: GT.M triggers this error when it encounters an incompatible ICU version.

Action: Install ICU version 3.6.x. See "V5.2-000 Release Notes" for instructions on installing ICU Version 3.6.x on a supported platform.

INVIDQUAL

INVIDQUAL, Invalid ID qualifier value xxxx

MUPIP Error: This indicates that an invalid value xxxx was assigned to the ID qualifier.

Action: Assign a valid value for ID qualifier.

INVINTMSG

INVINTMSG, Invalid interrupt message received

GT.CM Server Error: An invalid interrupt request was received. This may indicate a network problem.

Action: Check the DECnet error logs and other network components.

INVMEMRESRV

INVMEMRESRV ,Could not allocate GT.M memory reserve (xxxx)

All images: The memory reserver requested by the xxxx UNIX environment variable (\$gtm_memory_cache) or the VMS logical name (GTM_MEMORY_CACHE) could not be allocated. The following system message indicates why why the allocation could not be made.

Action: If value is to high, lower and retry. If value is reasonable determine what else would be preventing the allocation (process or system limits or usage by other system components).

INVMVXSZ

INVMVXSZ, Invalid block size for GOQ load format

MUPIP Error: This indicates that the LOAD command with the qualifier FORMAT=GOQ determined that the input file did not have the proper block size for that format.

Action: Determine how the file was created and use the proper specification for the FORMAT= qualifier.

INVNETFILNM

INVNETFILNM, Invalid file name following node designation in global directory

GT.CM Server Error: This indicates that the GT.CM Server received a node name that does not exist on the network.

Action: Use GDE to check the Global Directory on the originating node for typographical errors in a remote node file-specification.

INVOBJ

INVOBJ, Cannot ZLINK object file due to unexpected format

Run Time Error: This indicates that ZLINK encountered invalid records in the object file it was trying to integrate into the image.

Action: Determine whether ZLINK has the intended argument. If the object file has been damaged, recreate it with a ZLINK that specifies the source file using an .M extension.

INVPORTSPEC

INVPORTSPEC, Invalid port specification

Run Time Error: This indicates that the OPEN command socket parameter contained an invalid port number.

Action: Redefine the socket parameter to a value between 0 and 65535.

INVPROT

INVPROT, Invalid protocol specified by remote partner

GT.CM Server Error: This indicates that the remote networked system used a protocol incompatible with the locally installed version of GT.CM.

Action: Verify that both systems have compatible versions of GT.CM installed.

INVQUALTIME

INVQUALTIME, Invalid time qualifier value. Specify as xxxx=delta_or_absolute_time.

MUPIP Error: This indicates that time qualifier value specified for xxxx is invalid.

Action: Reissue the command with correct syntax.

INVREDIRQUAL

INVREDIRQUAL, Invalid REDIRECT qualifier value. xxxx

MUPIP Error: This indicates a syntax error in REDIRECT qualifier value.

Action: Reissue the command with correct syntax for REDIRECT.

INVROLLBKLVL

INVROLLBKLVL, Rollback level (xxxx) not less than the current \$tlevel (yyyy). Cannot rollback.

Run Time Error: This indicates that the application is attempting to ROLLBACK to a transaction level that is zero (0) or negative. The minimum transaction level that an application can be rolled back to is one (1).

Action: Review the logic and code path that led to the error and modify the code appropriately.

INVSPECREC

INVSPECREC, Invalid global modifier record

Run Time Error: This indicates that GT.M could not access the header information for a global; therefore, it could not determine the collation characteristics of the global.

Action: Use the %gbldef utility routine to investigate the current state of the global header and correct it.

INVSTACODE

INVSTACODE, Invalid value for second parameter of \$STACK (xxxx).

Run Time Error: This indicates that the intrinsic function \$STACK received an unrecognized string xxxx for the info (second) parameter.

Action: Make sure the second argument is "MCODE", "ECODE" or "PLACE".

INVSTRLEN

INVSTRLEN, Invalid string length xxxx: max yyyy

Run Time Error: This indicates that GT.M encountered a string with a length of xxxx that exceeds the maximum acceptable length yyyy in this context.

Action: Modify the string to an acceptable length.

INVSVN

INVSVN, Invalid special variable name

Compile Time Error: This indicates that a variable of the form "\$name" did not match any valid special variable name.

Action: Look for typographical errors, an improper special variable name abbreviation, or a missing \$ in an extrinsic.

INVTRNSQUAL

INVTRNSQUAL, Invalid TRANSACTION qualifier. Specify only one of TRANSACTION=[NO]SET or TRANSACTION=[NO]KILL.

MUPIP Error: This indicates that an invalid value was assigned to the -TRANSACTION qualifier.

Action: Specify appropriate value to the -TRANSACTION qualifier.

INVZDIRFORM

INVZDIRFORM, Illegal value (xxxx) specified for ZDIR_FORM

Run Time Error: This indicates that the value specified for ZDIR_FORM in the VIEW command is not recognized by GT.M.

Action: Specify a valid value for ZDIR_FORM.

INVZROENT

INVZROENT, xxxx is neither a directory nor an object library(DLL)

Run Time Error: This indicates that an invalid entry (xxxx), neither an object directory, nor a shared library, has been specified in \$ZROUTINES.

Action: Remove xxxx or replace it with a valid directory or a shared library.

INVZSTEP

INVZSTEP, Invalid ZSTEP qualifier

Run Time Error: This indicates that ZSTEP had an argument other than OVER, INTO, or OUTOF.

Action: ZSTEP only accepts these three keyword arguments. It does not accept variables or indirection. Use one of the valid arguments. If you need additional control, consider using ZBREAK.

IOEOF

IOEOF, Attempt to read past an end-of-file

Run Time/MUPIP Error: This indicates that a READ command for a run-time system or a MUPIP command attempted to move past an end-of-file.

Action: Verify that the \$ZEOF special variable is tested by the function before each read or that an EXCEPTION code string is assigned to handle EOFs. The USE command has a REWIND deviceparameter that allows you to read from the beginning of the file without having to CLOSE and OPEN again, which may facilitate recovery from this error. In the event of a MUPIP error, make sure the file being read is not corrupted.

IOERROR

IOERROR, Error occurred while doing aaaa in oooo operation -- called from module mmmm at line LLLL

Runtime Error: On UNIX this indicates a system call used to manage the underlying O/S device for a FIFO, PIPE or Sequential Disk failed in a way that GT.M did not anticipate.

Action: Use the OS documentation to investigate the failure.

IONOTOPEN

IONOTOPEN, Attempt to USE an I/O device which has not been opened

Run Time Error: This indicates that a USE command attempted to make the current device one that had not been OPENed. The current device remains unchanged when this error occurs.

Action: Look for a missing OPEN or an extra CLOSE command.

IORUNDOWN

IORUNDOWN, Error during image rundown

Run Time Error: This indicates that as part of image termination, the process attempted to deallocate all devices and files allocated in GT.M but encountered an error.

Action: Use the appropriate host shell commands to display the statuses of the devices being used by the process, and deallocate any device that is still allocated.

IOWRITERR

IOWRITERR, IO write by PID xxxx to block yyyy of database zzzz failed. PID aaaa retrying the IO.

Run Time Error: This error message is sent to operator log when a queued write fails and is about to be retried. If an error status is available, it follows this message.

Action: Appearance of this message usually indicates disk subsystem error condition. Check disk error logs, in addition to operator logs for accompanying messages.

IPADDRREQ

IPADDRREQ, Active connection requires IP address

Run Time Error: This indicates that an OPEN command specified a TCP mnemonicspace; however, it did not have a socket for an active connection or a LISTEN deviceparameter to designate a passive connection.

Action: Determine the type of connection desired and add a SOCKET=socket-id and/or a LISTEN deviceparameter.

IPCNOTDEL

IPCNOTDEL, xxxx : yyyy did not delete IPC resources for region zzzz

MUPIP Information: This indicates that MUPIP did not delete the shared system resources of the region. The shared system resources may still be being in use by some other processes.

Action: Find out if some other process was attached to the shared system resource. If appropriate, issue MUPIP RUNDOWN REG * to remove the shared resource.

ISOLATIONSTSCHN

ISOLATIONSTSCHN, Error changing NOISOLATION status for global xxxx within a TP transaction from aaaa to bbbb

Run Time Error: In GT.M, the VIEW "NOISOLATION" command changes the isolation-status of the global variable(s) specified. If a process attempts to change the global variable's isolation-status within a TP transaction after it has referenced the global variable in the same TP transaction, the ISOLATIONSTSCHN error gets triggered.

Action: Change the application to issue the VIEW "NOISOLATION" command in conformance with the allowed usage.

IVTIME

IVTIME, Invalid time specification: xxxx

Run Time Error: This indicates that a JOB command specified a SCHEDULE jobparameter with an invalid time xxxx.

Action: Specify the time for the jobparameter according to the OpenVMS time format. GT.M requires a space between the date and time; DCL commands require a colon (:). For more information about the OpenVMS time format, refer to the OpenVMS DCL Dictionary.

JIUNHNDINT

JIUNHNDINT, An error during \$ZINTERRUPT processing was not handled: eeee;

Run-time Error: When returning from code invoked by MUIP INTRPT (clearing \$ZININTERRPT), GT.M implicitly clears any error(s) detected while 1=\$ZININTERRUPT, sends this error notification to the operator log and continues processing; eeee is the mnemonic for the unhandled error.

Action: Fix \$ZINTERRUPT handler to either not generate the error or to correctly handle it before returning to interrupted code

JNLACCESS

JNLACCESS, Error accessing journal file xxxx

Run Time Error: GT.M sends this message to the OpenVMS Operator Communications facility (OPCOM) followed by other messages detailing the failure. xxxx is the file-specification for the inaccessible journal.

Action: Review the accompanying message(s) for additional information. This usually means an error while trying to write to the journal file.

JNLACTINCMPLT

JNLACTINCMPLT, Mupip journal action might be incomplete

MUIP Warning: This indicates that MUIP did not finish to completion successfully.

Action: Review and analyze the accompanying message(s).

JNLALIGNSZCHG

JNLALIGNSZCHG, Journal ALIGNSIZE is rounded up to xxxx blocks (closest next higher power of two)

Run Time Information: This indicates that the ALIGNSIZE specified in the MUIP SET JOURNAL command was not a perfect power of two. It has been rounded up to the closest next higher power of two and the new journal file created (if any) will use this value for ALIGNSIZE.

JNLALIGNTOOSM

JNLALIGNTOOSM, Alignsize xxxx (bytes) is smaller than block size yyyy (bytes) for aaaa bbbb. Using alignsize of cccc (bytes) instead.

MUIP warning: This indicates that the specified alignsize xxxx is smaller than the specified block size yyyy for the mentioned region/database (aaaa) file bbbb. MUIP will use the default cccc bytes instead of the specified xxxx.

Action: If the alignsize cccc used is not acceptable, choose some other legal value for alignsize and reissue the command.

JNLALLOCGROW

JNLALLOCGROW, Increased Journal ALLOCATION from [ssss blocks] to [aaaa blocks] to match AUTOSWITCHLIMIT for ffff nnnn

GDE or MUPIP Information: The utility increased the journal allocation value from ssss to aaaa for the journal files associated with ffff nnnn, which is either "database file" followed by a database file name or "region" followed by a region name. This indicates that the specified journal allocation and journal extension values combined exceed the specified journal autoswitchlimit and the utility has adjusted the journal allocation value accordingly.

Action: None required.

JNLBADLABEL

JNLBADLABEL, Specified File xxxx fdoes not have a GT.M Journal File Label

MUPIP Error: This indicates that the journal file indicated in the accompanying previous message does not match the expected format.

Action: If a command specification caused this error, determine whether the command has the proper file-specification. Make sure the journal file was created by the current GT.M version. Also ensure that a process is not using a journal file-specification for some other purpose.

JNLBADRECFMT

JNLBADRECFMT, Journal Record Format Error encountered for file xxxx at disk address yyyy

MUPIP Error: This indicates that MUPIP has found an error in the journal file.

Action: Report the entire incident context to your GT.M support channel.

JNLBUFINFO

JNLBUFINFO, Pid aaaa dsk bbbb free cccc bytcnt dddd io_in_prog eeee fsync_in_prog ffff dskaddr gggg freeaddr hhhh qiocnt iii
now_writer xxxx fsync_pid yyyy filesize zzzz cycle oooo errcnt pppp wrtsize qqqq fsync_dskaddr rrrr

Run Time Information: This message always accompanies some other GT.M journaling error message. This gives detailed information on the state of the journal buffers at the time of the accompanying error.

Action: For information purposes only. Review the accompanying message(s) for additional information.

JNLBUFFTOOLG

JNLBUFFTOOLG, Journal file buffer xxxx is greater than the maximum allowed size of yyyy. Journal file not created.

MUPIP Warning: This indicates that a MUPIP SET command with the JOURNAL qualifier failed because BUFFER_SIZE=xxxx exceeded the maximum number of pages allowed. yyyy is the maximum buffer size permitted.

Action: Reduce the BUFFER_SIZE= definition.

JNLBUFFTOOSM

JNLBUFFTOOSM, Journal file buffer xxxx is less than minimum of database block size in 512 byte pages + 1 (yyyy)

MUPIP Warning: This indicates that a SET command with the JOURNAL qualifier failed because the BUFFER_SIZE=xxxx was inadequate to handle a database block. yyyy is the minimum buffer size permitted for the existing block size.

Action: Increase the BUFFER_SIZE so it contains at least one more page than the database block, that is, bytes divided by 512.

JNLCLOSE

JNLCLOSE, Error closing journal file: xxxx

Run Time Error: This indicates that GT.M could not properly close journal file xxxx.

Action: Review the accompanying message(s) for additional information.

JNLCLOSED

JNLCLOSED, Journaling closed for database file dddd at transaction number xxx

GT.M run-time warning: This message indicates GT.M had to turn journaling OFF on the specified database. Other preceding messages identify the cause (e.g. lack of disk space while writing to journal file, permissions issue while auto-switching to new journal files etc.). The message also displays the database transaction number.

Action:

1. Fix the issue that caused journaling to get turned OFF in the first place (disk space, permissions etc.).
 2. Turn journaling back ON by issuing a MUPIP SET JOURNAL=ON or MUPIP BACKUP NEWJNL command. This command can work while processes are concurrently updating the database and causes GTM to journal subsequent updates in the journal file.
-

JNLCNTRL

JNLCNTRL, Journal control unsynchronized. Journaling closed for xxxx.

Run Time Error: This indicates that there is a discrepancy between the journal file updates and the database updates. The system is not updating journal files.

Action: Review the accompanying message(s) and take appropriate action. After the cause is resolved, to reestablish durability, perform a MUPIP BACKUP that turns journaling back on. Once the system is backup and running, contact your GT.M support channel with operator log information and any additional information that you feel is relevant for further diagnosis.

JNLCREATE

JNLCREATE, Journal file xxxx created for <database/region> yyyy with aaaa

MUPIP Information: This indicates that a journal file xxxx is created for database/region yyyy with NOBEFORE_IMAGES or BEFORE_IMAGES journaling option (aaaa).

JNLCREATERR ●

Last used version: **V5.4-001**

JNLCREATERR, Error creating journal file xxxx

Run Time Error: This indicates that there was an error while trying to create journal file xxxx.

Action: Review the accompanying message(s) for additional information.

JNLCRESTATUS

JNLCRESTATUS, xxxx at line aaaa for journal file yyyy, database file zzzz encountered error

Run Time Warning / MUPIP Warning: This indicates the creation of journal file yyyy for database file zzzz failed.

Action: Review the accompanying messages and take appropriate action.

JNLCYCLE

JNLCYCLE, Journal file xxxx causes cycle in the journal file generations of database file yyyy

MUPIP Error: This indicates that MUPIP encountered journal file xxxx causing cycle in the journal file generations of database file yyyy. (this is not explanatory, at least to me.)

Action: Contact your GT.M support channel with appropriate log messages.

JNLDBERR

JNLDBERR, Journal file xxxx does not correspond to database yyyy

Run Time Error: This indicates that GT.M could not open journal file xxxx for database file yyyy because the journal file header identifies itself as belonging to a different database file that does not exist in the system.

Action: Use a MUPIP SET command with the qualifier JOURNAL to create a journal file that matches the database.

JNLDBTNNOMATCH

JNLDBTNNOMATCH, Journal file xxxx has beginning transaction number aaaa but database yyyy has current transaction number bbbb

MUPIP Error: MUPIP JOURNAL FORWARD has found that journal file xxxx has beginning transaction number aaaa, but the corresponding database file yyyy has current transaction number bbbb. This condition may arise due to missing or duplicate transactions.

Action: Verify that the correct journal file names were specified. If appropriate, force forward recovery using the NOTNCHECK qualifier.

JNLDISABLE

JNLDISABLE, Specified journal option(s) cannot take effect as journaling is DISABLED on database file xxxx

MUPIP Warning: This indicates that none of the specified journal option(s) in MUPIP SET -JOURNAL or MUPIP BACKUP command took effect, because journaling was found DISABLED on database file xxxx.

JNLDSKALIGN

JNLDSKALIGN, Journal Record Alignment xxxx is not a multiple of 512

MUPIP Warning: This indicates that the system is not updating the journal file correctly.

Action: Close the journal file and begin journaling with a new journal file. If the error continues, check the disk subsystem.

JNLENDIANBIG

JNLENDIANBIG, Journal file jjjj is BIG endian on a LITTLE endian system

MUPIP Error: The MUPIP command on a little endian system specified journal file jjjj which was created on a big endian system. GT.M does not convert journal files with incompatible byte ordering.

Action: Set up operational procedures that ensure journal files are used on systems with the same byte ordering as where they are created. If necessary, extract journal file data on the source system and use an M program on the opposite endian system to restore it.

JNLENDIANLITTLE

JNLENDIANLITTLE, Journal file jjjj is LITTLE endian on a BIG endian system

MUPIP Error: The MUPIP command on a big endian system specified journal file jjjj which was created on a little endian system. GT.M does not convert journal files with incompatible byte ordering.

Action: Set up operational procedures that ensure journal files are used on systems with the same byte ordering as where they are created. If necessary, extract journal file data on the source system and use an M program on the opposite endian system to restore it.

JNLEOFPREZERO

JNLEOFPREZERO, Error while zeroing journal file xxxx

Run Time Error: Write operation fails. When the last process with write access to a journaled database leaves, the system closes the journal file by writing an end-of-file (EOF) record at a 512-byte boundary. To do this, the system zeroes the journal file from the current offset until the next aligned 512-byte boundary. It is this write operation whose failure causes this error.

Action: Check the integrity of the journal file by doing a MUPIP JOURNAL -FORWARD -VERIFY jnl-file.

JNLEXTEND

JNLEXTEND, Journal file extension error. Journal file xxxx closed.

Run Time Error: Journal file xxxx failed to extend. This causes journaling to be turned off for the region.

Action: Review the accompanying message(s) and take appropriate action. After the cause is resolved to reestablish durability, perform a MUPIP BACKUP that turns the journaling on again.

JNLEXTR

JNLEXTR, Error writing journal extract file: xxxx

MUPIP Error: This indicates that an error was encountered while trying to write to either the JNL EXTRACT file or lost-transaction file or broken-transaction file as part of a MUPIP JOURNAL command.

Action: Review the accompanying message(s) for additional information.

JNLFILECLOSERR

JNLFILECLOSERR, Error closing journal file xxxx

MUPIP Error: This indicates that MUPIP JOURNAL command failed to close the specified journal file xxxx.

Action: Review the accompanying message(s) for additional information.

JNLFILEDUP

JNLFILEDUP, Journal files xxxx and yyyy are the same

MUPIP Information: MUPIP JOURNAL -RECOVER -FORWARD does not allow duplicated journal files in forward recovery.

Action: Remove any duplicated journal file(s) and re-issue the forward recovery command.

JNLFILEOPNERR

JNLFILEOPNERR, Error opening journal file xxxx

MUPIP Error: This indicates that MUPIP JOURNAL command failed to open the specified journal file xxxx.

Action: Ensure the journal file name specified is correct. Review the accompanying message(s) for additional information.

JNLFILEXTERR

JNLFILEXTERR, Error during extension of journal file xxxx

Run Time Error: This indicates that an error was encountered during the course of journal file extension, while trying to determine the available space on the file system housing the journal file xxxx. This causes the journaling to be turned off.

Action: Locate appropriate disk space and adjust the journal file path. To reestablish durability, perform a MUPIP BACKUP that turns back journaling on again.

JNLFILNOTCHG

JNLFILNOTCHG, Journal file not changed

MUPIP Error: This indicates that the MUPIP SET-JNLFILE command was unable to change the journal file as specified.

Action: Review accompanying message(s) for additional information.

JNLFILOPN

JNLFILOPN, Error opening journal file xxxx for database file yyyy

Run Time Error: This indicates that GT.M was unable to open journal file xxxx for the specified database file.

Action: Review the accompanying message(s) for additional information.

JNLFLUSH

JNLFLUSH, Error flushing journal buffers to journal file xxxx

Run Time Error: This indicates that an attempt to write existing journal records to the journal file failed.

Action: Review the accompanying message(s) for additional information.

JNLFLUSHNOPROG

JNLFLUSHNOPROG, No progress while attempting to flush journal file xxxx

Run Time Warning: This warning message is logged to the operator facility. This indicates that a GT.M process waited for about 2 minutes for journal file xxxx to be flushed to disk from database shared memory. The accompanying TEXT message gives more details. The JNLFLUSHNOPROG message is preceded by JNLFLUSH messages.

Action: Indicative of a clogged disk subsystem on which journal file xxxx resides. Consider balancing disk subsystem load.

JNLFNF

JNLFNF, Journal file xxxx not found

MUPIP Information: This indicates that MUPIP did not find the specified journal file xxxx while executing the command.

JNLFSYNCERR

JNLFSYNCERR, Error synchronizing journal file xxxx to disk

Run Time Error: This indicates that the fsync() function on the journal file xxxx failed. This is likely a disk subsystem related problem.

Action: Review the accompanying messages for cause of the failure.

JNLFSYNCLSTCK

Severity: Error

JNLFSYNCLSTCK, Journaling fsync lock is stuck in journal file jjjj

Run Time Error: This indicates that the system has been unable to write to the journal file for a long duration and information useful for debugging will be logged in the syslog. It is printed after one minute and two minutes, if sync has not occurred yet, after which there will be an FSYNCTIMEOUT error.

Action: No action required, unless accompanied by FSYNCTIMEOUT. If this message comes up frequently without FSYNCTIMEOUT, you might still want to check the disk subsystem for hardware or software problems.

JNLINVALID

JNLINVALID, xxxx is not a valid journal file Region: yyyy

Run Time Error: This indicates that GT.M could not open journal file xxxx, due to an error that is detailed in the accompanying previous message(s). While trying to create a new journal file for the same region it encountered errors. yyyy is the region name associated with the journal.

Action: Review the accompanying error message(s) to determine the cause of the failure of the new journal file creation. After the cause is resolved, to reestablish durability perform a MUPIP BACKUP that turns journaling back on.

JNLINVALLOC

JNLINVALLOC, Journal file allocation xxxx is not within the valid range of yyyy to zzzz. Journal file not created.

MUPIP Warning: This indicates that a SET command modified with the JOURNAL qualifier failed because ALLOCATION=xxxx was less than the minimum or greater than the maximum number of blocks. yyyy is the minimum allocation permitted. zzzz is the maximum allocation permitted.

Action: Adjust the ALLOCATION= to adhere to the valid range.

JNLINVEXT

JNLINVEXT, Journal file extension xxxx is greater than the maximum allowed size of yyyy. Journal file not created.

MUPIP Warning: This indicates that a SET command modified with the JOURNAL qualifier failed because the EXTENSION=xxxx argument exceeded the maximum number of pages. yyyy is the maximum extension permitted.

Action: Reduce the EXTENSION= qualifier definition.

JNLINVSWITCHLMT

JNLINVSWITCHLMT, Specified AUTOSWITCHLIMIT xxxx falls outside of allowed limits aaaa and bbbb

MUPIP Error: This indicates that the specified autoswitchlimit for the journal file is outside of the allowed range [that is indicated in the error message].

Action: Specify an autoswitchlimit within the specified allowed range.

JNLMEMDSK ●

Last used version: **V5.3-001A**

JNLMEMDSK, Journal file unsynchronized with the journal buffer. Journaling closed for xxxx.

Run Time Error: This indicates that the structures in the journal buffer reflect an out-of-design situation. As a result, GT.M has shut down journaling for the xxxx region.

Action: Review the accompanying message(s) and take appropriate action. After the cause is resolved, to reestablish durability, perform a MUPIP BACKUP that turns journaling back on. Once the system is backup and running, contact your GT.M support channel with operator log information and any additional information that you feel is relevant for further diagnosis.

JNLMINALIGN

JNLMINALIGN, Journal Record Alignment xxxx is less than the minimum value of yyyy

MUPIP Warning: This indicates that a MUPIP SET JOURNAL command specified an alignsize for the new journal file, which is less than the minimum allowed yyyy.

Action: Specify an alignsize that is greater than the allowed minimum.

JNLMOVED

JNLMOVED, Journal file appears to have been moved. Journaling activity will not be done.

Run Time Error: This indicates that while opening a journal file the system encountered the journal file name in the database header file, pointing to a different location than the journal file ID.

Action: Use MUPIP SET to specify the correct journal file location.

JNLNAMLEN

JNLNAMLEN, Journal file xxxx: for database file yyyy exceeds maximum of zzzz

MUPIP Error: This indicates that the file-specification xxxx of the journal for database file yyyy exceeds the maximum supported length of zzzz.

Action: Modify the journal file-specification to adhere to the file length restrictions.

JNLNEWREC

JNLNEWREC, Target system cannot recognize journal record of type xxxx. Last recognized type is yyyy.

MUPIP Error: This error is logged to the replication server log file. This indicates that the GT.M application on the replication primary generated a newly introduced journal record (of type xxxx) that is not recognized by the secondary system. The highest numbered journal record type on the secondary is yyyy. A new type that can be transformed to an older type internally by the source server on the primary side will be automatically done. This error occurs when the primary source server cannot transform the new type to an older type due to impact on the application logic.

Action: The application either should not generate a journal record that is not recognized by the secondary, or, write a filter that transforms the unknown type to a known type on the target system. The replication source server on the primary should be restarted with the filter.

JNLNOCREATE

JNLNOCREATE, Journal file xxxx not created

MUPIP Error: This indicates that MUPIP could not create journal file xxxx.

Action: Review the accompanying message(s) for additional information.

JNLNOREPL

JNLNOREPL, Replication not enabled for journal file jjjj (database file dddd)

MUPIP Error: Replication Source Server startup encountered a database dddd with journal file jjjj for which replication was turned off because of a journaling issue and has not since been re-enabled.

Action: Use MUPIP SET to re-enable replication. Take steps to ensure that there is sufficient management of journal file space to prevent a reoccurrence of this issue.

JNLNMBKNOTPRCD

JNLNMBKNOTPRCD, Journal file xxxx does not match the current journal file yyyy of database file zzzz

MUPIP Error: This indicates that MUPIP JOURNAL BACKWARD cannot proceed because the journal file name xxxx is not same as the journal file name yyyy in the database file header of zzzz.

Action: Specify the correct journal file name for the database zzzz. If the database file header is not pointing to the correct journal file, fix it using MUPIP SET JOURNAL.

JNLNOBIJBACK

JNLNOBIJBACK, MUPIP JOURNAL BACKWARD cannot continue as journal file xxxx does not have before image journaling

MUPIP Error: This indicates that an attempt to use BACKWARD qualifier on xxxx journal file was made without enabling before image journaling on the file.

Action: Ensure before image journaling is enabled prior to the usage of BACKWARD qualifier, alternatively use FORWARD qualifier for nobefore image journaling enabled files.

JNLNOTFOUND

JNLNOTFOUND, File xxxx does not exist -- possibly moved or deleted

MUPIP Information: This indicates that MUPIP has encountered a journal file link name xxxx, which does not exist in the system anymore.

Action: This message is usually accompanied by a NORECOVERR error message. Either a closer turn-around-point mapping to an existing prior journal file generation needs to be specified, or the missing journal file needs to be restored from the backup.

JNLOPNERR

JNLOPNERR, Error opening journal file xxxx for region yyyy

Run Time/MUPIP Error: This indicates that GT.M could not open the journal file xxxx.

Action: Review and trouble shoot accompanying messages.

JNLPOOLBADSLLOT

JNLPOOLBADSLLOT, Source server slot for secondary instance xxxx is in an inconsistent state. Pid = pppp, State = ssss, SlotIndex = iiii

MUPIP Warning: This is a debugging message sent to the syslog (operator log) whenever a source server startup or showbacklog command finds a structure in the journal pool holding inconsistent information.

Action: Forward the information to your GT.M support channel. No action otherwise necessary. The source server command will automatically fix the inconsistency of that structure

JNLPOOLSETUP

JNLPOOLSETUP, Journal Pool setup error

Run Time/MUPIP Error: This indicates that an error occurred in the replication subsystem while opening the journal pool.

Action: Verify that the source server has been configured correctly. Review accompanying messages for more information about the cause of this error.

JNLPREVRECOV

JNLPREVRECOV, Journal file has nonzero value in prev_recov_end_of_data field

Run Time Error: This indicates that GT.M encountered a non-zero value for the journal file header prev_recov_end_of_data field. MUPIP JOURNAL RECOVER/ROLLBACK can cause the field to be non-zero but it cannot become a current generation journal file for GT.M run-time. Run-time considers the journal file as bad and switches to a new journal file cutting the back-link.

Action: Report the complete error to GT.M support along with appropriate log messages within the same time frame.

JNLPROCSTUCK

JNLPROCSTUCK, Journal file writes blocked by process xxxx

Run Time Error: This indicates that a GT.M process waited for nearly one minute and is not able to flush the journal as the journal write mechanism seems to be blocked by process xxxx.

Action: If the situation does not improve, kill the offending process xxxx. This may indicate an overloaded disk subsystem on which journal file xxxx resides. Consider balancing disk subsystem load. If necessary, report the entire incident context with operator log information to your GT.M support channel.

JNLQIOLOCKED

JNLQIOLOCKED, Error obtaining io_in_prog lock on journal file xxxx

Run Time Error: When the last process with write access to a journaled database exits, it does a cleanup of the journal file by attempting to get an exclusive write lock on the journal file. If that fails after waiting for nearly one minute, it issues this error.

Action: Attempt MUPIP RUNDOWN on the region making sure to run down all processes actively attached to this database. If the error persists even after no one is attached, contact your GT.M support channel before proceeding.

JNLRDERR

JNLRDERR, Error reading journal file xxxx: unable to initialize.

Run Time Error: This indicates that GT.M encountered an error while trying to read from the journal file xxxx. This can happen if the journal file size is less than the minimum size of the journal file header, or if the journal-file-specification does not match the intended file, or if the intended file is not accessible by the processes that update the database. GT.M automatically attempts to

create a new journal file. If the attempt to create a new journal file fails, GT.M issues another error and the intended update is lost and does not get registered in the database and journal.

Action: To reestablish durability, perform a MUPIP BACKUP that switches to a new set of journal files.

JNLRDONLY

JNLRDONLY, Journal file xxxx read only

MUPIP Error: This indicates that the process in use does not have journal file write privileges or the journal file has been set to read-only.

Action: You may be performing an invalid operation. Contact your Systems Administrator.

JNLREAD

JNLREAD, Error reading from journal file xxxx at offset yyyy

MUPIP Error: This indicates that MUPIP failed to read from journal file xxxx at offset yyyy.

Action: Review the accompanying message(s) for additional information.

JNLREADBOF

JNLREADBOF, Beginning of journal file encountered for xxxx

MUPIP Error: This indicates that MUPIP JOURNAL command reached the beginning of journal file xxxx, while processing backward, and is not able to process backward anymore. This maybe due to the time qualifiers used to control the length of the backward processing.

Action: Verify that the time qualifiers specified are as intended.

JNLREADEOF

JNLREADEOF, End of journal file encountered for xxxx

MUPIP Error: This indicates that MUPIP JOURNAL encountered the end-of-file for the journal file xxxx, before it completed processing.

Action: This error indicates an improperly closed journal file.

JNLRECFMT

JNLRECFMT, Journal file record format error encountered

Run Time/MUPIP Error: This indicates that MUPIP JOURNAL encountered an invalid record in the journal file.

Action: In the event of GT.M issuing this error message, use MUPIP BACKUP to ensure durability by creating a fresh set of journals consistent with the database. Else, to resume operation, restore the database from the last backup and play forward the updates using the appropriate MUPIP JOURNAL command. As soon as possible, report the entire incident context with information from the operator log and any other relevant information to your GT.M support channel.

JNLRECINCMPL

JNLRECINCMPL, Incomplete journal record at disk address aaaa for file jjjj while attempting to read seqno ssss

MUPIP Error: The replication Source Server had a problem with journal file jjjj at disk offset aaaa attempting to read the record with sequence number ssss.

Action: Report the entire incident context to your GT.M support channel for further analysis. Use MUPIP SET JOURNAL - EXTRACT to investigate the issue.

JNLRECTYPE

JNLRECTYPE, Journal record type does not match expected type

Run Time Error: This indicates that when GT.M tried to open the journal file as part of an M update, the end of the journal file as indicated by the journal-file-header did not contain an EOF journal record implying that the journal file is either damaged or corrupted. This message follows a JNLOPNERR message, which indicates the journal file name and the corresponding region. GT.M automatically attempts to create a new journal file and errors out if the attempt does not succeed, in which case the intended update is not reflected in the database and journal.

Action: To reestablish durability, perform a MUPIP BACKUP that switches to a new set of journal files consistent with the database.

JNLSENDOPER

JNLSENDOPER, pid = aaaa : status = bbbb : jpc_status = cccc : jpc_status2 = dddd : iosb.cond = eeee

Run Time Information: This message gives information on the process that encountered an error in GT.M journaling and the error code encountered. This message is always followed by a GT.M journaling error message that gives the error detail.

Action: Review the accompanying message(s) for additional information.

JNLSETDATA2LONG

JNLSETDATA2LONG, SET journal record has data of length xxxx. Target system cannot handle more than yyyy bytes.

MUPIP Error: This error message is logged to the replication server log file. The version of GT.M running on the replication primary system supports longer data lengths for globals than the GT.M version running on the secondary system.

Action: Until the secondary is upgraded to the newer version of GT.M, the application should not use the new feature of longer data lengths.

JNLSPACELOW

JNLSPACELOW, Journal file xxxx nearing maximum size, aaaa blocks to go

Run Time Information: This indicates that the journal file xxxx is approaching the maximum size specified for it. The system creates a new journal file when the limit is reached. Typically none, unless operational practice uses this as a trigger to intervene in journal file management.

JNLSTATE

JNLSTATE, Journaling state for <database/region> xxxx is now yyyy

MUPIP Information: This indicates that journal state for the database/region xxxx is now yyyy.

JNLSTATEOFF

JNLSTATEOFF, ROLLBACK or RECOVER BACKWARD cannot proceed as database file xxxx does not have journaling ENABLED and ON

MUPIP Error: This indicates that ROLLBACK or RECOVER cannot proceed because MUPIP encountered a database file xxxx, which does not have journaling ENABLED and ON.

Action: Verify that the file(s) specified is correct. Ensure that Journaling is ENABLED and ON for RECOVER BACKWARD to work.

JNLSTRESTFL

JNLSTRESTFL, Failed to restore journaling state for database file xxxx

MUPIP Error: This indicates that MUPIP RECOVER failed to restore journaling state prior to the MUPIP RECOVER command.

Action: Perform MUPIP INTEG on the database. If INTEG cannot be done for DBFLCORRP error, reset the field with appropriate command. If database is not in a good state, restore old database and journals and reissue the recover command. If this is the only error encountered during recovery, explicit journal switch using MUPIP BACKUP for a consistent copy of the database should be used and normal operations restored.

JNLSUCCESS

JNLSUCCESS, xxxx successful

MUPIP Success: This indicates that xxxx command has finished successfully.

JNLSWITCHSZCHG

JNLSWITCHSZCHG, Journal AUTOSWITCHLIMIT [aaaa blocks] is rounded down to [bbbb blocks] to equal the sum of journal ALLOCATION

Run Time Information: This indicates that the specified AUTOSWITCHLIMIT value was rounded down as little as possible to make it aligned to the ALLOCATION + a multiple of EXTENSION. The new journal file created (if any) will use this value for AUTOSWITCHLIMIT.

JNLSWITCHTOOSM

JNLSWITCHTOOSM, Journal AUTOSWITCHLIMIT [aaaa blocks] is less than journal ALLOCATION [bbbb blocks] for database file xxxx

Run Time Error: This indicates that the value of AUTOSWITCHLIMIT specified in a MUPIP SET JOURNAL command is less than the default or specified value of ALLOCATION. This error also indicates that the AUTOSWITCHLIMIT value specified was greater or equal to the ALLOCATION but in turn got rounded down, and this rounded down value is lesser than the ALLOCATION.

Action: Specify a higher value of AUTOSWITCHLIMIT.

JNLTMQUAL1

JNLTMQUAL1, Time qualifier BEFORE_TIME=xxxx is less than SINCE_TIME=yyyy

MUPIP Error: This indicates that the specified before time xxxx is earlier than the since time yyyy.

Action: Specify correct values for time qualifiers and make sure that BEFORE_TIME is specified to be later than the SINCE_TIME.

JNLTMQUAL2

JNLTMQUAL2, Time qualifier LOOKBACK_TIME=xxxx is later than SINCE_TIME=yyyy

MUPIP Error: This indicates that the specified lookback time xxxx is later than the since time yyyy.

Action: Specify correct values for time qualifiers and make sure that the LOOKBACK_TIME qualifier is set to an earlier time value than the SINCE_TIME qualifier.

JNLTMQUAL3

JNLTMQUAL3, Time qualifier BEFORE_TIME=xxxx is less than the journal file(s) minimum timestamp=yyyy

MUPIP Error: This indicates that the specified before time xxxx is earlier than the earliest timestamp yyyy, found in the journal file(s).

Action: Specify correct value(s) for time qualifiers.

JNLTMQUAL4

JNLTMQUAL4, Time qualifier BEFORE_TIME="xxxx is less than AFTER_TIME="yyyy"

MUPIP Error: This indicates that the specified BEFORE_TIME xxxx is earlier than the AFTER_TIME yyyy specified

Action: Specify correct values for the time qualifiers and make sure that AFTER_TIME is earlier than BEFORE_TIME qualifier.

JNLTNOUTOFSEQ

JNLTNOUTOFSEQ, End transaction aaaa of journal xxxx different from Begin transaction bbbb of next generation journal yyyy

MUPIP Error: MUPIP JOURNAL FORWARD command has found that the transaction numbers (aaaa and bbbb) of two consecutive generation journal files (xxxx and yyyy) are not in sequence. It is expected that the end transaction of a journal file is the same as the begin transaction of the immediately succeeding generation.

Action: Ensure the specification of journal file names is as intended. Verify if the journal file xxxx and yyyy are really in sequence. Find out if any of the transactions are missing or duplicate using MUPIP JOURNAL SHOW=HEAD FORWARD NOVERIFY. If appropriate, force forward recovery using the NOTNCHECK qualifier.

JNLTPNEST

JNLTPNEST, Mupip journal command found nested TP transactions for journal file jjjj at offset oooo at transaction number nnnn

MUPIP Warning: MUPIP JOURNAL -RECOVER or ROLLBACK encountered a TSTART record for transaction nnnn at offset oooo in journal file jjjj while already processing an uncommitted transaction. Since the run-time system should never produce this situation, the journal file is suspect. MUPIP discards the in-progress transaction and proceeds.

Action: Extract the journal file(s) and use the context from the message to find the transactions in question and adjust for any lost or tangled transaction(s).

JNLTRANS2BIG

JNLTRANS2BIG, Transaction needs an estimated [aaaa blocks] in journal file xxxx which exceeds the AUTOSWITCHLIMIT of bbbb

Run Time Error: This indicates that a database update transaction needs aaaa blocks of space in the journal file for its corresponding journal records and this exceeds the AUTOSWITCHLIMIT value of the current journal file.

Action: Increase the AUTOSWITCHLIMIT, or if the transaction is a TP transaction decrease the number of updates done within one transaction thereby decreasing its journal file space requirement.

JNLTRANSGTR

JNLTRANSGTR, Transaction number in journal is greater than in database

Run Time Warning: This indicates that GT.M was unable to open the journal file because its transaction number does not match the database files transaction number

Action: GT.M automatically closes the current journal file and creates a new one. To reestablish durability, perform MUPIP BACKUP to create a fresh set of journals consistent with the database. Review the accompanying message(s) for information on the journal file name.

JNLTRANSLSS

JNLTRANSLSS, Transaction number in journal is less than in database

Run Time Warning: This indicates that GT.M was unable to open the journal file because its transaction number does not match the database files transaction number.

Action: GT.M automatically closes the current journal file and creates a new one. To reestablish durability, perform MUPIP BACKUP to create a fresh set of journals consistent with the database. Review the accompanying message(s) for information on the journal file name.

JNLUNXPCTERR

JNLUNXPCTERR, Unexpected error encountered for Journal aaaa at disk address 0xbbbb

MUPIP Error: This indicates that MUPIP JOURNAL has detected an unexpected error in the journal file that prevents the command from proceeding. A recovery or rollback that uses this journal file cannot successfully complete.

Action: Report the entire incident context to your GT.M support channel.

JNLVSIZE

JNLVSIZE, Journal File xxxx has incorrect virtual_filesize aaaa Allocation is bbbb extension is cccc filesize is dddd file_system_block_size is eeee

Run Time Error: This indicates that journal file xxxx has incorrect value in the Virtual file size file header field. Either it is less than the actual filesize or it is not the same as allocation + n * extension.

Action: Run time system creates and switches to a new journal file and continues to run. Report to your GT.M support channel with any accompanying message(s).

JNLPVTINFO

JNLPVTINFO, Pid aaaa cycle mmmm fd_mismatch nnnn channel rrrr sync_io ssss pini_addr xxxx qio_active yyyy old_channel zzzz

Run Time Information: This message always accompanies some other GT.M journaling error message. This gives detailed information on the state of the journal buffers at the time of the accompanying error.

Action: For information purposes only. Review the accompanying message(s) for additional information.

JNLWRERR

JNLWRERR, Error writing journal file xxxx. Unable to update header Region: yyyy

Run Time Error: This indicates that GT.M encountered an error while updating the journal file header as part of trying to open the journal file.

Action: Review the accompanying message(s) for detail on the cause of the error. GT.M automatically closes the current journal file and creates a new one. To reestablish durability, perform MUPIP BACKUP to create a fresh set of journals consistent with the database.

JNLWRTDEFER

JNLWRTDEFER, Journal write start deferred

Run Time Information: This message always accompanies some other GT.M journaling message. This indicates that a flush of the journal buffer to the disk system was deferred since some other process is currently busy flushing the journal buffer.

Action: Review the accompanying message(s) for additional information.

JOBACTREF

JOBACTREF, Actual parameter in job command passed by reference

Compile Time Error: This indicates that arguments to JOB cannot be passed by reference.

Action: Arguments to JOB must be passed by value.

JOBARGMISSING

JOBARGMISSING, Missing job argument nnnn - can't skip non-trailing arguments to a JOB command in OpenVMS editions

Run time error: Indicates a JOB command parameter list contains two adjacent commas (,,), which should show an intention to not supply an actual value for the corresponding formallist parameter. nnnn is the ordinal number of the first argument with this issue.

Action: If the parameter was not intentionally skipped, correct the JOB command. If the parameter should be optional, reorder the lists so the parameter in question is at the end of the list where it can be omitted or establish a distinguished value, such as an empty string, to serve in place of an omitted value.

JOBEXAMDONE

JOBEXAMDONE, GT.M process aaaa completed job examine to xxxx

Run Time Information: This informational message reports that a \$ZJOBEXAM was performed and gives a complete file specification. The message is sent to operators log.

JOBEXAMFAIL

JOBEXAMFAIL, GT.M process aaaa executing \$ZJOBEXAM function failed with the preceding error message

Run Time Error: This is a secondary message that accompanies a \$ZJOBEXAM function error. This error message is sent to the operator log.

Action: Review the accompanying message(s) and take appropriate action.

JOBFAIL

JOBFAIL, JOB command failure

Run Time Error: This indicates that a JOB command did not complete successfully.

Action: Review the accompanying message(s) for additional information. If a STARTUP jobparameter is specified, make sure that the file is accessible and has the desired content.

On OpenVMS systems: Verify the logical names and the LOGIN.COM file of the process that is issuing the JOB command because differences in the environment are the most common reason for failure.

JOBLABOFF

JOBLABOFF, Label and offset not found in created process

Run Time Error: This indicates that a JOB command specified an entry reference that could not be located in the image used by the new job.

Action: Verify that the image being JOBbed is properly linked. If the image has been changed since it was last LINKed, determine whether the new job has access to the files necessary to ZLINK the changes. You can also LINK the image to include the changes.

JOBPARM ●

Last used version: **V5.4-000A**

JOBPARM, Error accessing job parameters

Run Time Error: GT.M could not retrieve the job information piped from the originating job.

Action: Check for appropriate permissions for the parent and child job environments, also for operational actions that might delete temporary pipes; if necessary, report the entire incident context to your GT.M support channel for additional diagnostic assistance.

JOBPARNOVAL

JOBPARNOVAL, This job parameter cannot take a value

Compile Time Error: This indicates that a JOB command specified a value for a jobparameter that does not accept a value.

Action: Modify the jobparameter or remove its argument.

JOBPARNUM

JOBPARNUM, The value of this job parameter must be an integer

Compile Time Error: This indicates that a JOB command specified a valid jobparameter but it did not assign an integer value, which is required.

Action: Verify that the jobparameter has an integer literal argument and not a string or variable argument.

JOBPARSTR

JOBPARSTR, The value of this job parameter must be a string

Compile Time Error: This indicates that a JOB command specified a valid jobparameter but did not assign the jobparameter a string value as expected.

Action: Ensure that the jobparameter has a string literal argument and not a variable or keyword argument.

JOBPARTOOLONG

JOBPARTOOLONG Total parameter length is too long for job command

Run Time Error: This indicates that the total length of jobparameters that must be passed to the created job exceeded the size of the buffer that was available to handle them.

Action: Use fewer and/or shorter items in parameter passing; consider passing information in a global.

JOBPARUNK

JOBPARUNK, Job parameter unknown

Compile Time Error: This indicates that a JOB command specified an invalid jobparameter keyword.

Action: Specify a valid jobparameter keyword. Refer to the Programmer's Guide for a valid keyword.

JOBPARVALREQ

JOBPARVALREQ, A value is required for this job parameter

Compile Time Error: This indicates that a JOB command specified a valid jobparameter but did not assign the jobparameter a value, which is required.

Action: Review the jobparameters for proper assignments.

JRTNULLFAIL

JRTNULLFAIL, Applying NULL journal record failed. Failure code: xxxx.

MUPIP Error: Issued by an Update Process, MUPIP JOURNAL -ROLLBACK or MUPIP JOURNAL -RECOVER indicating it encountered a database problem when it attempted to play a NULL journal record into the database. xxxx contains the failure codes for the four attempts. It is very likely that the database may have integrity errors or that the process-private data structures are corrupted.

Action: Report the entire incident context to your GT.M support channel for further analysis.

JUSTFRACT

JUSTFRACT, Fraction specifier to \$JUSTIFY cannot be negative

Run Time Error: This indicates that a \$JUSTIFY or \$FNUMBER function specified a negative value as its third argument.

Action: Modify the third argument of the function so that it specifies the positive number of fractional digits to which the function rounds its result.

KEY2BIG

KEY2BIG, Key size (xxxx) is greater than maximum (yyyy) for region zzzz

Run Time Error: This indicates that a SET command attempted to establish a global variable with a total subscript length xxxx, which exceeds the maximum length yyyy specified in the file header for the current region zzzz.

Action: Verify that the subscripts are as intended. Use the DSE DUMP command and qualifier FILEHEADER to examine the key size for the region. Modify KEYSIZE if required by the application.

KEYSIZIS

KEYSIZIS, Key size is xxxx

GDE/DSE Information: This message displays the maximum key size xxxx of the REGION with which you are working.

KEYTOOBIG

KEYTOOBIG, But record size xxxx can only support key size yyyy

GDE Warning: This indicates that an ADD, CHANGE, or TEMPLATE command specified a value for the KEYSIZE qualifier that is incompatible with the value xxxx assigned to RECORDSIZE. yyyy is the maximum value of KEYSIZE that this RECORDSIZE value can support.

Action: Review the accompanying message for the key size. Modify the key size and/or record size so that they are compatible.

KEYWRDAMB

KEYWRDAMB, xxxx is ambiguous for yyyy

GDE Error: This indicates that the keyword xxxx is ambiguous for the command or local qualifier yyyy.

Action: Ensure that the command or qualifier has enough characters to differentiate it from similar command elements.

KEYWRDBAD

KEYWRDBAD, xxxx is not a valid yyyy

GDE Error: This indicates that GDE did not encounter the valid syntax element, which was a command or qualifier. xxxx is the invalid element. yyyy designates either command or qualifier.

Action: Look for and correct typographical errors.

KILLABANDONED

KILLABANDONED, Abandoned kills counter is greater than zero for file ffff, tttt

Run Time Error: This indicates a process terminated during KILL cleanup in database file ffff; tttt is text warning of the implications. Generally, this leaves a database with block incorrectly marked busy errors. Such errors are benign in that they only cause blocks to be inappropriately unavailable. Nonetheless they should be addressed promptly to avoid operators becoming desensitized to errors in INTEGs.

Action: Use DSE MAP to carefully FREE individual incorrectly marked busy block. If there are many blocks, you can edit the output of the integ (run with NOMAP) to create a script for driving repeated DSE MAP FREE. Alternatively, if you can get standalone access, to the database you may use DSE MAP RESTORE - never use MAP RESTORE on an active database.

KILLBYSIG

KILLBYSIG, Process xxxx has been killed by a signal yyyy

Run Time Error: This indicates that the xxxx process failed due to signal yyyy.

Action: Preserve the core (dump) files and report the entire incident context to your GT.M support channel for further analysis.

KILLBYSIGSINFO1

KILLBYSIGSINFO1, iiii process xxxx has been killed by a signal yyyy at address aaaa (vaddr bbbb)

Run Time Error: This indicates that the process failed due to the yyyy signal, which occurred at the code address aaaa. bbbb is the virtual address attempting to be accessed from code address aaaa.

Action: Preserve the core (dump) files and report the entire incident context to your GT.M support channel for further analysis.

KILLBYSIGSINFO2

KILLBYSIGSINFO2, iiii process xxxx has been killed by a signal yyyy at address aaaa

Run Time Error: This indicates that the process iiii failed due to a signal, which occurred while attempting a memory access with an instruction at location aaaa.

Action: Preserve the core (dump) files and report the entire incident context to your GT.M support channel for further analysis.

KILLBYSIGSINFO3

KILLBYSIGSINFO3, iiii process xxxx has been killed by a signal yyyy accessing vaddress aaaa

Run Time Error: This indicates that the iiii (GTM, MUPIP, DSE, and so on) process failed due to the yyyy signal, which occurred while attempting to access virtual address aaaa.

Action: Preserve the core (dump) files and report the entire incident context to your GT.M support channel for further analysis.

KILLBYSIGUINFO

KILLBYSIGUINFO, Process xxxx has been killed by a signal yyyy from process zzzz with userid number aaaa

Run Time Error: This indicates that the process failed due to a signal, sent by another process zzzz, owned by user ID aaaa.

Action: Preserve the core (dump) files and report the entire incident context to your GT.M support channel for further analysis.

KRNLKILL

KRNLKILL, Process was terminated by SIGDANGER signal from the system --System swap space is too low --Report to System Administrator

Run Time Error: This indicates that a GT.M process received the SIGDANGER signal from AIX, which means that virtual memory is critically low. This message is sent to the user and the system log, notifying that the GT.M process is terminating.

Action: Report this condition to the system administrator as soon as possible. Check the system configuration to ensure sufficient swap space for the expected workload. Investigate whether the paging space is adequate.

LABELEXPECTED

LABELEXPECTED, Label expected in this context

Compile Time Error: This indicates that GT.M did not find a valid line reference where expected.

Action: Look for a missing label in an extrinsic or in a command such as DO, GOTO, or JOB.

LABELMISSING

LABELMISSING, Label referenced but not defined : xxxx

Compile Time Error: This indicates that a transfer of control command specified a label xxxx that does not exist in the routine.

Action: Look for a missing or misspelled label.

LABELONLY

LABELONLY, Routine xxxx was compiled for label-only entry.

Run Time Error: This indicates that a transfer of control command specified an offset in routine xxxx that was compiled with the `NOLINE_ENTRY` qualifier; therefore, it can be invoked only at a label.

Action: Modify the invocation or recompile the routine without the `NOLINE_ENTRY` qualifier.

LABELUNKNOWN

LABELUNKNOWN, Label referenced but not defined

Compile Time Error: This indicates that a transfer of control command specified a label that is not defined in the image.

Action: Look for an unresolved reference in the last LINK caused by a missing or misspelled label.

LASTFILCMPLD

LASTFILCMPLD, The file currently being compiled is xxxx

Compile Time Information: This indicates that the GT.M compiler encountered a `<CTRL>-C` in the input stream and issued this status. xxxx is the name of the routine the compiler is currently processing.

Action: Use `<CTRL>-Y` to abort the process.

LCKGONE

LCKGONE, Lock removed: xxxx

LKE Success: This indicates that CLEAR removed an M LOCK. xxxx is the resource name.

LCKSCANCELLED

LCKSCANCELLED, Error on remote node holding locks or zallocates. All locks and zallocates cancelled.

Run Time Error: This indicates that when a GT.M process encounters a network error that involves a node holding LOCKs and/or ZALLOCATEs, the process attempts to cancel all LOCKs and ZALLOCATEs regardless of their node.

Action: If GT.M determines that communication with any part of its lock database is suspect, it releases all locks to establish a known state and minimize the impact of the failure on remaining network processes. After this error occurs, ensure that any restart reinstates ALL locks.

LCKSGONE

LCKSGONE, Locks selected for deletion removed

LKE Success: This indicates that CLEAR removed an M LOCK on a remote database.

LCKSTIMOUT

LCKSTIMOUT, DAL timed lock request expired

Run Time Warning: This indicates that a call to seize a named M resource specified a timeout, and the resource was not available within the timeout.

Action: This is a normal signal to the calling process. If it occurs at an inappropriate time, use LKE to examine the lock environment.

LDBINFMT

LDBINFMT, Corrupt binary format header information

MUPIP Error: This indicates that LOAD terminated because of corrupt binary format header information.

Action: Verify whether you have loaded the correct tape and specified the correct file of binary extract format. If you have the right tape and file, use the host shell DUMP command to examine the file header as described in the *MUPIP* chapter of the Administration and Operations Guide. This error can also occur if you attempt to load an extract that the current version of MUPIP does not accept.

LDGOQFMT

LDGOQFMT, Corrupt GOQ format header information

MUPIP Verify Error: This indicates that MUPIP terminated the loading of a GOQ format file because of a corrupt file header.

Action: Ensure that the proper tape is mounted and review how it was created. Use the host shell DUMP command to examine the first few blocks of the tape.

LINKVERSION

LINKVERSION, This image must be relinked with the current version of GT.M

Run Time Fatal Error: This indicates that GT.M attempted to access an image that was created with a previous version of GT.M.

Action: Relink the image using the current version of GT.M. If the previous version of GT.M is still available, adjust the logical names to activate the appropriate old version of GT.M.

LISTENPASSBND

LISTENPASSBND, Controlmnemonic LISTEN can be applied to PASSIVE socket which is in the state BOUND ONLY

Run Time Error: This indicates that the LISTEN control mnemonic can only be applied to passive sockets in a bound state.

Action: Use ZSHOW to verify that the command syntax is correct. Use the USE command to bind the socket.

LITNONGRAPH

LITNONGRAPH, M standard requires graphics in string literals.

Compile-time warning: flags a standard violation. The generated code will accept the string, even though it contains “invisible” characters.

Action: Consider revising the literal to use \$CHAR() and possibly concatenation to make the code more maintainable.

LKENOFINISH

LKENOFINISH, LKE unable to finish all requested actions

LKE Error: This indicates that the previously reported error(s) prevented LKE from completing the requested action.

Action: Review the accompanying error message(s).

LKNAMEEXPECTED

LKNAMEEXPECTED, An identifier is expected after a ^ in this context

Compile Time Error: This indicates that the LOCK command specified an argument that started with ^, but does not contain a valid global name.

Action: Look for and correct any typographical errors or attempted naked references in LOCK names.

LKRUNDOWN

LKRUNDOWN, Error during lock database rundown

Run Time Error: This indicates that the process encountered an error when it attempted to release its LOCKs as part of image termination.

Action: Report this error to the group responsible for database integrity within your organization. Although this is not strictly a database error, other processes can be affected if LOCKs were left behind.

LKSECINIT

LKSECINIT, Error creating lock section for database xxxx

Run Time Error: This indicates that GT.M encountered a problem initializing the lock database associated with the database file (xxxx) it was trying to open.

Action: Review the accompanying message(s) for additional information.

LNKNOTIDLE

LNKNOTIDLE, Attempt to initiate operation before previous operation completed

GT.CM Server Error: This indicates that the networking protocol failed by trying to open an already accessed connection.

Action: Review DECnet error logs to determine the cause and location of the failure. Report the entire incident context to your GT.M support channel.

LOADABORT

LOADABORT, Aborting load at record xxxx

MUPIP Error: This indicates that LOAD encountered an error while processing input record number xxxx.

Action: Refer to the topic MUPIP LOAD Errors in About This Manual section in this manual.

LOADBGSZ

LOADBGSZ, Load error: BEGIN too small. No records loaded.

MUPIP Error: This indicates that an operation initiated by MUPIP LOAD with the qualifier FORMAT=GO did not take place because the record specified for the qualifier BEGIN= is negative or zero.

Action: Specify a record that is within the actual file.

LOADBGSZ2

LOADBGSZ2, Load error: BEGIN too large. No records loaded

MUPIP Error: This error is returned when the BEGIN qualifier's value exceeds the maximum (4GB - 1) value.

Action: Reduce the size of the parameter value and retry.

LOADCTRLY

LOADCTRLY, Control Y encountered during load. Load halting.

MUPIP Warning: This indicates that LOAD encountered a <CTRL>-Y in its input stream and terminated.

Action: The result of the LOAD is incomplete. If the LOAD was with FORMAT=GO, the database is usable. If the LOAD was with FORMAT=BIN, the database may be corrupt.

LOADEDDBG

LOADEDDBG, Load error: END smaller than BEGIN. No records loaded.

MUPIP Error: This indicates that a MUPIP LOAD operation did not occur because the record specified for the qualifier END= is smaller than the record specified for the qualifier BEGIN=.

Action: Specify a record for the qualifier END= that is greater than or equal to the record for the qualifier BEGIN=.

LOADED SZ

LOADED SZ, Load error: END too small. No records loaded.

MUPIP Error: This indicates that a MUPIP LOAD operation did not occur because the record specified for the qualifier END= is smaller than 2.

Action: Modify the qualifier END= value.

LOADED SZ2

LOADED SZ2, Load error: END too large. No records loaded.

MUPIP Error: This error is produced when the END qualifier's value exceeds the maximum (4GB - 1) value.

Action: Reduce the size of the parameter value and retry.

LOADEOF

LOADEOF, Load error: EOF reached prior to BEGIN record xxxx No records loaded

MUPIP Error: This indicates that LOAD did not transfer any records to the database because its input steam reached the end-of-file before the record specified by the qualifier BEGIN=xxxx.

Action: Specify a record for the qualifier BEGIN= that does not exceed the number of records in the file.

LOADFILERR

LOADFILERR, Error with load file xxxx

MUPIP Error: This indicates that LOAD encountered an error when opening its input file xxxx.

Action: Make sure correct load file has been specified. Review the accompanying message(s) for additional information.

LOADFMT

LOADFMT, Load error: bad format type. Must be GO, BINARY, or GOQ.

MUPIP Error: This indicates that a MUPIP LOAD operation did not take place because the qualifier FORMAT= specified an unsupported format.

Action: Look for and correct any typographical errors in the qualifier FORMAT= value.

LOADGD

LOADGD, Loading Global Directory xxxx

GDE Information: GDE displays this message at the beginning of a GDE session when Global Directory xxxx already exists.

LOADINVCHSET

LOADINVCHSET, Extract file CHSET xxx is incompatible with gtm_chset.

MUPIP Information: This indicates that a MUPIP LOAD operation did not take place because the value of the environment variable gtm_chset at the time of creating the extract file was not the same as the current value of gtm_chset.

Action: Determine whether to change the current character set or redo the EXTRACT with a different character set. Alternatively, you can edit the extract file so the EXTRACT file header matches the gtm_chset environment variable. This enables an M mode MUPIP LOAD to treat the input as a byte stream or a UTF-8 mode MUPIP LOAD, which either detects BADCHAR errors or not, depending on the setting of the gtm_badchar environment variable..

LOADRUNNING

LOADRUNNING, Cannot ZLINK an active routine xxxx

Run Time Error: This indicates that a ZLINK specified a routine xxxx, that is currently on the M invocation stack. A routine cannot be altered in the image if its current form may be required for continued processing.

Action: ZLINK the routine prior to or after running it. Use QUIT or ZGOTO to remove the routine from the M stack.

LOCKSPACEFULL

LOCKSPACEFULL, No more room for LOCK slots on database file ffff

Runtime Error: This indicates that the environment attempted more concurrent M LOCKs than the configured LOCK_SPACE for file ffff can support.

Action: Analyze the LOCK protocol for efficiency. Use mupip set -file -lock_space=size ffff to increase the lock space for region xxx. To avoid the same problem the next time you recreate the database, use GDE to make the analogous change to lock_space for the segment mapped to the ffff file in the global directory used to MUPIP CREATE this region

LOCKSPACEINFO

LOCKSPACEINFO, Region: rrrr: processes on queue: pppp/qqqq; LOCK slots in use: llll/kkkk; name space not full

Runtime Error: This indicates that the environment attempted more concurrent M LOCKs than the configured LOCK_SPACE for region rrrr can support. pppp processes are waiting on a lock. llll locks are in use. qqqq and kkkk indicate maximum number of process queue entries, and maximum number of locks respectively.

Action: Analyze the LOCK protocol for efficiency. Use mupip set -region -lock_space=size "rrrr" to increase the lock space for region rrrr. To avoid the same problem the next time you recreate the database, use GDE to make the analogous change to lock_space for the segment mapped to the ffff file in the global directory used to MUPIP CREATE this region.

LOCKSPACEUSE

LOCKSPACEUSE, Estimated free lock space: xxx% of pppp pages.

LKE Information: SHOW command displays the amount of free space along with the number of pages configured for lock space.

Action: If the free lock space report does not show a comfortable amount of free space, use MUPIP SET -LOCK_SPACE to increase the space; remember to also use GDE to revise the LOCK_SPACE in the global directory used to create the region in question so the change remains when the database is recreated.

LOGOFF

LOGOFF, No longer logging to file xxxx

GDE Information: This indicates that a LOG command with the qualifier OFF terminated logging of GDE commands to log file xxxx.

Action: When appropriate, resume logging with LOG and the qualifier ON[=]. GDE closes the log file(s) at the end of the GDE session.

LOGON

LOGON, Logging to file xxxx

GDE Information: This indicates that a LOG command with the qualifier ON[=] initiated the logging of GDE commands to log file xxxx.

Action: You can suspend logging with LOG and the qualifier OFF.

LOGTOOLONG

LOGTOOLONG, Environment variable eeee is too long. Maximum length allowed is llll bytes.

Information Message: This error is triggered whenever the length of an environment variable that GT.M cares about exceeds the maximum allowed limit.

Action: The maximum allowed limit is indicated in the message. Specify a value for the environment variable within this length.

LOWSPACECRE

LOWSPACECRE, Disk space for database file xxxx is not enough for yyyy future extension. aaaa blocks are needed, only bbbb available.

MUPIP Warning: This indicates that the database file xxxx was created but it was found that the file system/volume does not have enough space for even yyyy future extensions.

Action: Check the allocations and extension sizes specified in the Global Directory. If no extensions are anticipated, no action is required. Otherwise, consider moving some files to another file system/volume, or reconfiguring the file system/volume housing the database file.

LPARENMISSING

LPARENMISSING, Left parenthesis expected

Compile Time Error: This indicates that GT.M did not find a left parenthesis in the next source position.

Action: Look for invalid subscripts in indirection operations and errors in SET \$PIECE commands.

LPARENREQD

LPARENREQD, xxxx Left parenthesis expected

MUPIP Error: This indicates that LOAD failed because it found xxxx in the input stream where it expected to find a left parenthesis.

Action: Refer to the topic MUPIP LOAD Errors in About This Manual section of this manual.

LQLENGTHNA

LQLENGTHNA, Listening queue length xxxx not appropriate, it should be between 1 and 5

Run Time Error: This indicates that the GT.M listening queue restricts the number of pending connections between one (1) and five (5).

Action: Specify the number of pending connections in the queue as a number between one and five.

LSEXPECTED

LSEXPECTED, A line separator is expected here

Compile Time Error: This indicates that a source line did not specify a space or tab before the first command.

Action: Look for and correct typographical errors. If missing, put a tab or space at the beginning of the line.

LVNULLSUBS

LVNULLSUBS Null subscripts not allowed in local variables

Run Time Error: This indicates that an attempt was made to set a local variable with null subscript.

Action: Modify the subscript generation to avoid the null subscript or change the LVNULLSUBS parameter for this job or process. For information on changing LVNULLSUBS, refer to documentation on GTM\$DEFAULTS and the VIEW command in the Programmer's Guide.

LVORDERARG

LVORDERARG, Argument to local variable \$NEXT must be subscripted

Compile Time Error: This indicates that a \$NEXT function specified an unsubscripted local variable as an argument.

Action: Use the \$ORDER function or ZWRITE command to display local variables.

LVSTARALON

LVSTARALON, The * name cannot be deleted or renamed

GDE Error: This indicates that a DELETE or RENAME command attempted to delete or rename the * namespace. The * namespace is protected because it is associated with namespaces that are not explicitly mapped.

Action: None.

MAPBAD

MAPBAD, xxxx for yyyy does not exist

GDE Information: This indicates that a NAME points to a REGION or a REGION points to a SEGMENT that does not exist. xxxx is the missing object. yyyy describes the type of the object. When you enter the VERIFY or EXIT command, GDE displays this message after it verifies the global directory.

Action: Use the ADD command to add the REGION or SEGMENT.

MAPDUP

MAPDUP, xxxx and yyyy both map to zzzz

GDE Information: This indicates that GDE encountered two REGIONs mapped to the same SEGMENT or two SEGMENTs mapped to the same FILE. xxxx and yyyy are the REGIONs or SEGMENTs with the same mapping. zzzz is the SEGMENT or FILE with more than one mapping. When you enter the VERIFY or EXIT command, GDE displays this message after it verifies the global directory.

Action: Delete mappings to eliminate duplication.

MAXACTARG

MAXACTARG, Maximum number of actual arguments exceeded

Compile Time Error: This indicates that a DO or extrinsic function supplied an actualist with more than 32 elements.

Action: Modify the routine so that it passes fewer parameters explicitly.

MAXARGCNT

MAXARGCNT, Maximum number of arguments xxxx exceeded

Compile Time Error: This indicates that a command or function specified more than xxxx arguments.

Action: Modify the program so it uses fewer arguments with a single command or function.

Run Time Error: This indicates that a SET of a \$ZROUTINES has more than the allowed number of elements in an array. The maximum number of arguments is xxxx.

Action: Limit the number of elements in the \$ZROUTINES SET argument to fewer than xxxx.

MAXBTLEVEL

MAXBTLEVEL, Global xxxx reached maximum level

MUPIP Error: This indicates that the global-variable-tree for global xxxx reached the maximum level permissible. Very likely, MUPIP REORG was specified with a fill-factor much less than 100. Small fill-factors can cause REORG to distribute existing GDS-

blocks (in order to accommodate the fill-factor requirement), in turn causing block-splits, which might lead to an increase of the tree height.

Action: If MUIPIP reorg was specified with a small fill-factor, try higher number (close to 100) to reduce tree-height. Other techniques include increasing GDS-block-size or killing unwanted portions of the tree.

MAXFORARGS

MAXFORARGS, Maximum number of arguments to a single FOR command exceeded

Compile Time Error: This indicates that a FOR statement specified more than 127 arguments.

Action: Modify the routine so that it uses fewer arguments in one FOR command.

MAXGTMPATH

MAXGTMPATH, The executing module path is greater than the maximum xxxx

Run Time Error: This indicates that the path specified for the mumps executable environment variable has a length limitation of xxxx.

Action: Move the directory or use a link to shorten the path.

MAXNRSUBSCRIPTS

MAXNRSUBSCRIPTS, Maximum number of subscripts exceeded

Compile Time Error: This indicates that a subscripted variable exceeded the maximum limit of 31 subscripts.

Action: Modify the routine to observe this limit on subscripts in a single variable.

MAXSEMGETRETRY

MAXSEMGETRETRY, Failed to get ftok semaphore after tttt tries because it is being continually deleted

Run Time Error: A process was unable to open a database file because on every one of tttt tries it found something kept deleting the IPC semaphore that gates access to the file.

Action: Check for one or more rogue processes disrupting IPC semaphore, or for damage to the Operating System semaphore services.

MAXSSREACHED

MAXSSREACHED, Maximum snapshots - mmmm - for region rrrr reached. Please wait for the existing snapshots to complete before starting a new one.

Mupip error: Starting this snapshot would exceed the maximum number of snapshots.

Action: Wait for a currently active process using snapshots to complete or terminate an existing snapshot activity.

MAXTRIGNEST

MAXTRIGNEST, Maximum trigger nesting level LLLL exceeded

Run Time Trigger Error: GT.M limits trigger invocation depth to LLLL.

Action: If you are sure that you do not have an application code bug or misfeature, reduce the depth of trigger invocation, possibly by consolidating triggers.

MAXSTRLEN

MAXSTRLEN, Maximum string length exceeded

Run Time Error: This indicates that a string exceeded the maximum limit of 1,048,576 bytes. In M mode, each byte holds a character, but in UTF-8 mode, a character may take between one and four bytes.

Action: Modify the routine so it uses shorter strings.

MAXTRACEHEIGHT

MAXTRACEHEIGHT, <The maximum trace tree height (xxxx) has been exceeded. The trace information will be incomplete>

Run time information: The internal GT.M data structure used to gather information during M Profiling can not hold all the information.

Action: Not all lines executed will be reported in the global specified by VIEW "TRACE". There is no impact on the actual execution of the user program. Report the entire incident context with all information necessary to reproduce this error to your GT.M support channel.

MAXTRACELEVEL

MAXTRACELEVEL, The maximum traceable level of xxxx has been exceeded. The frame information will not be maintained.

Run Time Information: This indicates that M profiling reached the maximum level (\$ZLEVEL) it can manage. The information for frames will not be maintained from this point on. Line information is maintained and reported correctly and information gathered about frames upto this point will be reported. Further MAXTRACELEVEL errors will not be reported to the users.

Action: Determine if the number of levels of execution is as intended, if not, correct the application.

MBXRONLY

MBXRONLY, Mailbox is read only, cannot write to it

Run Time Error: This indicates that a WRITE command attempted to access a mailbox that was opened read-only.

Action: Verify that the routine is using the right mailbox and that the mailbox was opened with the appropriate deviceparameter.

MBXWRONLY

MBXWRONLY, Mailbox is write only, cannot read from it

Run Time Error: This indicates that a READ command attempted to access a mailbox that was opened write-only.

Action: Verify that the routine is using the correct mailbox and that the mailbox was opened with the appropriate deviceparameter.

MEMORY

MEMORY, Central memory exhausted during request for xxxx bytes

Run/Compile Time Error: This indicates that the compiler or the run-time system could not allocate sufficient storage.

Action: Look for very large variables. This error can also be caused by problems in the GT.M environment, such as using components of different versions or different platforms. Verify that there is no such problem in the environment.

MEMORYRECURSIVE

MEMORYRECURSIVE, Memory Subsystem called recursively

Run Time Error: This indicates that GT.M made an error calling the memory subsystem.

Action: Report the entire incident context to your GT.M support channel.

MERGEDESC

MERGEDESC, Merge operation not possible. xxxx is descendent of yyyy.

Run Time Error: This indicates that GT.M was not able to MERGE xxxx into yyyy or vice versa, because xxxx is a descendent of yyyy.

Action: Modify the routine to avoid MERGE operation between two variables where one is the descendant of the other.

MMBEFOREJNL

MMBEFOREJNL, BEFORE image journaling cannot be set with MM access method in database file ffff

MUPIP error: MM access method is incompatible with BEFORE_IMAGE journaling.

Action: If you require BEFORE_IMAGE journaling, use the BG access method. If you wish to use MM, turn off BEFORE_IMAGE journaling before selection MM as the access method.

MERGEINCOMPL

MERGEINCOMPL, Error encountered during MERGE; operation may be incomplete

Run Time Error: This indicates that GT.M was not able to complete MERGE operation.

Action: Review the accompanying message(s) for additional information.

MMNOBEFORIMG

MMNOBEFORIMG, MM segments do not support before image journaling

GDE Information: This indicates that a JOURNAL=BEFORE_IMAGE region qualifier appeared on a segment that has segment qualifier ACCESS_METHOD=MM.

Action: Change the segment qualifier to ACCESS_METHOD=BG or use NOBEFORE_IMAGE for the region.

MMNOBFORRPL

MMNOBFORRPL, Replication cannot be used in database file ffff which uses MM access method and NOBEFORE image journaling

MUPIP Error: You can't turn on replication for MM access method database file ffff.

Action: Forgo replication for the file or change the access method to BG.

MMNODYNDWNGRD

MMNODYNDWNGRD, Unable to use dynamic downgrade with MM access method for region xxx. Use BG access method for downgrade

MUPIP/Runtime Error: An attempt was made to use MM mode on a database that has not completed being downgraded. MM mode is only supported on fully downgraded or fully upgraded databases.

Action: Use MUPIP SET FILE or MUPIP SET REGION with the ACCESS_METHOD parameter to set the access mode to BG. Then complete the file downgrade using MUPIP REORG DOWNGRADE or file upgrade using MUPIP REORG UPGRADE. And finally set the access mode back to MM using the MUPIP SET FILE or MUPIP SET REGION command again.

MMNODYNUPGRD

MMNODYNUPGRD, Unable to use MM access method for region yyy until all database blocks are upgraded

MUPIP/Runtime Error: An attempt was made to use MM mode on a database that has not completed being upgraded. MM mode is only supported on fully upgraded databases.

Action: Use MUPIP SET FILE or MUPIP SET REGION with the ACCESS_METHOD parameter to set the access mode to BG. Then complete the file upgrade using MUPIP REORG UPGRADE. And finally set the access mode back to MM using the MUPIP SET FILE or MUPIP SET REGION command again.

MRTMAXEXCEEDED

MRTMAXEXCEEDED, Maximum value of xxxx for SOCKET deviceparameter MOREREADTIME exceeded.

Compile Time/ Run Time Error: GT.M triggers this error when MOREREADTIME exceeds its maximum value of 999ms.

Action: Specify a value between 1 and 999. Never set MOREREADTIME to 0 as it may cause a CPU to "spin". See "Input/Output Processing" Chapter of the Programmer's Guide for more information.

MTANSIFOR

MTANSIFOR, Use of ANSI labels does not allow stream format

Run Time Error: This indicates that the LABEL="ANSI" device parameter appeared in a device command for a tape with STREAM format.

Action: Modify the format to FIXED or VARIABLE, delete the LABEL device parameter, or use LABEL="DOS11" for STREAM format.

MTANSILAB

MTANSILAB, Tape label is not in valid ANSI format

Run Time Error: This indicates that the magnetic tape device driver expected an ANSI label but did not find one.

Action: Verify that the tape was properly made and that current processing device parameters match.

MTBLKTOOBIG

MTBLKTOOBIG, Magtape BLOCK_SIZE exceeds maximum size allowed

Run Time Error: This indicates that an OPEN command specified a BLOCK_SIZE deviceparameter that is larger than this type of tape can accommodate.

Action: Review the routine to ensure that block sizes do not exceed host I/O driver limitations.

MTBLKTOOSM

MTBLKTOOSM, Magtape BLOCK_SIZE is less than xxxx bytes

Run Time Error: This indicates that an OPEN command specified a BLOCK_SIZE device parameter that is smaller than this type of tape can handle.

Action: Review the program to ensure that block sizes do not fall below host driver limitations. For example, the industry standard for 9-track tapes is to treat blocks smaller than 14 bytes as noise in an inter-record gap.

MTDOSFOR

MTDOSFOR, Use of DOS-11 labels requires stream format

Run Time Error: This indicates that the LABEL="DOS11" deviceparameter appeared in a device command for a tape with FIXED or VARIABLE format.

Action: Modify the format to STREAM, delete the LABEL deviceparameter, or use LABEL="ANSI" for FIXED or VARIABLE format.

MTDOSLAB

MTDOSLAB, Tape label is not in valid DOS-11 format

Run Time Error: This indicates that the magnetic tape device driver did not find the expected DOS11.

Action: Verify that the tape was properly made and that the current processing deviceparameters match.

MTFIXRECSZ

MTFIXRECSZ, BLOCK_SIZE xxxx must be a multiple of fixed record size yyyy

Run Time Error: This indicates that an OPEN command specified a BLOCK_SIZE that is not a multiple of its FIXED RECORDSIZE.

Action: Ensure that the block size is evenly divisible by the record size.

MTINVLAB

MTINVLAB, Invalid label type specified in magtape OPEN

Run Time Error: This indicates that the LABEL= deviceparameter appeared in a device command for a tape with an argument evaluating to arguments other than "DOS11" or "ANSI", which are the only valid ones.

Action: Remove or modify the LABEL=.

MTIOERR

MTIOERR, I/O Error with magnetic tape device xxxx

Run Time Error: This indicates that an attempt to access a magnetic tape encountered a hardware error on tape drive xxxx.

Action: Review accompanying system error messages. If appropriate, try another tape or another drive. Initiate diagnostics on the tape drive in question.

MTIS

MTIS, Magnetic tape: xxxx

Run Time Information: This message identifies magnetic tape xxxx as the current device at the time of an error.

Action: Review the accompanying message(s) for additional information.

MTNOSKIP

MTNOSKIP, SKIP operation not supported on this device

Run Time Error: This indicates that the program attempted to use the SKIP deviceparameter for a type of tape that is not able to SKIP.

Action: Remove the SKIP deviceparameter or select a type of tape that supports SKIP.

MTRDBADBLK

MTRDBADBLK, Block read too small, contained only xxxx bytes, block size = yyyy

Run Time Error: This indicates that a READ command encountered a block on the tape that contained xxxx bytes, which is less than the BLOCKSIZE=yyyy deviceparameter specified.

Action: Verify that the proper tape is mounted and that the appropriate BLOCKSIZE is used when OPENing the magtape.

MTRDONLY

MTRDONLY, Cannot write to a READONLY magtape

Run Time Error: This indicates that a WRITE command attempted to access a read-only tape.

Action: Verify that the proper tape is mounted. If the tape is to be written, insert the write ring and remount. If the tape is properly mounted, ensure that the tape was OPENed NOREADONLY.

MTRDTHENWRT

MTRDTHENWRT, Attempt to read after a write to a magtape

Run Time Error: This indicates that a READ command followed a WRITE command to a magnetic tape with no intervening positioning toward the beginning of the tape.

Action: Look for a logic problem or a missing USE command. If previously written data is to be read, the tape must first be repositioned by a REWIND, a SKIP, or a SPACE with negative arguments.

MTRECGTRBLK

MTRECGTRBLK, Magtape record size cannot exceed block size

Run Time Error: This indicates that an OPEN command attempted to initialize a magnetic tape with a RECORDSIZE that exceeds the BLOCKSIZE.

Action: Review the program and modify the value for BLOCKSIZE or RECORDSIZE.

MTRECTOOBIG

MTRECTOOBIG, Magtape record size exceeds maximum allowed

Run Time Error: This indicates that an OPEN command attempted to initialize a magnetic tape with a RECORDSIZE larger than this type of tape can accommodate. Note that the maximum size allowed may depend on whether the records are FIXED or VARIABLE.

Action: Modify the routine to use smaller record sizes.

MTRECTOOSM

MTRECTOOSM, Magtape record size is too small for record type

Run Time Error: This indicates that an OPEN command attempted to initialize a magnetic tape with a variable length record size that is smaller than this type of tape can accommodate.

Action: Modify the routine to use a larger record size or use FIXED length records. For example, in OpenVMS the minimum variable RECORDSIZE requires a single byte of data and 4 bytes of overhead in every variable-length record.

MUBCKNODIR

MUBCKNODIR, MUPIP backup aborted due to error in output directory

MUPIP Error: This indicates that the output directory specified in a BACKUP command could not receive the output file.

Action: Use the host shell commands to verify that the output directory exists, that it is properly protected, and has enough space.

MUDWNGRDNOTPOS

MUDWNGRDNOTPOS, Start VBN value is [xxx] while downgraded GT.M version can support only [yyy]. Downgrade not possible

MUPIP Error: Older versions of GT.M require the first GDS block be at Virtual Block Number yyy but it is at VBN xxx. This is likely due to the file initially being created using a newer version of GT.M and thus cannot be downgraded.

Action: To use the database with an older version of GT.M, it must be extracted with the current version and loaded into the older version both in ZWR format.

MUDWNGRDNRDY

MUDWNGRDNRDY, Database xxx is not ready to downgrade - still yyy database blocks to downgrade

MUPIP Error: A MUPIP DOWNGRADE was attempted when the file-header blks_to_upgrd counter was not equal to the database used block count. This means that not all database blocks have been converted to V4 format.

Action: Before the database file-header can be downgraded, all of the blocks in the database must be downgraded to V4 format. This is normally accomplished with MUPIP REORG DOWNGRADE. If this fails to set the counter correctly, run MUPIP INTEG (not FAST) on the region which will compute and set the correct counter.

MUDWNGRDTN

MUDWNGRDTN, Transaction number 0xaaa in database xxx is too big for MUPIP [REORG] DOWNGRADE. Renew database with MUPIP INTEG TN_RESET

MUPIP Error: A MUPIP DOWNGRADE or MUPIP REORG DOWNGRADE was attempted when the database transaction number was greater than 4,026,531,839 (the TN_RESET warning limit for V4 databases).

Action: Before the database can be downgraded, the transaction number must be reset with the MUPIP INTEG TN_RESET command. This requires standalone access to the database and may take a significant amount of time.

MUFILRNDWNFL

MUFILRNDWNFL, File: xxxx rundown failed

MUPIP Error: This indicates that the RUNDOWN command could not close a database.

Action: This message indicates that information in memory may need to be transferred to disk. Review the accompanying message(s) for additional information.

MUFILRNDWNSUC

MUFILRNDWNSUC, File successfully rundown

MUPIP Success: This indicates that RUNDOWN ensured that the disk file is current.

MUINFOSTR

MUINFOSTR, xxxx : aaaa

MUPIP Information: MUINFOSTR message is issued by a variety of MUPIP commands to inform the user of the command's progress. This indicates the string xxxx has the value aaaa.

Action: None necessary.

MUINFOUINT4

MUINFOUINT4, xxxx : aaaa [0xbbbb]

MUPIP Information: MUINFOUINT4 message is issued by a variety of MUPIP commands to inform the user of the command's progress. This indicates the string xxxx has the decimal value aaaa and hexadecimal value bbbb.

Action: None necessary.

MUINFOUINT6

MUINFOUINT6 <tttt : vvvv [0x!hhhh] ; \$H=dddddd,ttttt

MUPIP Information: This is secondary information message that provides additional context for some other MUPIP message; tttt is explanatory text, vvvv is a numeric value, hhhh is the hexadecimal equivalent of vvvv, ddddd and ttttt are a date and time in \$HOROLOG format.

Action: Refer to the preceding message.

MUINFOUINT8

MUINFOUINT8, xxxx : aaaa [0xbbbb]

MUPIP Information: MUINFOUINT4 message is issued by a variety of MUPIP commands to inform the user of the command's progress. This indicates the string xxxx has the decimal 8-byte value aaaa and hexadecimal 8-byte value bbbb.

Action: None necessary.

MUJNLDBMISSING ●

Last used version: **V5.4-000**

MUJNLDBMISSING, Journal files for required database dddd missing in the MUPIP JOURNAL command

MUPIP Error: MUPIP JOURNAL processing requires journal files for database dddd in order to perform the requested recovery, but the invoking command did not supply a path for those files.

Action: Revise the command to include the appropriate journal specification(s) and reissue it.

MUJNLPREVGEN

MUJNLPREVGEN, Previous generation journal file xxxx included for database file yyyy

MUPIP Information: This indicates that MUPIP included journal file xxxx for database file yyyy for recovery.

MUJNLSTAT

MUJNLSTAT, xxxx at yyyy

MUPIP Information: This displays the system time yyyy, when the step xxxx was executed.

MUJPOOLRNDWNFL

MUJPOOLRNDWNFL, Jnlpool section (id = xxxx) belonging to the replication instance yyyy rundown failed

MUPIP Error: This indicates that an attempt to run-down the shared memory for a journal pool failed; xxxx is the resource ID of the memory and yyyy is the instance designation.

Action: Analyze the preceding messages for additional information on the failure before attempting the run-down again.

MUJPOOLRNDWNSUC

MUJPOOLRNDWNSUC, Jnlpool section (id = xxxx) belonging to the replication instance yyyy successfully rundown

MUPIP Information: This indicates that the journal pool for instance yyyy was successfully closed and removed; xxxx is the shared memory resource id for the pool and yyyy is the instance designation.

MUKILLIP

MUKILLIP, Kill in progress indicator is set for file xxxx, incorrectly marked busy errors should follow

MUPIP Warning: This indicates that the kill-in-progress flag (shows up as KILLS in progress in DSE DUMP file) is set to a non zero value for database file xxxx.

Action: If there are no accompanying integrity errors, no action is required. Else fix those integrity errors and then perform a MUPIP INTEG -F[AST] -FILE on the database which will then reset the kill-in-progress flag to zero.

MULOGNAMEDEF

MULOGNAMEDEF, logical name xxxx, needed to start replication server is already defined for this job. Check for an existing or improperly terminated server.

MUPIP Error: This indicates that the logical name xxxx is already defined, which prevents MUPIP from starting the replication server. Either there is an already running server, or a previous server was not properly terminated.

Action: Check for an existing or improperly terminated server, use MUPIP RUNDOWN to clean up.

MULTFORMPARM

MULTFORMPARM, This formal parameter is multiply defined

Compile Time Error: This indicates that an element appears more than once in a formallist.

Action: Modify the formallist.

MULTLAB

MULTLAB, This label has been previously defined

Compile Time Error: This indicates that a label is defined more than once in the routine.

Action: Rework the labels so that each one is unique. If labels contain more than eight characters, they are truncated to eight characters, which can cause conflicts.

MUNOACTION

MUNOACTION, MUPIP unable to perform requested action

MUPIP Error: This indicates that MUPIP encountered an error, which prevented the requested action.

Action: Review the accompanying message(s) for additional information to identify the cause.

MUNODBNAME

MUNODBNAME, A database name or the region qualifier must be specified

MUPIP Error: This indicates that a MUPIP command did not have a FILE or REGION or JNLFILE qualifier.

Action: Add one of the required qualifiers to the command.

MUNODWNGRD

MUNODWNGRD, MUPIP downgrade did not occur because of preceding errors

MUPIP Error: This indicates that MUPIP failed to downgrade a database.

Action: Review the accompanying message(s) for additional information.

MUNOFINISH

MUNOFINISH, MUPIP unable to finish all requested actions

MUPIP Error: This indicates that MUPIP encountered an error, which prevented the requested action from completing. The action has partially completed.

Action: Review the accompanying message(s) for additional information to identify the cause.

MUNOSTRMBKUP

MUNOSTRMBKUP, Database xxxx has a block size larger than yyyy and thus cannot use stream (incremental) backup

MUPIP Warning: GT.M does not support bytestream (a.k.a incremental) backup of a database file that is created with a GDS block size larger than xxxx. As of GT.M version V5.0-000, this limit is 32256 bytes. MUPIP CREATE issues MUNOSTRMBKUP warning when creating a database file with a block size that exceeds the limit. MUPIP BACKUP -BYTESTREAM issues MUNOSTRMBKUP error and skips backing up a file that has block size that exceeds the limit. NOTE: Comprehensive BACKUP does not impose any limit on the GDS block size of the database file being backed up.

Action: Create the database file with a block size that does not exceed the limit.

MUNOTALLSEC

MUNOTALLSEC, WARNING: not all global sections accessed were successfully rundown

MUPIP Warning: This indicates that RUNDOWN encountered at least one database that appeared to be in use and therefore could not be processed.

Action: If appropriate, initiate actions to cause all GT.M users to exit from GT.M and repeat the MUPIP RUNDOWN.

MUNOUPGRD

MUNOUPGRD, MUPIP upgrade did not occur because of preceding errors

MUPIP Error: This indicates that MUPIP could not upgrade a database from one version to another.

Action: Review the preceding error messages. Review all the release notes for the new version and for all versions between it and the existing version.

MUPCLIERR

MUPCLIERR, Action not taken due to CLI errors

MUPIP Error: This indicates that a MUPIP command did not process because of invalid syntax.

Action: Review the command documentation for correct syntax. Review the accompanying message(s) if any, for more information about the cause of this error.

MUPGDERR

MUPGDERR, Command aborted due to global directory errors

MUPIP Error: This indicates that a MUPIP command failed because it required a Global Directory and that file was inaccessible or damaged.

Action: Verify that GTM\$GBLDIR / gtmgbldir identifies the proper file and that the file is accessible to the process. Use GDE to recreate the Global Directory, if necessary.

MUPGRDSUCC

MUPGRDSUCC, Database file xxx successfully yyy to zzz

MUPIP Information: The database file header for xxx has been upgraded or downgraded to the version zzz format.

MUPIPINFO

MUPIPINFO, xxxx

MUPIP Information: The LOAD command with the FORMAT qualifier defined to GO or GOQ uses this message to display the input file label xxxx.

MUIPSIG

MUIPSIG, STOP (signal xxxx) issued from process yyyy to process zzzz

MUPIP Information: This message indicates that process yyyy issued a MUPIP STOP to process zzzz, and xxxx signal is sent to process zzzz. Note that the message is logged at the time the signal is sent, regardless of when or if it is processed by process zzzz.

MUPJNLINTERRUPT

MUPJNLINTERRUPT, Database file xxxx indicates interrupted MUPIP JOURNAL command. Restore from backup for forward recovery.

MUPIP Information: This indicates that MUPIP Journal command did not proceed because a previous MUPIP JOURNAL command was terminated abnormally.

Action: The failed MUPIP command may be reissued if appropriate, else restore the database and journal files from backup to proceed with RECOVER FORWARD.

MUPRECFLCK

MUPRECFLCK, Database file xxxx is locked by MUPIP RECOVER. Could not secure access.

Run Time Error: This indicates that GT.M could not open a database file xxxx because MUPIP JOURNAL with the RECOVER qualifier was applying a journal to the file.

Action: Wait for RECOVER command to complete.

MUPRESTERR

MUPRESTERR, MUPIP RESTORE aborted due to preceding errors

MUPIP Error: This indicates that a RESTORE operation failed, which left the database in an indeterminate state.

Action: Review the preceding errors for additional information.

MUQUALINCOMP

MUQUALINCOMP, Incompatible qualifiers - FILE and REGION

MUPIP Error: This indicates that the FILE and REGION qualifiers cannot be used in the same command.

Action: Choose one or the other.

MURAIMGFAIL

MURAIMGFAIL, MUPIP RECOVER failed while processing after-image journal record. Failure code: xxxx.

MUPIP Error: This indicates that MUPIP RECOVER/ROLLBACK encountered an error when processing an after-image journal record written for each DSE database update. xxxx contains the failure codes for the four attempts. It is very likely that the database may have integrity errors or that the process-private data structure are corrupted.

Action: Attempt the MUPIP RECOVER/ROLLBACK again. If the error persists, report the entire incident context with as much information about the system as possible to your GT.M support channel.

MUREORGFALL

MUREORGFALL, MUPIP REORG failed. Failure code: xxxx.

MUPIP Error: This indicates that a REORG encountered a database error with failure code xxxx.

Action: Report this error to the group responsible for database integrity within your organization.

MUREPLPOOL

MUREPLPOOL, Error with replpool section xxxx

MUPIP Error: This indicates that MUPIP RUNDOWN command found the specified replication pool shared memory section having a problem.

Action: Refer to the subsequent message text for details.

MUREPLSECDEL

MUREPLSECDEL, Replication section xxxx deleted

MUPIP Information: This indicates that a replication pool was successfully closed and removed; xxxx is the shared memory resource ID for the pool.

MUREPLSECNOTDEL

MUREPLSECNOTDEL, Replication section xxxx not deleted

MUPIP Error: This indicates that an attempt to rundown the shared memory for a replication pool failed; xxxx is the resource ID of the memory.

Action: Review the preceding messages for additional information on the failure before attempting a rundown again.

MUREUPDWNGRDEND

MUREUPDWNGRDEND, Region xxxx : MUPIP REORG UPGRADE/DOWNGRADE finished by pid aaaa [0xbbbb] at transaction number [0xcccc]

MUPIP Information: This is an informational message printed by MUPIP REORG UPGRADE or DOWNGRADE when the reorg has successfully completed its upgrade or downgrade respectively.

Action: None necessary.

MURPOOLRNDWNFL

MURPOOLRNDWNFL, Recvpool section (id = xxxx) belonging to the replication instance yyyy rundown failed

MUPIP Error: This indicates that an attempt to rundown the shared memory for a receive pool failed; xxxx is the resource ID of the memory and yyyy is the instance designation.

Action: Review the preceding messages for additional information on the failure before attempting a rundown again.

MURPOOLRNDWNSUC

MURPOOLRNDWNSUC, Recvpool section (id = xxxx) belonging to the replication instance yyyy successfully rundown

MUPIP Information: This indicates that the receive pool for the specified instance was successfully closed and removed; xxxx is the shared memory resource ID for the pool and yyyy is the instance designation.

MUSECDEL

MUSECDEL, Section xxxx deleted

MUPIP Information: This indicates that RUNDOWN removed the global memory section xxxx that is associated with an inactive database.

MUSECNOTDEL

MUSECNOTDEL, Section xxxx not deleted

MUPIP Information: This indicates that RUNDOWN could not eliminate the global memory section xxxx that is associated with an apparently inactive database.

MUSELFBKUP

MUSELFBKUP, Database file xxxx can not be backed upon itself

MUPIP Error: This indicates that GT.M attempted to perform a backup that would have overlaid the database being backed up.

Action: Modify the name of the output file and reissue the command.

MUSTANDALONE

MUSTANDALONE, Could not get exclusive access to xxxx

MUPIP Information: This indicates that the process required but could not get exclusive access to the listed resource.

Action: Retry the process at a time when there are no other users accessing the resource or log off the resource users.

MUTEXERR

MUTEXERR, Mutual Exclusion subsystem failure

Run Time Error: This indicates that GT.M encountered a system error while initializing mutual exclusion resource(s).

Action: Review the accompanying message(s) for more information about the cause of the error.

MUTEXFRCDTERM

MUTEXFRCDTERM, Mutual Exclusion subsystem detected forced termination of process xxxx. Crit salvaged from region yyyy.

Run Time Warning: This indicates that GT.M confirmed inappropriate termination of the process xxxx, while holding crit on region yyyy.

Action: Determine the cause of the termination and take appropriate action.

MUTEXLCKALERT

MUTEXLCKALERT, Mutual Exclusion subsystem ALERT - Lock attempt threshold crossed for region xxxx. Process yyyy is in crit.

Run Time Warning: This indicates that a process attempting to access a crit for the region xxxx has waited longer than GT.M determined threshold to get crit for a region. This is caused if another process is either currently using the crit, or was killed while using the crit and its process ID is reused.

Action: Monitor the system to determine if the process has been killed or not. Identify and kill the offending process if necessary, and release the permissions for that resource.

MUTEXRELEASED

MUTEXRELEASED, Process xxxx [aaa] has released the critical section for database yyyy to avoid deadlock. \$TLEVEL: pppp
t_tries: qqqq

Run Time Information: This indicates an out-of-design state within GT.M that was recoverable.

Action: If this message is frequent, report the entire incident context to your GT.M support channel.

MUTEXSRCCLNUP

MUTEXSRCCLNUP, Mutex subsystem leftover resource xxxx removed.

Run Time Information: This indicates that GT.M removed leftover system resource xxxx, used by the mutual exclusion subsystem. The resource was leftover due to abnormal termination of a GT.M component.

MUTNWARN

MUTNWARN, Database file xxxx has 0xaaa more transactions to go before reaching the transaction number limit (0xbbbb). Renew database with MUPIP INTEG TN_RESET.

MUPIP Warning: This indicates that MUPIP INTEG detected that the transaction numbers in the named database are approaching the maximum number as specified by the Maximum TN Warn field in the database file header. The actual maximum TN is less than this theoretical limit. DSE DUMP FILEHEADER shows what the limit is. The actual limit reflects some overhead used, for example, during a TN_RESET operation.

Action: Use MUPIP INTEG with the qualifier TN_RESET to reset the transaction numbers in the database. If the database is in the V4 format, consider converting it to the V5 format.

MUTRUNC1ATIME

MUTRUNC1ATIME, Process with PID iiii already performing truncate in region rrrr

MUPIP Information: Issued when a REORG -TRUNCATE on a region rrrr detects some other active REORG process concurrently processing a truncation.

Action: No action required. The other process will complete the truncate.

MUTRUNCBACKINPROG

MUTRUNCBACKINPROG, Truncate detected concurrent backup in progress for region rrrr

MUPIP Information: REORG truncate process detected concurrent backup. Database file not truncated.

Action: Ensure the backup has completed and rerun MUPIP REORG -TRUNCATE command.

MUTRUNCERROR

MUTRUNCERROR, Truncate of region rrrr encountered service error eeee

MUPIP Error: This indicates that a system call failed during REORG truncate.

Action: Use the OS documentation to investigate the failure.

MUTRUNCFAIL

MUTRUNCFAIL, Truncate failed after reorg

MUPIP Error: This indicates that REORG encountered an unexpected error. Truncate may be partially complete.

Action: Review accompanying message(s) for more information.

MUTRUNCNOSPACE

MUTRUNCNOSPACE, Region rrrr has insufficient space to meet truncate target percentage of yyyy

MUPIP Information: Issued when REORG truncate determines that there is not enough free space at the end of the file; database file not truncated.

Action: If appropriate specify a larger threshold.

MUTRUNCNOTBG

MUTRUNCNOTBG, Region rrrr does not have access method BG

MUPIP Error: The truncate feature is only supported with the BG access method.

Action: Use the BG access method for files you wish to truncate.

MUTRUNCNOV4

MUTRUNCNOV4, Region rrrr is not fully upgraded from V4 format.

MUPIP Error: The truncate feature is only available for fully upgraded database files.

Action: In order to use truncate, first upgrade the database file to the current major version.

MUTRUNCPERCENT

MUTRUNCPERCENT, Truncate threshold percentage should be from 0 to 99

MUPIP Error: This indicates the the value entered for MUPIP REORG -TRUNCATE is invalid.

Action: Specify a valid threshold percentage.

MUTRUNCSSINPROG

MUTRUNCSSINPROG, Truncate detected concurrent snapshot in progress for region rrrr

MUPIP Information: REORG truncate process detected concurrent snapshot; database file not truncated.

Action: Ensure snapshot, for example a MUPIP INTEG, has completed and rerun the MUPIP REORG -TRUNCATE command.

MUTRUNCSUCCESS

MUTRUNCSUCCESS, Database file dddd truncated from oooo blocks to nnnn at transaction tttt

MUPIP Information: This operator log message indicates that the specified database file was truncated by MUPIP REORG as described by the message.

MUUPGRDNRDY

MUUPGRDNRDY, Database xxx has not been certified as being ready to upgrade to yyy format

MUPIP Error: The named database file is in an older format than is in use by this GT.M version and has not been certified as ready for use by this GT.M version.

Action: Run DBCERTIFY to certify the database as being ready for upgrade.

NAMEEXPECTED

NAMEEXPECTED, A local variable name is expected in this context

Compile Time Error: This indicates that an actualname or a formallist item did not specify a local variable name.

Action: Look for and correct typographical errors. Verify that actualnames and formallist items are local variable names.

NAMSTARTBAD

NAMSTARTBAD, xxxx must start with '%' or an alphabetic character

GDE Error: This indicates that an ADD or CHANGE command specified a name xxxx that does not begin with a '%' sign or an alphabetic character.

Action: Add a '%' sign or an alphabetic character to the beginning of the name.

NCTCOLLDIFF

NCTCOLLDIFF, Source and destination for MERGE cannot have different numerical collation type

Run Time Error: This indicates that two arguments of the MERGE command have different numerical collation type.

Action: Use the %GBLDEF utility to set the same numerical collation type for both the arguments or use another method, such as a \$ORDER() loop or MUPIP EXTRACT and LOAD to move the data.

NEGFRACPWR

NEGFRACPWR, Invalid operation: fractional power of negative number

Run Time Error: This indicates that the power of an exponentiation operation is negative and contains a fractional portion. This type of operation produces an imaginary component in its result, and M does not specify such operations.

Action: Modify the code to prevent negative powers with fractional parts in exponentiation operations or trap the resulting errors.

NESTFORMP

NESTFORMP, Formal parameter list cannot be combined with nested line

Compile Time Error: This indicates that a line included both a formallist and a nesting level-indicator (.).

Action: Parameter passing is incompatible with argumentless DO commands. Remove the formallist or the level-indicator and reorganize the routine, if appropriate.

NETDBOPNERR

NETDBOPNERR, Error while attempting to open database across net

Run Time Error: This indicates that the GT.M encountered an error when it attempted to open a database that is on a remote node served by a GT.CM server.

Action: Review subsequent message(s) to determine the nature of the problem.

NETFAIL

NETFAIL, Failure of Net operation

Run Time Error: This indicates that a network failure occurred but it could not be traced to any current activity.

Action: If the problem persists, contact the group responsible for database operations on your network.

NETLCKFAIL

NETLCKFAIL, Lock operation across Net failed

Run Time Error: This indicates that a LOCK, ZALLOCATE, or ZDEALLOCATE that involved a remote database failed.

Action: This network failure involves M LOCKs. Retry the operation from a point that establishes all necessary LOCKs. If the problem persists, contact the group responsible for database operations on your network.

NEWJNLFILECREAT

NEWJNLFILECREAT, Journal file xxxx nearing maximum size. New journal file created.

Run Time/MUPIP Information: This indicates that GT.M created a new journal file as it reached the maximum allowed journal size.

Action: Refer to the documentation on maximum allowed journal file size.

NLMISMATCHCALC

NLMISMATCHCALC, Location of xxxx expected at yyyy, but found at zzzz

Run Time Error: This indicates that the shared memory location of xxxx shows a layout problem. Typically, this is caused by attempting to use databases opened by a GT.M version, different from the currently running version.

Action: Rundown the database and ensure a stable and consistent database configuration before attempting to use it again.

NOACTION

NOACTION, Not updating Global Directory xxxx

GDE Information: This indicates that GDE did not write a new version of existing Global Directory xxxx due to a QUIT or an EXIT when no changes had been made.

Action: GDE displays this message when you EXIT GDE without making any changes to the Global Directory. It also displays this message when you terminate a GDE session with the QUIT command. If you made changes you want to save, you must restart GDE , perform your work and save the changes before exiting GDE.

NOALIASLIST

NOALIASLIST, Parenthetical lists of multiple arguments cannot have a preceding alias introducer or include alias (*) forms

Run Time Error: This indicates the argument for a SET command attempted to assign an alias using a parenthesized list as a left-hand argument, which is unsupported syntax.

Action: Correct the code in question to avoid the parenthesized list.

NOCANONICNAME

NOCANONICNAME, Value is not a canonic name (xxxx).

Run Time Error: This indicates that the argument supplied to \$QLENGTH, or the first argument to \$QSUBSCRIPT is not a valid glvn.

Action: Pass valid argument to \$QLENGTH/\$QSUBSCRIPT

NOCCPPID

NOCCPPID, Cannot find CCP process ID

CCE Error: This indicates that a CCE DUMP did not complete because it could not find a process with the name for the CCP.

Action: The CCP is not running properly on your node. Report this error to the group responsible for clustered databases at your site.

NOCHLEFT

NOCHLEFT, Unhandled condition exception (all handlers exhausted) process terminating

Run Time Fatal Error: This indicates an internal error in handling of error conditions in GT.M.

Action: Report the entire incident context with the complete operator log generated to your GT.M support channel.

NOENDIANCVT

NOENDIANCVT, Unable to convert the endian format of file dddd due to eeee

MUPIP error: One of the requirements for the MUPIP ENDIANCVT command was not met. The problems include: "database format is not the current version", "minor database format is not the current version", "some blocks are not upgraded to the current version", "kills in progress", "the database is frozen", "a GT.CM server accessing the database", "recovery was interrupted", "database creation in progress", "wc_blocked is set- rundown needed", "the database is corrupted".

Action: Resolve the reported conditions and repeat the command.

NOEXCLUDE

NOEXCLUDE, None of the excluded variables exist

MUPIP Information: This indicates that MUPIP REORG did not find any of the variables specified in the EXCLUDE qualifier to be present in the database.

Action: Verify the names specified in the EXCLUDE qualifier in case you expected them to be present in the database file and not be reorged.

NOEXCNOZTRAP

NOEXCNOZTRAP, Neither an exception nor a Ztrap is specified

Run Time Warning: This indicates that a \$CTRAP character arrived but no EXCEPTION or \$ZTRAP existed to handle it.

Action: Determine why these circumstances coincide. This error never appears on a device; in this particular case, it is assigned to the image termination status as a warning.

NOEXIT

NOEXIT, Cannot exit because of verification failure

GDE Information: This indicates that GDE encountered errors in the REGION-SEGMENT or SEGMENT-FILE mappings and cannot exit.

Action: Review the accompanying message(s) for additional information. Verify the mappings and modify them as appropriate.

NOFORKCORE

NOFORKCORE, Unable to fork off process to create core. Core creation postponed.

Run Time Warning: This indicates that the process, which failed was unable to create a memory dump file, possibly due to lack of system resources.

Action: Reduce the number of users and stop any unnecessary processes.

NOJNL

NOJNL, ssss segments do not support journaling.

MUPIP Error: This error indicates that the segment type ssss does not support journaling.

Action: For more information on Journaling, refer to the GT.M Journaling chapter in the Administration and Operations Guide.

NOJNLPOOL

NOJNLPOOL, No journal pool info found in the replication instance of xxxx

Run Time / MUPIP Error: This indicates that GT.M / MUPIP did not get replication information from the instance file specified. Replication instance file was not initialized because replication did not start, or some other process reset the replication instance file.

Action: Start the source server if it was not started. If the source server was running, stop the server and perform MUPIP RUNDOWN (if MUPIP RUNDOWN fails, try MUPIP RUNDOWN region *).

NOLBRSRC

NOLBRSRC, Object libraries cannot have SRC paths associated

Run Time Error: This indicates that GTM\$ROUTINES / gtmroutines or a SET \$ZROUTINES attempted to place a source specification (SRC qualifier / source directory path) on an object library.

Action: Remove the source specification. GT.M does not use the qualifier SRC= or source directories on object libraries. On OpenVMS, if you must provide access to source corresponding to objects in an object library, move the objects to a directory. If the objects exist only in the library, use LIBRARY with the qualifier EXTRACT to recreate them. On UNIX, if you must provide access to sources corresponding to objects in the shared library, attach the source directory to an existing object directory entry. Since GT.M does not support automatic recompilations into libraries, care must be taken when providing access to sources of library routines.

NOLOCKMATCH

NOLOCKMATCH, No matching locks were found in rrrr

LKE Information: SHOW or CLEAR, found that no LOCKs match the specified criteria in region rrrr; note that specifying no search criteria acts like a wildcard, checking all LOCKs in the region.

Action: If this is not the expected result, check the search criteria and / or research the LOCK protocol to validate its correct operation.

NOLOG

NOLOG, Logging is currently disabled. Log file is xxxx.

GDE Error: This indicates that GDE is not logging user activities.

Action: Use the LOG -ON command to turn on GDE logging.

NONASCII

NONASCII, ssss is illegal for a oooo as it contains non-ASCII characters

GDE Error: The specification ssss contains non-ASCII characters which are required for an object of type oooo.

Action: Chose an object name or value containing only ASCII characters.

NONEGATE

NONEGATE, Qualifier xxxx cannot be negated

GDE Error: This indicates that the qualifier does not support this usage.

Action: Review the Administration and Operations Guide or the Programmer's Guide for the correct usage.

NONUTF8LOCALE

NONUTF8LOCALE, Locale has character encoding (cccc) which is not compatible with UTF-8 character set

Run Time Error: This error is reported by GT.M when it recognizes that the LC_CTYPE locale category cccc (as shown by the UNIX locale command) does not use UTF-8 character encoding when gtm_chset is "UTF-8".

Action: Set the environment variable LC_CTYPE to a Unicode locale name with UTF-8 character encoding. Note that LC_ALL, if defined, overrides LC_CTYPE. The name of the locale varies between different UNIX platforms, but mostly in the form of <lang>_<country>.<charset>, where each element (without the angular brackets) has the form shown below:

- <lang> is the language code in lower case (such as en, or de).
- <country> is the country name in upper case (such as US, GB)
- <charset> is the character set encoding (such as UTF-8, ISO8859-1)

Refer to the operating system manuals for the specific details of available locale names on the system.

NOPINI

NOPINI, PINI journal record expected but not found in journal file xxxx at offset yyyy

MUPIP Error: This indicates that MUPIP did not encounter a valid and expected Process Initialization Record (PINI), at offset yyyy of the journal file xxxx.

Action: Run MUPIP JOURNAL EXTRACT -FULL DETAIL FORWARD FENCE=NONE NOERROR_LIMIT xxxx. Report the entire incident context with the appropriate extract file and error message(s) to your GT.M support channel.

NOPLACE

NOPLACE, Line specified in a ZBREAK cannot be found

Run Time Warning: This indicates that the ZBREAK was ignored because it specified a line that could not be found in the image.

Action: Modify the ZBREAK or ZLINK the routine that has the missing label.

NOPREVLINK

NOPREVLINK, Journal file xxxx has a null previous link

MUPIP Error: This indicates that MUPIP found the previous link of journal file xxxx to be null, when it needed to process backward beyond the journal files first record.

Action: Verify the specified resync or time qualifiers are as intended. If correct values were specified, and null link was due to operator action, restore the journal generation link and reissue the command.

NOPRINCIO

NOPRINCIO Unable to write to principal device

Run Time Fatal Error: This indicates that GT.M attempted to enter direct mode but it could not write to the principal device.

Action: Determine whether the process was intended to use direct mode. Examine how the code handles the principal device.

NORECVPOOL

NORECVPOOL, No receiver pool info found in the replication instance of xxxx

Run Time / MUPIP Error: This indicates that GT.M / MUPIP did not get replication information from the instance file specified. Replication instance file was not initialized because receiver server did not start or, some other process reset the replication instance file.

Action: Start the receiver server if the server was not started. If the receiver server was running, stop the server and perform MUPIP RUNDOWN (if MUPIP RUNDOWN fails, try MUPIP RUNDOWN region *).

NOREGION

NOREGION, REGION not found: xxxx

LKE Error: This indicates that a SHOW or CLEAR command with the qualifier REGION=xxxx could not locate region xxxx in the current Global Directory.

Action: Look for and correct any typographical errors in the region name. Use GDE to look in the Global Directory for names of defined regions.

NOREPLCTDREG

NOREPLCTDREG, Replication subsystem found no region replicated for dddd ffff

MUPIP Warning: This indicates that the replication system is present, but no globals are configured for replication in ffff where dddd is "instance file" for UNIX and "global directory" for OpenVMS.

Action: Use MUPIP SET to specify which database regions to replicate.

NORESYNCSUPPLONLY

NORESYNCSUPPLONLY, NORESYNC only supported for Supplementary Instances

MUPIP Error: Issued by a Receiver Server on a non-Supplementary Instance when it is started with a -NORESYNC; -NORESYNC only applies to Supplementary Instances started with -UPDOK, not to non-Supplementary Instances.

Action: Use this qualifier only in a valid context.

NORESINCUPTATERONLY

NORESINCUPTATERONLY, NORESYNC qualifier only allowed on a Supplementary Instance which allows local updates

MUPIP Error: Issued by a Receiver Server when started with -NORESYNC on a Supplementary Instance but started without -UPDOK; -NORESYNC applies only to Supplementary Instances started with -UPDOK.

Action: Use this qualifier only in a valid context.

NOSELECT

NOSELECT, None of the selected variables exist, halting

MUPIP Information: This indicates that a MUPIP EXTRACT or REORG operation did not occur because the global variables specified by the SELECT= qualifier do not exist.

Action: Look for an inappropriate definition for GTM\$GBLDIR / gtmgbldir or typographical errors in the specified variables.

NOSOCKETINDEV

NOSOCKETINDEV, There is no socket in the current socket device

Run Time Error: This indicates that either no sockets have been established for the device or that all the sockets attached to the device have been closed prior to the current READ command.

Action: Review the logic managing the sockets and correct it.

NOSPACECRE

NOSPACECRE, Not enough space to create database file xxxx. aaaa blocks are needed, only bbbb available.

MUPIP Error: This indicates that the requested file was not created because the file system (in UNIX) or volume (in OpenVMS) did not have sufficient space.

Action: Check the allocation size specified in the global directory. Choose another location or reconfigure the file system or volume.

NOSPACEEXT

NOSPACEEXT, Not enough disk space for file xxxx to extend. aaaa blocks needed. bbbb blocks available.

Run Time Error: This indicates that there is not adequate space to do a needed journal file extension of the currently specified extension size of aaaa. In OpenVMS, this causes journaling to be turned off for the region.

Action: Locate appropriate disk space and adjust the journal file path. To reestablish durability, perform a MUPIP BACKUP that turns journaling on again.

NOSTARFILE

NOSTARFILE, Only star(*) argument can be specified with xxxx

MUPIP Error: This indicates that the qualifier xxxx, specified with the MUPIP JOURNAL command allows only star (*) as an argument.

Action: Specify star (*) as an argument instead of explicit journal file names.

NOSUBSCRIPT

NOSUBSCRIPT, No such subscript found (xxxx)

Run Time Error: This indicates that the second argument to \$QSUBSCRIPT is less than -1.

Action: Pass a value greater than -1 as the second argument to \$QS.

NOSUCHPROC

NOSUCHPROC, Process xxxx does not exist no need to yyyy it

Run Time Information: This indicates the specified process xxxx does not exist, to which an attempt to signal yyyy was made. This may occur in normal operation, but is reported to the operator logging facility in case an abnormal situation needs to be studied.

NOSUPPLSUPPL

NOSUPPLSUPPL, Instance ssss is configured to perform local updates, so it cannot receive from Supplementary Instance iiiii

MUPIP Error: Issued by a Receiver Server or a MUPIP JOURNAL -ROLLBACK -FETCHRESYNC on a Supplementary Instance ssss started with -UPDOK attempted to connect to instance iiiii, but found IIII is also a Supplementary Instance. A Supplementary Instance that permits local updates can only replicate updates that originate on a non-Supplementary Instance.

Action: Reconfigure the instances to a supported configuration.

NOTALLDBOPN

NOTALLDBOPN, Not all required database files were opened

MUPIP Fatal Error: This indicates that all databases needed for replication could not be opened; the server will not start.

Action: Refer to the accompanying message(s) to determine why all required files would not open. Fix the problem and retry.

NOTALLJNLEN

NOTALLJNLEN, Journaling not enabled and on for all regions

MUPIP Warning: This indicates that MUPIP found that not all regions had journaling enabled.

NOTALLREPLON

NOTALLREPLON, Replication state is not on for all regions

MUPIP Warning: This indicates that not all the database of the current global directory has replication state ON.

NOTERMENTRY

NOTERMENTRY, TERM = "xxxx" has no "terminfo" entry. Possible terminal handling problems.

Run Time Information: This indicates that while opening a terminal device, the value of the environment variable TERM was xxxx for which no matching entry was found in the terminfo database. GT.M uses this information in the terminfo entry to perform terminal specific functions such as cursor movement on screen clearing. With an incorrect entry, such functions are not performed properly.

Action: Exit GT.M and set the TERM environment variable to a value which exists in the terminfo database and which matches the terminal or terminal emulator being used. See the UNIX user documentation for more information about terminfo.

NOTERMENV

NOTERMENV, Environment variable TERM not set. Assuming "unknown."

Run Time Information: This indicates that the TERM environmental variable indicating the terminal type in use does not have a value specified.

Action: Find the correct value for the TERM environmental variable for the terminal in use and specify that terminal type.

NOTERMINFODB

NOTERMINFODB, No "terminfo" database. Possible terminal handling problems.

Run Time Information: This indicates that the operating system could not find the terminfo database. The database may be deleted or moved to different location. GT.M needs this database to display information and accept user input correctly.

Action: This message reflects an operating system problem. Refer to the operating system troubleshooting.

NOTEXTRINSIC

NOTEXTRINSIC, Quit does not return to an extrinsic function, argument not allowed

Run Time Error: This indicates that a QUIT command specified an argument but did not match to an extrinsic function or special variable.

Action: Look for a missing double space after a QUIT, a faulty logic path, or a routine that should be invoked as an extrinsic but was invoked with a DO.

NOTGBL

NOTGBL, xxxx "^" Expected

Run Time/MUPIP Error: This indicates that VIEW argument expression for tracing is not a valid global name. In case of MUPIP error, it indicates that LOAD aborted because it encountered xxxx in its input stream, which is not a valid global name.

Action: Correct the argument of the VIEW command to point to a valid global name. For MUPIP error, refer to the topic MUPIP LOAD Errors in About This Manual section of this manual.

NOTPOSITIVE

NOTPOSITIVE, xxxx qualifier must be given a value greater than zero

MUPIP Error: This indicates that the value assigned to the xxxx qualifier value is negative (less than zero).

Action: Assign a value greater than zero (0) for qualifier xxxx.

NOTPRINCIO

NOTPRINCIO, Output currently directed to device xxxx

Run Time Warning: This message displays the current device xxxx when it is not the principal device and the process enters Direct Mode.

Action: To redirect all I/O to the terminal, note the current device or save it in a temporary variable and USE \$P. If you decide to resume program execution, remember to restore the current device with a USE command.

NOTREPLICATED

NOTREPLICATED, Transaction number xxxx generated by the yyyy process (PID = zzzz) is not replicated to the secondary

DSE Information: This indicates that a transaction generated by DSE update was not replicated to secondary side. This is an expected behaviour. Inappropriate DSE updates while running replication can cause primary and secondary to be out of sync.

NOTRNDMACC

NOTRNDMACC, Only random access files are supported as backup files for non-incremental backup

MUPIP Error: This indicates that for comprehensive BACKUP, only random access files are supported, other types of devices, for example, TCP devices and pipes, are not supported.

Action: Perform the BACKUP to a random access file.

NOTTOEOFONPUT

NOTTOEOFONPUT, Not positioned to EOF on write (sequential organization only)

Run Time Error: This indicates that a WRITE command attempted to update a sequential disk file that was not positioned to end-of-file (EOF).

Action: Read to end-of-file or OPEN the file with the APPEND deviceparameter if you want to add to the file. If you need a fresh copy of the file, OPEN it with the NEWVERSION deviceparameter.

NOVALUE

NOVALUE, Qualifier xxxx does not take a value

GDE Error: This indicates that GDE encountered a value for a qualifier that does not accept a value. xxxx is the name of the qualifier.

Action: Specify the qualifier without a value.

NOZBRK

NOZBRK, No zbreak at that location

Run Time Information: This indicates that a ZBREAK command attempted to remove a ZBREAK from a line that did not specify one. Therefore, no action occurred.

Action: Review the current ZBREAKs using ZSHOW "B". All breaks can be removed using ZBREAK -*.

NOZTRAPINTRIG

NOZTRAPINTRIG, Use of \$ZTRAP in a database trigger environment (\$ZTLEVEL greater than 0) is not supported.

Run Time Trigger Error: GT.M requires the use of \$ETRAP for error handling within trigger logic.

Action: Modify the application code to use \$ETRAP to handle errors in trigger logic.

NULLCOLLDIFF

NULLCOLLDIFF, Null collation order cannot be different for all regions

Run Time Error: Standard null collation setting is not same for all regions.

Action: Using GDE show or DSE dump fileheader, check standard null collation field for all regions and make sure they are same.

NULSUBSC

NULSUBSC, Null subscripts are not allowed for region: xxxx

Run Time Error: This indicates that a global variable specified a null subscript in a database reference that mapped to region xxxx, which does not accept null subscripts.

Action: Look for the source of the null subscript(s). You can use GDE and/or DSE to remove the prohibition.

NUMOFLOW

NUMOFLOW, Numeric overflow

Compile/Run Time Error: This indicates that a numeric literal or a string evaluated to a numeric that exceeds the numeric range of GT.M.

Action: Look for the source of the large number.

NUMUNXEOR

NUMUNXEOR, xxxx unexpected end of record in numeric subscript

MUPIP Error: This indicates that LOAD aborted because it encountered an improperly formatted numeric subscript xxxx in its input stream.

Action: Refer to the topic MUPIP LOAD Errors in About This Manual section in this manual.

NUMPROCESSORS

NUMPROCESSORS, Could not determine number of processors

Run Time Warning: This indicates that the process was unable to determine the number of CPUs in the machine. (The subsequent message(s) give more detailed information.) This causes the number to default to one (1), which if incorrect may cause sub-optimal tuning.

Action: Analyze the accompanying message(s). If you require assistance, report the entire incident context to your GT.M support channel.

OBJDUP

OBJDUP, xxxx yyyy already exists

GDE Error: This indicates that an ADD command attempted to add a NAME, REGION, or SEGMENT xxxx, that already exists. yyyy is the NAME, REGION, or SEGMENT.

Action: Use the CHANGE command or specify a different object name.

OBJFILERR

OBJFILERR, Error with object file I/O on file xxxx

Run Time Error: This indicates that [auto]ZLINK processing encountered an error when it attempted to access object file-specification xxxx.

Action: Use host shell commands to examine the file and its protection.

OBJNOTADD

OBJNOTADD, Not adding xxxx

GDE Error: This indicates that GDE did not add the specified NAME, REGION, or SEGMENT. xxxx is the NAME, REGION, or SEGMENT specified with the ADD command.

Action: Review the accompanying message(s) for additional information.

OBJNOTCHG

OBJNOTCHG Not changing xxxx

GDE Error: This indicates that GDE has not changed the specified NAME, REGION, or SEGMENT. xxxx is the NAME, REGION, or SEGMENT specified with the CHANGE command.

Action: Review the accompanying message(s) for additional information.

OBJNOTFND

OBJNOTFND, xxxx does not exist

GDE Error: This indicates that a CHANGE or DELETE command specified a NAME, REGION, or SEGMENT that does not exist. xxxx is the NAME, REGION, or SEGMENT.

Action: Use the ADD command or look for and correct any typographical errors.

OBJREQD

OBJREQD, xxxx required

GDE Error: This indicates that an ADD, CHANGE, DELETE, RENAME, or TEMPLATE command does not specify a NAME, REGION, or SEGMENT. xxxx is the required object-type.

Action: Look for a missing space or supply the NAME, REGION, or SEGMENT.

OFFSETINV

OFFSETINV, Entry point xxxx+yyyy not valid

Compile/Run Time Error: This indicates that GT.M encountered a label xxxx and an offset yyyy that did not fall within the actual lines of the routine.

Action: Modify the routine so the entry point is a valid entryref.

OLDBINEXTRACT

OLDBINEXTRACT, Loading an older version (xxxx) of binary extract

Run Time Error: This indicates that a MUPIP LOAD input file is of an older type that may not properly deal with collations other than the default (standard M) collation.

Action: No action is required if collation is not an issue. If collation is an issue, the source of the EXTRACT should be upgraded and the MUPIP EXTRACT re-run before the LOAD.

ORLBKCMPLT

ORLBKCMPLT, ONLINE ROLLBACK completed successfully on instance iiiii corresponding to dddd

MUPIP Information: Issued by MUPIP ROLLBACK -ONLINE when it successfully completes work on database file dddd on instance iiiii.

Action: None required.

ORLBKFRZOVER

ORLBKFRZOVER, tttt : FREEZE on region rrrr (ddd) cleared

MUPIP Information: Issued by MUPIP ROLLBACK -ONLINE when it clears a FREEZE on region rrrr mapped to database file dddd; tttt is the time it cleared the FREEZE.

Action: None required.

ORLBKFRZPROG

ORLBKFRZPROG, tttt : waiting for FREEZE on region rrrr (dddd) to clear

MUPIP Information: Issued by MUPIP ROLLBACK -ONLINE when it encounters a region rrrr mapped to database file dddd which is frozen; tttt is the time it encountered the condition.

Action: ROLLBACK waits for a period determined by the gtm_db_startup_max_wait environment variable, after which it clears the FREEZE and proceeds. The ROLLBACK is inappropriate due to the conditions that lead to the FREEZE, cancel the ROLLBACK, otherwise cancel the FREEZE or wait for ROLLBACK to clear it automatically.

ORLBKNOSTP

ORLBKNOSTP, ONLINE ROLLBACK proceeding with database updates. MUPIP STOP will no longer be allowed

MUPIP Information: Issued by MUPIP ROLLBACK -ONLINE when it starts processing that cannot be interrupted without jeopardizing database integrity.

Action: Wait for the ROLLBACK to complete.

ORLBKNOV4BLK

ORLBKNOV4BLK, Region rrrr (dddd) has V4 format blocks. Database upgrade required. ONLINE ROLLBACK cannot continue

MUPIP Error: Issued by MUPIP ROLLBACK -ONLINE when it finds the region rrrr mapped to database file dddd contains V4 format blocks - online rollback does not support old format blocks.

Action: Upgrade the database to the current major version before attempting to use online rollback.

ORLBKSTART

ORLBKSTART, ONLINE ROLLBACK started on instance iiiii corresponding to dddd

MUPIP Information: Issued by MUPIP ROLLBACK -ONLINE when it starts work on database file dddd on instance iiiii.

Action: None required.

ORLBKTERMNTD

ORLBKTERMNTD, ONLINE ROLLBACK terminated on instance iiiii corresponding to dddd with the above errors

MUPIP Error: Issued by MUPIP ROLLBACK -ONLINE when it encounters issues that prevent it from operating on database file dddd on instance iiiii.

Action: Analyze and address the errors in the output preceding this message.

OMISERVHANG

OMISERVHANG, GTCM OMI server is hung

GT.CM Error: The GT.CM OMI server has gone a long time, exceeding the design expectation without acknowledged activity. At the point of this error, GT.M creates a core image of the GT.CM OMI server.

Action: Investigate the state of the server and its clients. Restart processes including the server as appropriate; refer diagnostic information to the the group responsible for database integrity at your operation.

OPCOMMITTED

OPCOMMITTED n errors and m MBFULLs sending prior operator messages

Informational Message: GT.M issues this message to the operator log if any operator messages prior to the immediately preceding one had not been sent due to errors from \$SNDOPR. m is the number of time a persistent MBFULL error prevented a messages from being sent and n is the number of other errors whose reports were bypassed.

Action: None.

OPENCONN

OPENCONN Error opening TCP connection

Run Time Error: This indicates that the process of opening a socket resulted in a device error.

Action: Review the accompanying message(s) for additional information.

OPRCCPSTOP

OPRCCPSTOP, The Cluster Control Program has been halted by an operator stop request

CCE Error: This indicates that a CCE STOP command halted access to clustered databases from this node.

Action: Contact the group responsible for databases at your site for information about when clustered operation will resume.

ORDER2

ORDER2, Invalid second argument to \$ORDER. Must be -1 or 1

Run Time Error: This indicates that the second argument to a \$ORDER function was not a 1 or -1, which are the values the standard permits.

Action: Modify the argument.

ORLBKCMPLT

ORLBKCMPLT, ONLINE ROLLBACK completed successfully on instance iiiii corresponding to dddd

MUPIP Information: Issued by MUPIP ROLLBACK -ONLINE when it successfully completes work on database file dddd on instance iiiii.

Action: None required.

ORLBKFRZOVER

ORLBKFRZOVER, tttt : FREEZE on region rrrr (ddd) cleared

MUPIP Information: Issued by MUPIP ROLLBACK -ONLINE when it clears a FREEZE on region rrrr mapped to database file dddd; tttt is the time it cleared the FREEZE.

Action: None required.

ORLBKFRZPROG

ORLBKFRZPROG, tttt : waiting for FREEZE on region rrrr (dddd) to clear

MUPIP Information: Issued by MUPIP ROLLBACK -ONLINE when it encounters a region rrrr mapped to database file dddd which is frozen; tttt is the time it encountered the condition.

Action: ROLLBACK waits for a period determined by the gtm_db_startup_max_wait environment variable, after which it clears the FREEZE and proceeds. The ROLLBACK is inappropriate due to the conditions that lead to the FREEZE, cancel the ROLLBACK, otherwise cancel the FREEZE or wait for ROLLBACK to clear it automatically.

ORLBKNOSTP

ORLBKNOSTP, ONLINE ROLLBACK proceeding with database updates. MUPIP STOP will no longer be allowed

MUPIP Information: Issued by MUPIP ROLLBACK -ONLINE when it starts processing that cannot be interrupted without jeopardizing database integrity.

Action: Wait for the ROLLBACK to complete.

ORLBKNOV4BLK

ORLBKNOV4BLK, Region rrrr (dddd) has V4 format blocks. Database upgrade required. ONLINE ROLLBACK cannot continue

MUPIP Error: Issued by MUPIP ROLLBACK -ONLINE when it finds the region rrrr mapped to database file dddd contains V4 format blocks - online rollback does not support old format blocks.

Action: Upgrade the database to the current major version before attempting to use online rollback.

ORLBKSTART

ORLBKSTART, ONLINE ROLLBACK started on instance iiii corresponding to dddd

MUPIP Information: Issued by MUPIP ROLLBACK -ONLINE when it starts work on database file dddd on instance iiii.

Action: None required.

ORLBKTERMNTD

ORLBKTERMNTD, ONLINE ROLLBACK terminated on instance iiii corresponding to dddd with the above errors

MUPIP Error: Issued by MUPIP ROLLBACK -ONLINE when it encounters issues that prevent it from operating on database file dddd on instance iiii.

Action: Analyze and address the errors in the output preceding this message.

OUTOFSPACE

OUTOFSPACE, Database file xxxx ran out of disk space. Detected by process aaaa. Exit without clearing shared memory due to the disk space constraints. Make space and then perform mupip rundown to ensure database integrity.

Run Time Fatal Error: This indicates that the specified database is full and cannot extend due to lack of disk space. The database could not properly run down.

Action: Examine the space management procedures and take actions to prevent any reoccurrence of this error. The database might get damaged if you do not make enough space for MUPIP RUNDOWN to succeed.

PADCHARINVALID

PADCHARINVALID, PAD deviceparameter cannot be greater than 127.

Runtime Error: The PAD deviceparameter (valid only for Sequential Disk files) specified in the open command can be between 0 and 127 (both inclusive).

Action: Specify a value within the allowed range.

PARBUFSM

PARBUFSM, Parse buffer too small

Run Time Error: This indicates that an attempt to parse a file-specification exceeded the maximum length for file-specifications.

Action: Review the file-specification for valid syntax; if it is a logical name / environment variable, confirm its definition.

PARFILSPC

PARFILSPC, Parameter: xxxx file specification: yyyy

Run Time Error: This indicates that a JOB command jobparameter xxxx specified an invalid file-specification yyyy.

Action: Review the file-specification for valid syntax based on the operating system.

PARNORMAL

PARNORMAL, Parse successful

Run Time Information: This indicates that the parse was completed successfully.

PATCLASS

PATCLASS, Illegal character class for pattern code

Compile Time Error: This indicates that a pattern match specified an invalid pattern class code.

Action: Look for a code that is not supported by the current code definitions or for a literal match that is not enclosed in quotes.

PATCODE

PATCODE, Illegal syntax for pattern

Compile Time Error: This indicates that a pattern match specified an invalid syntax.

Action: Look for a missing repeat count.

PATLIT

PATLIT, Illegal character or unbalanced quotes for pattern literal

Compile Time Error: This indicates that a pattern match included the start of a string literal that did not finish properly.

Action: Look for unbalanced quotes in the string literal.

PATLOAD

PATLOAD, Error loading pattern file xxxx

Compile Time Error: This indicates that GT.M failed to load the pattern file.

Action: Review accompanying messages for additional information about the cause of this error.

PATMAXLEN

PATMAXLEN, Pattern code exceeds maximum length

Compile Time Error: This indicates that a pattern match specification required more temporary storage than is available.

Action: Modify the routine so it uses shorter pattern specifications.

PATNOTFOUND

PATNOTFOUND, Current pattern table has no characters with pattern code xxxx

Run Time Error: This indicates that the specified pattern code does not exist in the pattern table.

Action: Update the pattern table with the code, or change the program to make sure the specified pattern code is not referenced in the table.

PATTABNOTFND

PATTABNOTFND, Pattern table xxxx not found

Run Time Error: This indicates that an attempt to load a pattern table failed because it was not found in the file described by the logical name / environment variable gtm_pattern_file or loaded by the VIEW "PATLOAD" command.

Action: Use host shell commands to examine the file and modify either the file or the VIEW command that performs the load.

PATTABSNTAX

PATTABSyntax, Error in xxxx at line yyyy

Compile/Run Time Error: This indicates that GT.M found an error on line yyyy of the file xxxx that defines the patterns to be used by the pattern match operator.

Action: Modify the pattern match file and reload it. For more information, refer to the "Internationalization" chapter in the Programmer's Guide.

PATUPPERLIM

PATUPPERLIM, Pattern code upper limit is less than lower limit

Compile Time Error: This indicates that a pattern match specified a repeat count range with an upper limit that is below the lower limit.

Action: Look for improperly ordered repeat count ranges.

PCONDEXPECTED

PCONDEXPECTED, Post-conditional expression expected but not found

Compile Time Error: This indicates that a colon (:) appeared to start a postconditional but it was not followed by a valid postconditional expression.

Action: Look for unwanted colons or missing post-conditional expressions.

PERMGENDIAG

PERMGENDIAG, Permissions: Proc(uid:uuuu,gid:gggg), DB File(uid:vvvv,gid:hhhh,perm:pppp), Lib File(gid:iiii,perm:qqqq), Group Mem(opener:jjjj,owner:kkkk)

Run Time Information: This shows the permissions involved in a resource creation for the process, the associated database file, the libgtmshr and the process group membership.

Action: Typically none, but if you have a permission issue use this key information for diagnosis.

PERMGENFAIL

PERMGENFAIL, Failed to determine access permissions to use for creation of xxxx for file yyyy

MUPIP/Run Time Error: This message indicates that GT.M was unable to determine the permissions to use when creating a file or resource associated with database file yyyy. xxxx may be "ipc resources", "journal file", "backup file", or "snapshot file".

Action: Note the user and group ownership of the database file and \$gtm_exe/libgtmshr.*, and the user and group permissions of the GT.M process, and report them to your GT.M support channel.

PRCNAMLEN

PRCNAMLEN, Process name xxxx length is greater than yyyy

Run Time Error: This indicates that a JOB command PROCESS_NAME=xxxx jobparameter specified a value that exceeds the maximum acceptable length yyyy.

Action: Modify the process name so that it does not exceed yyyy characters.

PREFIXBAD

PREFIXBAD, xxxx must start with an alphabetic character to be a yyyy

GDE Error: This indicates that an ADD, CHANGE, DELETE, RENAME, or TEMPLATE command specified a REGION or SEGMENT name that does not begin with an alphabetical character. xxxx is the REGION or SEGMENT name. yyyy is the object-type.

Action: Look for and correct typographical errors.

PREMATEOF

PREMATEOF, Premature end of file detected

MUPIP Error: LOAD detected an end-of-file when it was expecting additional records.

Action: Refer to the About this Manual section on MUPIP LOAD errors earlier in this manual.

PREVJNLLINKCUT

PREVJNLLINKCUT, Previous journal file name link set to NULL in new journal file xxxx created for database file yyyy

Run Time/MUPIP Error: This indicates that GT.M or MUPIP has removed the link of previous journal file name and set it to NULL in the new xxxx journal files header. This could possibly be because journal state was ON for the database file yyyy and its corresponding journal file was inaccessible, this triggered MUPIP or GT.M to create new journal file xxxx clearing the previous generation journal file name(s).

Action: If the error is issued by GT.M review the accompanying message(s) in the operator log.

If a MUPIP SET -JOURNAL=ON command produces this message for the region in the operator log, it may indicate that one or more of the current generation journal files are damaged/missing and new journal files were created with no back pointers to the previous journal files. FIS recommends taking a database backup at the earliest convenience because a MUPIP RECOVER/ROLLBACK will not be able to go back past xxxx. If this message is for a specified region(s), consider switching the journal files for all regions (with REGION "**") that the process has opened (all journaled/replicated regions in the instance if replication is in use) to ensure that the RECOVER/ROLLBACK for other regions remains unaffected.

No action is required if the MUPIP BACKUP -NEWJNLFILES=NOPREVLINK issues the error.

PREVJNLLINKSET

PREVJNLLINKSET, Previous generation journal file name is changed from xxxx to yyyy

MUPIP Information: This indicates that MUPIP SET -JNLFILE command has changed previous generation journal file name from xxxx to yyyy.

PREVJNLNOEOF

PREVJNLNOEOF, A previous generation journal file xxxx does not have valid EOF

MUPIP Error: This indicates that while opening the previous generation journal file xxxx MUPIP encountered the journal file in an inconsistent state; it had not been terminated properly.

Action: Report the entire incident context to your GT.M support channel.

PRIMARYISROOT

PRIMARYISROOT, Attempted operation not valid on root primary instance xxxx

MUPIP Error: If a replication instance has local updates enabled, that is: the Source Server that created the journal pool was started with -UPDOK, issuing any Source Server command with the start, activate or deactivate qualifiers where the command explicitly specifies the propagateprimary qualifier or that qualifier is implicitly assumed by default, or, if this is a not a supplementary instance, attempting to start a Receiver Server causes MUPIP to issue this error.

Action: Do not start a Receiver Server on a root primary non-supplementary instance. Use rootprimary qualifier instead of propagateprimary in the source server command.

PRIMARYNOTROOT

PRIMARYNOTROOT, Attempted operation not valid on non-root primary instance xxxx

MUPIP Error: If a replication instance is not a root primary (the journal pool already exists and was created by a source server command that specified propagateprimary), issuing a source server command with the start or deactivate qualifiers that has the rootprimary qualifier explicitly specified (or implicitly assumed) on this instance will cause this error to be issued. This error can also be issued by the receiver server or mupip rollback if the instance that the source server is running on is not a root primary and it connects to a receiver server or a mupip journal -rollback -fetchresync running on an instance that was formerly a root primary and has not yet had a mupip replic -source -losttncomplete command run either explicitly or implicitly on it.

Action: Use propagateprimary qualifier instead of rootprimary in the source server command. If this error is issued by the receiver server or fetchresync rollback, the secondary instance has to be brought up as the secondary of a root primary since it was a root primary immediately before this. The rule is that any instance that was previously a root primary should be brought up as a secondary of the new root primary. This will create a lost transaction file that needs to be applied on the new root primary. Once that is done, a mupip replic -source -losttncomplete command should be run either explicitly or implicitly on this instance before trying to bring this up as a secondary of a propagating primary.

PROCTERM

PROCTERM, uuuu process termination due to cccc from eeee

Utility Warning: A utility uuuu, typically MUPIP, executing application code, possibly from a trigger, encountered a command cccc to terminate at \$zposition location pppp.

Action: It is not typically wholesome for MUPIP to terminate this way - review your error handling and trigger definitions for a possible bug or misfeature.

PROTNOTSUP

PROTNOTSUP, Protocol xxxx not supported

Run Time Error: This indicates that the protocol specified on the CONNECT or LISTEN deviceparameters is not currently supported.

Action: Use TCP/IP protocol, by specifying TCP for the protocol string.

QUALBAD

QUALBAD, xxxx is not a valid qualifier

GDE Error: This indicates that GDE encountered a command with an invalid qualifier xxxx.

Action: Look for and correct typographical errors in the qualifier.

QUALDUP

QUALDUP, xxxx qualifier appears more than once in the list

GDE Error: This indicates that GDE encountered the qualifier xxxx more than once in the command.

Action: Specify the qualifier only once in the list.

QUALEXP

QUALEXP, Qualifier expected but not found

Run Time Error: This indicates that a \$ZROUTINES function did not encounter a qualifier, which is the next valid syntax element.

Action: In OpenVMS, look for a missing / (slash) preceding SRC or NOSRC. In UNIX, look for a missing right parenthesis or extra left parenthesis in a source directory specification.

QUALREQD

QUALREQD, xxxx required

GDE Error: This indicates that a command was missing the required xxxx qualifier.

Action: Enter the missing qualifier.

QUALVAL

QUALVAL, Qualifier value required but not found

Run Time Error: This indicates that a \$ZROUTINES function did not specify a value for the SRC qualifier.

Action: The SRC qualifier requires a value.

QUITALSINV

QUITALSINV, QUIT * return when the extrinsic was not invoked with SET *

Run Time Error: A [sub-]routine tried to pass an alias back to the caller, but the routine was not invoked to accept an alias return.

Action: Rework either the invocation or the return, or troubleshoot why the inappropriate invocation occurred. If the routine should conditionally return an alias, use \$QUIT to select the proper type of return.

QUITARGLST

QUITARGLST, Quit cannot take a list of arguments

Compile Time Error: This indicates that a QUIT specified multiple arguments; M accepts only one argument.

Action: Look for a missing space after the QUIT or a typographical error. Modify the QUIT argument so that it consists of a single expression.

QUITARGREQD

QUITARGREQD, Quit from an extrinsic must have an argument

Run Time Error: This indicates that a QUIT did not specify an argument but it corresponded to an invocation by an extrinsic function or special variable.

Action: Review the interface between the extrinsic and the invoked routine. Modify the QUIT or the invocation.

QUITARGUSE

QUITARGUSE, Quit cannot take an argument in this context

Compile Time Error: This indicates that a QUIT in the scope of a FOR command specified an argument.

Action: Look for a missing space after the QUIT.

RANDARGNEG

RANDARGNEG, Random number generator argument must be greater than or equal to one

Run Time Error: This indicates that a \$RANDOM function specified a zero or a negative argument.

Action: Look for the source of the argument. If you want to generate a 0 or 1 result, the argument should be 2 because a seed of 1 always produces the less-than-random result of 0.

RBWRNNOTCHG

RBWRNNOTCHG, Not all specified databases were changed

MUPIP Warning: This indicates that the MUPIP SET command did not include a value for the RESERVED_BYTES qualifier, or, the value specified was out of permissible range. Therefore, it did not modify the specified database.

Action: Verify the RESERVED_BYTES qualifier specification.

RCVR2MANY

RCVR2MANY, The instance already has the maximum supportable number of receiver servers nnnn active

MUPIP Error: Issued by a Receiver Server on a Supplementary Instance (started with -UPDOK) which found it exceeds nnnn, the currently permitted number of Receiver Servers.

Action: Reconfigure the instances to a supported configuration.

RCVRMANYSTRMS

RCVRMANYSTRMS, Receiver server now connecting to source stream NNNN but had previously connected to a different stream nnnnMUPIP Error: Issued by a Receiver Server on a Supplementary Instance (started with -UPDOK) which had formerly connected to a source server corresponding to non-Supplementary stream nnnn, later disconnected and on reconnection found the Source Server corresponds to a different non-Supplementary stream NNNN.

Action: Mixing of non-Supplementary streams are not allowed in the same Receiver Server process. Restart the Receiver Server.

RDFLTOOLONG

RDFLTOOLONG, Length specified for fixed length read exceeds the maximum string size

Run Time Error: The size specified in the fixed length READ is too large. It should not exceed 1048576, which is the maximum string size supported by GT.M.

Action: Modify the READ to specify a length less than or equal to 1048576.

RDFLTOOSHORT

RDFLTOOSHORT, Length specified for fixed length read less than or equal to zero

Run Time Error: This indicates that a READ fixed length (#) specified a value of less than one.

Action: Change the length (i.e., the portion of the READ argument that appears after the delimiter (#)) to a valid value, or add a postconditional to the READ command to suppress the length when it is less than or equal to zero.

REC2BIG

REC2BIG, Record size (xxxx) is greater than maximum (yyyy) for region: zzzz

Run Time Error: This indicates that a SET attempted to create a database node with a combined length of keys and data (xxxx) that exceeds the maximum length yyyy permitted for region zzzz.

Action: Use smaller data records or keys in the program. If you want to enlarge the record size for the region, use GDE to change the Global Directory and recreate the database with MUPIP CREATE. If it is necessary to permit the data without allowing time to

rebuild the database, use DSE CHANGE with the FILEHEADER and RECORD-MAX-SIZE qualifiers. Be careful when you increase the size for existing databases; use GDE to ensure that they have proper characteristics the next time they are CREATED.

RECCNT

RECCNT, Last LOAD record number: xxxx

MUPIP Information: EXTRACT and LOAD use this message to display xxxx, the total number of records processed.

RECNOCREJNL

RECNOCREJNL, Recover could not create new journal file xxxx.

Run Time Error: This message is generated when recover/rollback fails to create the temporary forw_phase journal file. The database is restored to a consistent state with some earlier time, but recovery to the specific time could not be performed.

Action: Review the accompanying messages for additional information. Refer to the *Maintaining Database Integrity* chapter in the Administration and Operations Guide.

RECORDSTAT

RECORDSTAT, gggg: Key cnt: kkkk max subsc len: ssss max data len: dddd max rec len: rrrr

MUPIP information: LOAD and EXTRACT use this to report on some characteristics of the global variables they processed, where gggg is the unsubscripted global name, kkkk is the number of unique data cells in the array, ssss is the maximum subscripted key length, dddd is the maximum data length and rrrr is the maximum combined length of keys and subscripts.

Actions: Use the information as appropriate.

RECSIZENOTEVEN

RECSIZENOTEVEN, RECORDSIZE [xxxx] needs to be a multiple of 2 if ICHSET or OCHSET is UTF-16, UTF-16LE or UTF-16BE

Runtime Error: This error is issued when the OPEN command specifies an ICHSET or OCHSET or CHSET of UTF-16 or UTF-16LE or UTF-16BE and the RECORDSIZE specified (xxxx) is not a multiple of 2.

Action: Specify a RECORDSIZE that is a multiple of 2.

RECSIZIS

RECSIZIS, Record size is xxxx

GDE Information: This message displays the record size of the REGION with which you are working.

Action: Review the accompanying message(s) for additional information.

RECTOOBIG

RECTOOBIG, Block size xxxx and yyyy reserved bytes limit record size to zzzz

Error Messages

GDE Warning: This indicates that an ADD, CHANGE, or TEMPLATE command specified a value for the qualifier RECORDSIZE that is incompatible with the value of xxxx specified for BLOCKSIZE. zzzz is the maximum RECORDSIZE supported by this BLOCKSIZE, and yyyy RESERVED_BYTES for the block. GDE displays this message with other error messages, including one that reports the specified record size.

Action: Modify the RECORDSIZE, BLOCKSIZE and/or RESERVED_BYTES, so they are compatible.

RECVPOOLSETUP

RECVPOOLSETUP, Receive Pool setup error

Run Time Error: This indicates that an error occurred in the replication subsystem while initializing the receive pool.

Action: Verify that the receiver server has been configured correctly. See accompanying messages for more information about the cause of this error.

REGIS

REGIS, in region xxxx

GDE Information: This message displays the name of the REGION with which you are working.

Action: Review the accompanying message(s) for additional information.

REGNTFND

REGNTFND, Region referenced not initialized

GT.CM Server Error: This indicates that there has been a region management error. The region may not be present, or there may have been an error during initialization.

Action: Record any accompanying messages and if necessary report the entire incident context to your GT.M support channel.

REGSSFAIL

REGSSFAIL, Process pppp encountered error contributing to the snapshot for region rrrr - the snapshot is no longer valid.

Mupip error: A GT.M process encountered failure while opening snapshot file or attaching to shared memory or writing a block to the snapshot file, any of which invalidate the snapshot file. The original error should be in the operator log.

Action: Examine the operator log for messages issued by process pppp to obtain details of the failure and take action, possibly by modifying file access characteristics or user roles, to address the problem.

RENAMEFAIL

RENAMEFAIL, Rename of file xxxx to yyyy failed

MUPIP Warning: This indicates that MUPIP failed in its attempt to rename the existing file xxxx to yyyy, before creating the new xxxx file.

Action: Check the accompanying message(s) for additional information.

REORGCTRLY

REORGCTRLY, User interrupt encountered during database reorg -- halting

MUPIP Information: This indicates that a REORG was interrupted. The reorganization is incomplete but the database is intact with no loss of data.

Action: Initiate REORG with RESUME qualifier, if appropriate.

REORGINC

REORGINC, Reorg was incomplete. Not all globals were reorged.

MUPIP Warning: This indicates that MUPIP did not reorg all the globals because of some resource constraint errors.

Action: Review the accompanying message(s) for more information.

REPL2OLD

REPL2OLD, Instance IIII uses a GT.M version that does not support connection with the current version on iiiii.

MUPIP Error: Issued by a Source Server, Receiver Server or MUPIP JOURNAL -ROLLBACK -FETCHRESYNC on Instance iiiii attempted to connect to instance IIII, but found IIII is running an earlier version that does not support the current replication protocol. This can indicate either that the older version is just too old for any connection with the newer version (in case the older version is less than V5.1-000) or that the older version doesn't have the logic required to support a Supplementary Instance (in case the older version is less than V5.5-000). Note that IIII may not be available if the older instance uses a version of GT.M less than V5.1-000.

Action: Upgrade the GT.M version on IIII to a version that can support communication with the current version or, if this is a Supplementary Instance, that can deal with a Supplementary Instance or choose another appropriate instance for the connection.

REPLACCSEM

REPLACCSEM, Error with replication access semaphore (id = xxxx) for instance file aaaa

MUPIP Error: This indicates problem with the semaphore xxxx associated with the instance designated by aaaa.

Action: Review the accompanying message(s) for details.

REPLBRKNTRANS

REPLBRKNTRANS, Replication subsystem found transaction xxxx broken or missing in the journal files

MUPIP Error: This indicates that while attempting to read the transaction with journal sequence number xxxx from journal files, the source server could not find all (or any) journal records belonging to that transaction.

Action: Restore the journal generation links, and/or the journal files. Deactivate and activate the source server (or shutdown and restart the source server). If the journal files that are needed are no longer available, follow the procedure Restoring secondary from backup of Primary detailed in the *Replication* chapter of Administration and Operations Guide.

REPLCOMM

REPLCOMM, Replication subsystem communication failure

MUPIP Error: This is a generic error indicating that there has been a communication error between the two systems performing replication.

Action: Review the accompanying message(s) for more information about the cause of this error.

REPLERR

REPLERR, XXXX

MUPIP Warning: This indicates that GT.M is performing tasks that may result in an error.

Action: Review accompanying messages for more information about why GT.M generated this message.

REPLEXITERR

REPLEXITERR, Replication process encountered an error while exiting

Run Time Error: This indicates that the source, receiver or update process encountered an error during exit processing.

Action: Review accompanying message(s) for more information.

REPLFILIOERR

REPLFILIOERR, Replication subsystem file I/O error xxxx

MUPIP Error: This indicates that the system was unable to perform an I/O operation on a file on the replication primary server. The accompanying message also tells whether there was a read error or a write error and names the file on which the error occurred.

Action: Review the accompanying message(s) for more information about the cause of this error.

REPLFILTER

REPLFILTER, Replication filter subsystem failure

MUPIP Error: This indicates that the replication filter subsystem failed to start the application filter.

Action: In order to restart the filter subsystem, it is necessary to shut down the replication server and restart. If this error continues to appear, report the entire incident context to your GT.M support channel.

REPLFTOKSEM

REPLFTOKSEM, Error with replication semaphores for instance file xxxx

MUPIP Error: This indicates that MUPIP could not create semaphore for replication instance file.

Action: Review the accompanying message(s) for more information.

REPLGBL2LONG

REPLGBL2LONG, Global names longer than 8 characters cannot be handled at the secondary

Source Server Error: A journal record contains a global variable name longer than 8 characters and the version running at the secondary does not support global variables longer than 8 characters.

Action: Upgrade the secondary to V5.0-FT01 or later. If upgrade is not possible, avoid replicating regions that contain globals longer than 8 characters.

REPLINFO

REPLINFO, xxxx

Run Time Information: The information is contained in the message.

REPLINSTACC

REPLINSTACC, Error accessing replication instance file xxxx

Run Time/MUPIP Error: This indicates that some internal GT.M errors were encountered while accessing the specified replication instance file defined by \$gtm_repl_instance.

Action: Refer to the accompanying message(s) for additional information.

REPLINSTDBSTRM

REPLINSTDBSTRM, Replication instance file rrrr has seqno xxxx for Stream nnnn while database has a different seqno XXXX

MUPIP Error: Issued by the first source server started on a supplementary instance if the journal stream sequence numbers (for any non-supplementary stream from 0 through 15) stored in the instance file do not match those stored in the database file headers. This is possible if a database was recreated or refreshed from a backup on another instance without correspondingly recreating the instance file.

Action: If the database file is known to be accurate, recreate the instance file. If not, reinitialize this instance from a backup of some other instance in the same LMS Group (see Action section of REPLINSTDBMATCH error for more details on this).

REPLINSTMISMATCH

REPLINSTMISMATCH, Process has replication instance file ffff (jnlpool shmid = ssss) open but database dddd is bound to instance file gggg (jnlpool shmid =tttt)

Run Time Error: An update is being attempted on the replicated database dddd which is associated with the replication instance file ffff and journal pool shared memory id ssss but the currently updating process has a different replication instance file gggg or journal pool shmid tttt open.

Action: A replicated database can only be updated by processes that have the same replication instance file (defined by the environment variable gtm_repl_instance) open. Ensure the environment variable gtm_repl_instance is defined to be the same for all processes that update the same replicated database file. This error is also possible if the replication instance file was recreated (while

processes were still accessing the replication instance). In this case, the name ffff and gggg would be the same but the corresponding journal pool shared memory ids would be different. To recover from this situation, shut down all processes accessing the instance from before and after the instance file recreate. Run an argumentless mupip rundown to clean up the older journal pool tttt and restart the instance. One more case of this error is when gggg is the empty string and tttt is 4294967295 (i.e. $2^{32} - 1$). In this case, the Source Server was started with a global directory that (a) did not have this database file listed and hence this database was not associated with the journal pool that the Source Server created OR (b) had this database file listed but later the source server and this particular database file was shut down while other database files in this global directory were still active. To recover from (a), add this database file to the global directory used by the Source Server and restart the instance. The Source Server (which is the first process to start on a replicated instance) only binds replicated databases from its global directory to the journal pool that it creates. No other replicated database file can be bound with this journal pool. To recover from (b), restart the Source Server.

REPLINSTUNDEF

REPLINSTUNDEF, Replication instance environment variable \$gtm_repl_instance is undefined

Run Time/MUPIP Error: This indicates that the replication instance environment variable \$gtm_repl_instance is undefined.

Action: Define the environment variable to the appropriate instance file.

REPLINSTCREATE

REPLINSTCREATE, Error creating replication instance file xxxx

Runtime Error: There was an error when GT.M or MUPIP tried to create the replication instance file. The error detail accompanies this message.

Action: Look at the accompanying message that gives error details. Possible causes are file permissions, system quotas, and so on. If possible, correct the cause and retry creating the replication instance file. If the error persists, report to your GT.M support channel along with the error details.

REPLINSTDBMATCH

REPLINSTDBMATCH, Replication instance file xxxx has seqno xxxx while database has a different seqno yyyy

MUPIP Error: This error is issued by the first source server that is started on a replication instance or a mupip journal -rollback command if the journal sequence numbers stored in the instance file does not match that stored in the database file header. This is possible if the database was recreated or refreshed from a backup on another instance without correspondingly recreating the instance file.

Action: If this instance is not the root primary, this error can be handled by restoring both the database and the instance file from a previous backup (consistent backup of the instance file AND database files taken together at the same time) and restarting the instance. Subsequent to such a restore, all transactions since the last backup will be sent across from this instance's primary. Alternatively, this can be handled by shipping a copy of the database from any other instance (either the primary or any other secondary/tertiary), recreating the instance file and starting this instance as a secondary with the -updateresync qualifier. In either case, this procedure has to be repeated on all tertiary instances etc. that descend from this instance ensuring that for every primary-secondary instance pair, the secondary is not ahead of the primary in terms of journal sequence number. If this instance is the root primary, restoring from a prior backup may not be viable as it may mean loss of transactions that occurred after the backup. The alternative way to handle this error is to recreate the instance file on the root primary, ship a copy of the database from the primary and recreate instance files on ALL secondaries (tertiaries etc.) and restart the secondaries with the -updateresync qualifier. In addition, report the entire incident context to your GT.M support channel.

REPLINSTFMT

REPLINSTFMT, Format error encountered while reading replication instance file xxxx. Expected yyyy. Found zzzz.

Runtime Error: This error is issued by GT.M or MUPIP whenever it tries to open the replication instance file and finds that it was created with a format that the current version of GT.M cannot interpret. GT.M also produces this error when it encounters:

1. an instance file created on a different endian system or
2. an instance file created by a 32-bit (or 64-bit) version of GT.M that is different from the current 64-bit (or 32-bit) version of GT.M.

Action: Recreate the instance file using the mupip replic -instance_create command with the current version of GT.M.

 The REPLINSTCORRV message that was displayed in V5.0-000 has now been replaced by REPLINSTFMT

REPLINSTNMLEN

REPLINSTNMLEN, Replication instance name xxxx should be 1 to 15 characters long

MUPIP Error: This error is issued by the mupip replic instance_create command if the instance name was specified either through the name qualifier or through the environment variable gtm_repl_instname and if name was longer than 15 characters or was the empty string.

Action: Specify a valid instance name that is 1 to 15 characters long.

REPLINSTNMSAME

REPLINSTNMSAME, Primary and Secondary instances have the same replication instance name xxxx

MUPIP Error: This error is issued by any source server command where the -instsecondary qualifier specifies a secondary instance name that matches the name of the primary instance the command is started from.

Action: Two instances should never have the same name. Recreate the instance file on the secondary with a different name and restart the receiver server with the updateresync qualifier.

REPLINSTNMUNDEF

REPLINSTNMUNDEF, Replication instance name not defined

MUPIP Error: This error is issued by the mupip replic -instance_create command if the -name qualifier was not specified and if the environment variable gtm_repl_instname is not defined either.

Action: Specify the instance name using the -name qualifier.

REPLINSTCLOSE

REPLINSTCLOSE, Error closing replication instance file xxxx

Runtime Error: There was an error when GT.M or MUPIP tried to close the replication instance file. The error detail accompanies this message.

Action: Look at the accompanying error detail. Possible causes are file permissions, system quotas, etc. Fix the cause if possible. If not, report the entire incident context to your GT.M support channel for further analysis.

REPLINSTOPEN

REPLINSTOPEN, Error opening replication instance file xxxx

Runtime Error: There was an error when GT.M or MUPIP tried to open the replication instance file. The error detail accompanies this message.

Action: Look at the accompanying error detail. Possible causes are file permissions, system quotas, and so on. Fix the cause if possible. If not, report the entire incident context to your GT.M support channel for further analysis.

REPLINSTREAD

REPLINSTREAD, Error reading xxxx bytes at offset yyyy from replication instance file ffff

Runtime Error: There was an error when GT.M or MUPIP tried to read from the replication instance file. The error detail accompanies this message.

Action: Look at the accompanying error detail. Possible causes are file permissions, system quotas, etc. Fix the cause if possible. If not, report the entire incident context to your GT.M support channel for further analysis.

REPLINSTNOHIST

REPLINSTNOHIST History record for xxxx not found in replication instance file yyyy

MUPIP Error: The source server or receiver server issue this message as an error while mupip rollback issues this message as a warning when they scan the replication instance file looking for a history record corresponding to a journal sequence number that is lesser than the earliest sequence number or greater than the latest sequence number stored in the instance file. This means that the replication instance files on the primary and secondary have differing level of history detail (possible if the instance file was later recreated in one instance) and that it is no longer possible to determine the sync point (resync seqno) between the two instances.

Action: If mupip rollback issues this error, it truncates the replication instance file history. This means that if this instance is a secondary, it should be brought up with the `-updateresync` qualifier. If the source or receiver server issue this error, this error needs to be handled by ensuring the primary and secondary databases are in sync (by shipping a copy of the database from the primary to the secondary if not already done), recreating the instance file on the secondary (if not already done) and start the receiver server on the secondary with the `-updateresync` qualifier.

REPLINSTSECLN

REPLINSTSECLN Secondary replication instance name xxxx should be 1 to 15 characters long

MUPIP Error: This error is issued by any mupip replic `-source` command that specifies a secondary instance name. This error is issued if the secondary instance name was specified either through the `-instsecondary` qualifier or through the environment variable `gtm_repl_instsecondary` and if the name was longer than 15 characters or was the empty string.

Action: Specify a valid secondary instance name that is 1 to 15 characters long.

REPLINSTSECMTCH

REPLINSTSECMTCH Secondary replication instance name xxxx sent by receiver does not match yyyy specified at source server startup

MUPIP Error: This error is issued by a source server that connects to a receiver server on the secondary and finds that the secondary instance name sent by the receiver does not match the secondary instance name specified (INSTSECONDARY qualifier) when the source server was started. The source server terminates after issuing this error.

Action: Restart the source server with the correct -instsecondary qualifier value. Also make sure the instance name in the -instsecondary qualifier and the host/port information in the secondary qualifier of the source server startup command correspond to each other.

REPLINSTSECNONE

REPLINSTSECNONE, No information found for secondary instance xxxx in instance file yyyy

MUPIP Error: This error is issued by any mupip replic source command that specifies a replicating (secondary) instance name (except for the one which specifies -start) if no information on this name can be found in the instance file. This is possible if no Source Server was ever started since the initialization of this instance file for such a replicating instance.

Action: Make sure the replicating instance name is correct. If it is, make sure a Source Server for that replicating instance has been started at least once in the life of the instance file even if it is currently not up and running.

REPLINSTSECUNDF

REPLINSTSECUNDF Secondary replication instance name not defined

MUPIP Error: This error is issued by any mupip replic -source command that requires a secondary instance name to be specified. The source server commands that require this qualifier are those that have any of -activate, changelog, deactivate, needrestart, start, statslog or stopsourcefilter specified. The secondary name can be specified either through the INSTSECONDARY qualifier or through the environment variable gtm_repl_instsecondary. If neither of them is specified, this error is issued.

Action: Specify the secondary instance name using the INSTSECONDARY qualifier.

REPLINSTSEQORD

REPLINSTSEQORD ssss has seqno xxxx which is less than last record seqno yyyy in replication instance file zzzz

MUPIP Error: This error is issued in one of two scenarios. The instance file consists of a sequence of history records that should correspond to an increasing range of sequence numbers. They need to hence have their starting sequence number in increasing order. If an attempt is made to append a history record with a starting sequence number that is lesser than the last history record currently existing in the instance file, the source or receiver server issues this error. In this case ssss would be the string New history record. This error is also issued if at journal pool creation time, the source server notices that the instance file header has a value of the current seqno that is lesser than the starting seqno of the last history record in the instance file. In this case ssss would be the string Instance file header.

Action: If this instance is not the root primary, this error can be handled by restoring both the database and the instance file from a previous backup (consistent backup of the instance file AND database files taken together at the same time) and restarting the instance. Subsequent to such a restore, all transactions since the last backup will be sent across from this instance's primary.

Alternatively, this can be handled by shipping a copy of the database from any other instance (either the primary or any other secondary/tertiary), recreating the instance file and starting this instance as a secondary with the UPDATERESYNC qualifier. In either case, this procedure has to be repeated on all tertiary instances etc. that descend from this instance ensuring that for every primary-secondary instance pair, the secondary is not ahead of the primary in terms of journal seqno. If this instance is the root primary, restoring from a prior backup may not be viable as it may mean loss of transactions that occurred after the backup. The alternative way to handle this error is to recreate the instance file on the root primary, ship a copy of the database from the primary and recreate instance files on ALL secondaries (tertiaries etc.) and restart the secondaries with the UPDATERESYNC qualifier. In addition, report the entire incident to your GT.M support channel.

REPLINSTNDALN

REPLINSTNDALN Could not get exclusive access to replication instance file xxxx

MUPIP Error: This error is issued by MUPIP REPLIC INSTANCE_CREATE if it finds that the replication instance file it is attempting to create already exists and is being used (the journal pool for that instance exists) by GTM and/or MUPIP process(es).

Action: Shutdown all GTM and/or MUPIP processes that are using the replication instance file and reissue the command. If it fails even though you know for sure there is no other GT.M or MUPIP process accessing the replication instance file, delete the instance file and reissue the command.

REPLUPGRADEPRI

REPLUPGRADEPRI, Attempted operation requires primary instance xxxx to support multi-site replication

Obsolete Error: No longer issued by GT.M

REPLUPGRADESEC

REPLUPGRADEPRI, Attempted operation requires primary instance xxxx to support multi-site replication

Obsolete Error: No longer issued by GT.M

REPLINSTWRITE

REPLINSTWRITE Error writing xxxx bytes at offset yyyy from replication instance file ffff

Runtime Error: There was an error when GT.M or MUPIP tried to write to the replication instance file. The error detail accompanies this message.

Action: Look at the accompanying error detail. Possible causes are file permissions, system quotas, etc. Fix the cause if possible. If not, report the entire incident context to your GT.M support channel.

REPLJNLCLOSED

REPLJNLCLOSED, Replication in jeopardy as journaling for database file ddd. Current region seqno is xxx[XXX] and system seqno is yyy[YYY]

GT.M run-time warning: This message indicates that GT.M turned OFF journaling and switched replication from ON to WAS_ON on the specified database. Other preceding messages identify the cause (for example, lack of disk space while writing to journal file,

permissions issue while auto-switching to new journal files, and so on). The message also displays the region and journal sequence numbers. From this point, replicating updates on the primary to the secondary might or might not work depending on the backlog on the primary until replication/journaling gets turned back ON.

Action:

1. First, correct the cause (lack of disk space, permission issues, and so on) that turned journaling OFF.
2. Execute the MUPIP SET REPLICATION=ON or MUPIP BACKUP REPLICATION=ON command to turn replication (and journaling) ON and switch to a new set of journal files. This command can work while processes are concurrently updating the database and causes GTM to journal subsequent updates in both the journal file and journal pool (rather than only in the journal pool as it does when replication is in the WAS_ON state).
3. Execute the MUPIP REPLIC -SOURCE -SHOWBACKLOG command. Note down the value of "sequence number of last transaction written to journal pool".
4. Execute the above command at regular intervals and note down the value of "sequence number of last transaction sent by source server."
5. If the "sequence number of last transaction sent by source server" is greater than "sequence number of last transaction written to journal pool", it means that the source server successfully sent all journal records during the time interval when journaling was turned OFF. In this case, no further action is required.
6. On the other hand, if the "sequence number of last transaction sent by source server" is less than "sequence number of last transaction written to journal pool" and reports the same value across repeated SHOWBACKLOG commands, then check the source server log file for any error messages - most likely a NOPREVLINK error from the source server. This means the source server could not locate the corresponding journal records required from the journal files to replicate a particular sequence number and therefore, it failed to synchronize the primary and secondary. In this case, take an online backup of the primary, restore it on the secondary and start the secondary with the UPDATERESYNC qualifier to synchronize the secondary with the primary.

REPLJNLCNFLCT

REPLJNLCNFLCT, Journaling cannot be turned nnnn on database file ffff as the replication state is rrrr and must also be turned nnnn in the same command

MUPIP Warning: This message indicates that the requested journaling state (nnnn) and current replication state (rrrr) do not match and the command must explicitly specify an outcome such that they do match.

Action: Issue one or more commands that leave journaling and replication either both ON or both OFF.

REPLLOGOPN

REPLLOGOPN, Replication subsystem could not open log file xxxx : yyyy. Logging done to zzzz

MUPIP Error: This indicates that GT.M could not find the log file or did not have access permission to open the log file. If there is another log file available (a previously opened file), GT.M writes to the other log file. If there is no other log file available, GT.M does not log.

Action: Check the log file permissions, and if permissions are correct, move the log file and specify that GT.M should log to the new location.

REPLNOBEFORE

REPLNOBEFORE, NOBEFORE option cannot be used when the current replication state is ON for a database file xxxx

MUPIP Warning: This indicates that GT.M could not use NOBEFORE journal option because replication is already turned ON for the database file xxxx.

Action: Use BEFORE option for the database file xxxx. NOBEFORE image journaling is not currently supported for replication. If NOBEFORE is necessary, use -replication=OFF option.

REPLNOMULTILINETRG

REPLNOMULTILINETRG, Sequence number ssss contains a trigger definition too large for transmission to the current replicating instance, which does not support multi-line triggers - stopping replication.

MUPIP Error: The source server logs this message whenever it detects replication to a V5.4-001 or V5.4-000A instance of a transaction that involves multi-line triggers spanning across multiple blocks in the database, which is possible if the multi-line trigger definition is large enough that it does not fit in a single database block.

Action: Upgrade the replicating instance to at least V5.4-002 or eliminate any usages of large multi-line triggers spanning across database blocks from the application logic.

REPLNOTON

REPLNOTON, Replication is not on for journal file xxxx, rollback will not continue

MUPIP Error: This indicates that ROLLBACK cannot proceed because MUPIP encountered xxxx, a journal file for which replication is not turned ON.

Action: ROLLBACK cannot be used for a journal file if it does not have replication state ON, use MUPIP JOURNAL RECOVER instead.

REPLNOXENDIAN

REPLNOXENDIAN, SSSS side is running on a GT.M version that does not support cross-endian replication. Upgrade the SSSS side to at least V5.3-003 to support cross-endian replication. Cannot continue.

MUPIP Error : The originating or the replicating instance in a cross-endian replication environment issue this error when a replication startup detects that the other side is running on a GT.M version without needed cross-endian support.

Action : Upgrade the instance for which the error was reported to a version with support for cross-endian replication.

REPLOFFJNLON

REPLOFFJNLON, Replication state for database file <xxx> is OFF but journaling state is enabled.

Replication Error: In a replicated environment, this indicates that the database file <xxx> cannot have journaling ENABLED or ON when the replication state is OFF. This is an out of design situation due to implications on recovery as journal files can't be a mix of SET/KILL records that were created when replication was ON and those created when replication was OFF.

Action: In order to prevent this situation, enable replication state for the database file <xxx> (using MUPIP SET -REPLICATION=ON or MUPIP BACKUP -REPLICATION=ON) or disable journaling using MUPIP SET -JOURNAL=DISABLE whichever is desirable.

REPLPOOLINST

REPLPOOLINST, Error with replication pool xxxx for instance file yyyy

MUPIP Error: This indicates that MUPIP encountered an error for the replication shared memory of shared memory xxxx.

Action: Refer to the accompanied message(s) for detailed information.

REPLRECFMT

REPLRECFMT, Replication journal record format error encountered

MUPIP Fatal: This indicates that a formatting error has been encountered by the replication source server for a journal record.

Action: Report the entire incident context along with any GT.M logs, dump, and/or core files created within the same timeframe to your GT.M support channel.

REPLREQROLLBACK

REPLREQROLLBACK, Replication instance file xxxx indicates abnormal shutdown. Run MUPIP JOURNAL ROLLBACK first.

MUPIP Error: This error is issued by MUPIP REPLIC SOURCE -START if it is about to create the journal pool and finds that the replication instance file header indicates the journal pool was not cleanly shutdown previously. This may cause the instance file not to correspond to the database and/or journals.

Action: Run MUPIP JOURNAL ROLLBACK to cleanup the instance file, database and journal files before starting a source server on this instance.

REPLREQRUNDOWN

REPLREQRUNDOWN, Error accessing replication instance xxxx. Must be rundown on cluster node yyyy.

Run Time/MUPIP Error: This indicates that GT.M could not open the specified replication instance file because it was not properly closed on the cluster node yyyy.

Action: Issue MUPIP RUNDOWN command on the cluster node. A GT.M process or, replication server on that node may have been terminated by a method other than MUPIP STOP. If MUPIP RUNDOWN with no parameters fails, try MUPIP RUNDOWN region * with an appropriate global directory.

REPLSTATE

REPLSTATE, Replication state for region/database file xxxx is now yyyy

MUPIP Information: This indicates that replication state for region/database file xxxx, is now yyyy.

REPLSTATEERR

REPLSTATEERR, Replication state cannot be changed to the specified value for database file <xxx>.

MUPIP BACKUP Error: This indicates that the specified change in the replication state cannot be done due to the reason described in a following GTM-E-TEXT message.

Action: If the message indicates "Standalone access required", try to enable the replication in the standalone mode using MUPIP SET REPLICATION. If the message suggests switching journal file, specify the backup qualifier NEWJNL in the command line.

REPLSTATEOFF

REPLSTATEOFF, ROLLBACK cannot proceed as database xxxx does not have replication ON

MUPIP Error: This indicates that ROLLBACK command cannot proceed because the specified database xxxx does not have replication state ON.

Action: Ensure replication is turned ON for a database, before executing the ROLLBACK command. Alternatively, if REPLICATION was not used on the database, use MUPIP JOURNAL RECOVER.

REPLTRANS2BIG

REPLTRANS2BIG, Transaction xxxx of size yyyy too large to be accommodated in the zzzz pool

MUPIP Error: This indicates that the size of the incoming transaction is larger than the specified receive pool.

Action: The receiver server must be shut down and restarted with a larger receive pool size or, if possible, break the file transmission into smaller files.

REPLUPGRADEPRI

REPLUPGRADEPRI, Attempted operation requires primary instance xxxx to support multi-site replication

MUPIP Error: This error is issued if an attempt is made to start an active source server or activate a passive source server on a propagating primary instance while the receiver server on that instance is connected to a primary that has not yet been upgraded to the multi-site version of GT.M. This error is also issued when the receiver server on a propagating primary finds that the primary it connects to does not support multi-site replication and that there is at least one active source server running on the instance at that time.

Action: An active source server cannot be running on a secondary instance at the same time that the receiver server on this instance is connected to a primary that does not support multi-site functionality. Upgrade the primary instance identified in the message to the version of GT.M that supports multi-site replication functionality and then start active source servers.

REPLUPGRADESEC

REPLUPGRADESEC, Attempted operation requires secondary instance xxxx to support multi-site replication

MUPIP Error: This error is issued in three cases. 1) If a source server is currently connected to a dual-site secondary (i.e. a secondary running on a version of GT.M that does not support multi-site functionality), starting additional source servers will issue this error. 2) If a source server finds more than one source server (active or passive) running on the same instance at the time it connects to a

dual-site secondary it will issue this error. 3) On a propagating primary instance, a source server that connects to a dual-site tertiary instance will issue a this error at connection time.

Action: Upgrade the secondary instance identified in the message to the version of GT.M that supports multi-site replication functionality and then start multiple source servers.

REPLWARN

REPLWARN, xxxx

MUPIP Warning: This indicates that GT.M is performing tasks that may result in an error.

Action: Review accompanying messages for more information about why GT.M generated this message.

REPLXENDIANFAIL

REPLXENDIANFAIL, SSSS side encountered error while doing endian conversion at journal sequence number JJJJ

Error: The originating or the replicating instance in a cross-endian replication environment report this error whenever they detect that the endian conversion failed.

Action: Restart replication - if the transmission caused the problem, it's probably intermittent. Perform a MUPIP JOURNAL - EXTRACT -DETAIL on the journal files and search for the sequence number JJJJ to look for anything different about that journal record. If the report is on the secondary, take a fresh backup on the originating instance requesting new journal files, refresh the replicating instance and restart replication.

REQ2RESUME

REQ2RESUME, Request to resume suspended processing received from process xxxx owned by userid yyyy

Run Time Information: The information is logged to the operator facility. This indicates that a suspended process received signal SIGCONT to resume processing. This happens when a process is suspended while holding a scarce resource in order to permit other processes to access the resource. On systems that do not support advanced signal information, xxxx and yyyy both are 0 (zero).

REQDVIEWPARM

REQDVIEWPARM, Required View parameter is missing

Run Time Error: This indicates that the failed program attempted to use the VIEW function without specifying a keyword argument or with an invalid keyword argument.

Action: Programmer's should consult the Programmer's Guide for the correct syntax.

REQRUNDOWN

REQRUNDOWN, Error accessing database xxxx. Must be rundown on cluster node yyyy.

Run Time Error: This indicates that GT.M could not open database file xxxx due to a prior improper shutdown on cluster node yyyy. A GT.M process on cluster node yyyy may have failed to attach a database memory segment or it was terminated by a method other than MUPIP STOP.

Action: If you are using before image journaling, perform a MUPIP JOURNAL RECOVER/ROLLBACK operation to address this issue. Otherwise, run the MUPIP RUNDOWN command from cluster node yyyy. If MUPIP RUNDOWN with no parameters does not work, specify the region name or file-specification with the qualifier REGION or FILE, respectively.

RESOLVESEQNO

RESOLVESEQNO, Resolving until sequence number dddd [0xxxxx]

MUPIP Information: This indicates MUPIP JOURNAL ROLLBACK expects to do backward processing until it reaches sequence number with hexadecimal value xxxx (decimal value dddd). This is usually the common sequence number agreed upon between the primary and secondary by a -FETCHRESYNC rollback or the sequence number specified in a -RESYNC rollback.

Action: No action required.

RESOLVESEQSTRM

RESOLVESEQSTRM, Resolving until stream sequence number Stream nnnn : Seqno dddd xxxx

MUPIP Information: This indicates MUPIP JOURNAL ROLLBACK expects to do backward processing until it reaches the stream sequence number whose hexadecimal value is xxxx (decimal value dddd). This is usually the common stream sequence number agreed upon between the primary and secondary by a -FETCHRESYNC rollback or the stream sequence number specified in a -RESYNC rollback where -RSYNC_STRM is also specified.

Action: No action required.

RESUMESTRMNUM

RESUMESTRMNUM, Error with stream number specified in RESUME qualifier

MUPIP Error: Issued by a Receiver Server as an accompanying message to a UPDSYNCINSTFILE error. The stream number nnnn can be any integer value from -1 through 15.

Action: No action required for this message. Action is required for the preceding UPDSYNCINSTFILE error.

REUSEINSTNAME

REUSEINSTNAME, Error with instance name specified in REUSE qualifier

MUPIP Error: Issued by a Receiver Server when started with the -REUSE qualifier in case of either inappropriate use of this qualifier or an inappropriate instance name specified as a value to this qualifier.

Action: An accompanying GTM-I-TEXT message describes the particular error situation. Take appropriate corrective action based on that.

RHMISSING

RHMISSING, Right-hand side of expression expected

Compile Time Error: This indicates that a binary operator did not specify a corresponding right-hand value.

Action: Look for missing or invalid expressions.

RLBKCONFIGBNDRY

RLBKCONFIGBNDRY, Rollback encountered journal records indicating current source **iiii** replaced old source **oooo**; cannot rollback past sequence number **ssss**

MUPIP Error: Issued by MUPIP JOURNAL -ROLLBACK indicating it found a change in the source configuration from instance **oooo** to **iiii** for stream **ssss** within the range of its attempt to process. Since it cannot dynamically reconfigure the source connections to deal with this, it stops.

Action: Investigate whether a shorter rollback that won't encounter this configuration change would be useful. If not, refresh this instance from a backup or use the `-RSYNC_STRM` qualifier if appropriate.

RLBKSTRMSEQ

RLBKSTRMSEQ, Stream journal seqno of the instance after rollback is Stream **nnnn** : Seqno **dddd** **xxxx**

MUPIP Information: On a Supplementary Instance, MUPIP JOURNAL -ROLLBACK issues this message for each stream (from 0 through 15) that has at least one update. This message indicates how many updates in each stream this Supplementary Instance has processed.

Action: No action required.

RLBKJNLNOBIMG

RLBKJNLNOBIMG, Journal file **jjjj** has NOBEFORE_IMAGE journaling.

MUPIP information: MUPIP JOURNAL ROLLBACK displays this informational message whenever it finds journal file **jjjj** with NOBEFORE_IMAGE journaling (DSE DUMP -FILE for the corresponding database reports "Journal Before imaging" as FALSE).

As there are no before-image records in this journal file, MUPIP JOURNAL ROLLBACK does not roll back the database. Instead, it only generates a lost-transaction file.

Action: No user action required except to confirm that this type of rollback is appropriate and expected.

RLBKNOBIMG

RLBKNOBIMG, ROLLBACK cannot proceed as database **dddd** has NOBEFORE_IMAGE journaling

MUPIP Error: Rollback relies on BEFORE_IMAGE journaling and **dddd** did not have it turned on, so no rollback is currently possible.

Action: Restore the database from a backup and use forward recovery or a replication resynchronization to recover the database state. Use BEFORE_IMAGE journaling for databases you wish to be able to rollback.

RLBKLOSTTNOONLY

RLBKLOSTTNOONLY, ROLLBACK will only create a lost transaction file (database and journal files will not be modified)

MUPIP information: MUPIP JOURNAL -ROLLBACK displays this informational message at startup if it finds at least one database region with NOBEFORE_IMAGE journaling. In such a case, MUPIP JOURNAL -ROLLBACK can only create broken and lost transaction files (if appropriate) but otherwise not modify the database, journal files, or replication instance files.

Action: No user action required except to confirm that this type of rollback is appropriate and expected.

RLBKJNSEQ

RLBKJNSEQ, Journal seqno of the instance after rollback is xxxx[yyyy]

MUPIP Information: This indicates that the journal sequence number of the instance after MUPIP JOURNAL ROLLBACK command is xxxx.

RMBIGSHARE

RMBIGSHARE, File with BIGRECORD specified may only be SHARED if READONLY

Run Time Error: An OPEN command specified BIGRECORD and SHARED without also specifying READONLY. BIGRECORD files may only be shared if all uses are READONLY.

Action: If the file will only be read, add READONLY to the OPEN. If the file is to be written, remove the SHARED.

RMNOBIGRECORD

RMNOBIGRECORD File record size requires BIGRECORD parameter

Run Time Error: The RECORDSIZE specified is larger than 32767. The BIGRECORD parameter must be specified before a RECORDSIZE larger than 32767.

Action: Modify the OPEN to specify BIGRECORD before RECORDSIZE.

RMSRDONLY ●

Last used version: **V5.4-002A**

RMSRDONLY, Cannot write to a read-only sequential file

Run Time Error: This indicates that a WRITE command attempted to access a read-only file.

Action: Look for a missing USE or improper deviceparameters on the OPEN.

RMWIDTHPOS

RMWIDTHPOS, File record size or width must be greater than zero

Run Time Error: This indicates that the WIDTH deviceparameter specified a negative argument.

Action: Modify the routine to ensure a positive WIDTH.

RMWIDTHTOOBIG

RMWIDTHTOOBIG, File record size too big

Run Time Error: The RECORDSIZE specified is too large. For disk files the maximum is one megabyte.

Action: Modify the OPEN to use a smaller RECORDSIZE.

RNDWNSEMFAIL

RNDWNSEMFAIL, Attempting to acquire gds_rundown semaphore when it is already owned

Run Time Error: This indicates a logic error in GT.M.

Action: Report the entire incident context to your GT.M support channel.

RNDWNSKIPCNT

RNDWNSKIPCNT, A total of nnnn process(es) skipped database rundown due to a concurrent ONLINE ROLLBACK

Run Time Information: nnnn processes disconnected from a one or more databases while online rollback was running and therefore relied on online rollback or other remaining processes to complete any database shutdown.

Action: None required typically. If there's cleanup required, use MUPIP RUNDOWN.

ROLLBKINTERRUPT

ROLLBKINTERRUPT, Database file xxxx indicates interrupted ROLLBACK. Reissue the MUPIP JOURNAL ROLLBACK command.

MUPIP Error: This indicates that ROLLBACK has been interrupted on xxxx database file. Though MUPIP JOURNAL RECOVER command is issued instead of MUPIP JOURNAL ROLLBACK, but a previous MUPIP JOURNAL ROLLBACK command was terminated abnormally. Note that, when ROLLBACK is interrupted, only ROLLBACK can be used to fix the interrupted operation.

Action: Reissue the MUPIP JOURNAL ROLLBACK command.

ROUTINEUNKNOWN

ROUTINEUNKNOWN, Routine could not be found

Run Time Error: This indicates that a command (such as DO or JOB) or a \$TEXT function referred to a routine that is not in the running image.

Action: Look for unresolved reference warnings on the LINK that created the image. This error occurs on a JOB command when the routine is not in the image or available for ZLINKing, or when an auto-ZLINK finds that the routine reference has been damaged by an incomplete LINK.

RPAREN

RPAREN, List must end with right parenthesis or continue with comma

GDE Error: This indicates that a qualifier that accepts a list of arguments had an improper list format.

Action: Modify the command so that the list is enclosed in parentheses () and separated by commas (,).

RPARENMISSING

RPARENMISSING, Right parenthesis expected

Compile Time Error: This indicates that an expression, function, or subscripted variable contained a left parenthesis and no matching right parenthesis.

Action: Look for and correct any typographical errors.

RPARENREQD

RPARENREQD, xxxx Right parenthesis expected

MUPIP Error: This indicates that LOAD failed because it encountered xxxx in its input stream when it expected to find a right parenthesis.

Action: Refer to the topic MUPIP LOAD Errors in About This Manual section in this manual.

RSVDBYTE2HIGH

RSVDBYTE2HIGH, Record size ssss is greater than the maximum allowed for region rrrr with Block size bbbb and Reserved bytes cccc

Run Time Error: The attempted database update would result in a record size that is greater than what is allowed by the current database block size and reserved byte setting.

Action: If the Reserved Bytes setting for the database region identified is non-zero, try reducing it to allow this update. Otherwise, modify the update to reduce the resulting record size.

RSYNCSTRMSUPPONLY

RSYNCSTRMSUPPONLY, RSYNC_STRM qualifier only supported for Supplementary Instances

MUPIP Error: Issued by MUPIP JOURNAL -ROLLBACK indicating the -RSYNC_STRM qualifier only applies to Supplementary Instances - Business Continuity instances require comprehensive synchronization.

Action: Reissue the command without the -RSYNC_STRM qualifier.

RSYNCSTRMVAL

RSYNCSTRMVAL, RSYNC_STRM qualifier can only take on a value from 0 to 15

MUPIP Error: Issued by a MUPIP JOURNAL -ROLLBACK -RESYNC command which also specifies -RSYNC_STRM with a stream number outside of the range of 0 through 15.

Action: Specify a stream number within the allowed range.

RTNNAME

RTNNAME, Routine name expected here

Compile Time Error: This indicates that an entry reference specified a circumflex without a valid routine name.

Action: Look for a missing routine name in commands such as DO, GOTO, and JOB.

RTSLOC

RTSLOC, At M source location xxxx

Run Time Information: GT.M uses this message to display the line, offset, and routine where it encountered a run-time error.

Action: Review the accompanying message(s) for additional information.

RUNPARAMERR

RUNPARAMERR, Error accessing parameter for run command

Run Time Error: This indicates that a MUMPS -RUN command had a missing or invalid argument.

Action: Ensure that a MUMPS -RUN has an argument that specifies a valid entryref ([label]^routinename).

RWARG

RWARG, This is not a legal argument for a READ command

Compile Time Error: This indicates that a READ command specified an invalid argument.

Action: Look for a non-alphanumeric character in the READ argument.

RWFORMAT

RWFORMAT, A valid format expression (!, #, or ?expr) expected here

Compile Time Error: This indicates that a READ or WRITE command specified a format with invalid trailing characters.

Action: Look for and correct any typographical errors.

SCNDDDBNOUPD

SCNDDDBNOUPD, Database updates not allowed on the secondary

Run Time Error: This indicates that updates on secondary are currently not allowed as they may lead to inconsistency between primary and secondary.

Action: If you need to do an implicit database update on the secondary, contact the group responsible for maintaining database integrity at your operation.

SECNODZTRIGINTP

SECNODZTRIGINTP, Sequence number ssss contains \$ZTRIGGER() updates made inside a transaction which the current replicator does not support. The replicator must be upgraded to at least V5.4-002 to support this type of transaction. Cannot continue

MUPIP Error: The originating instance encountered a \$ZTRIGGER() function within a transaction with sequence number ssss. However, the replicating instance is running a pre-V5.4-002 version which cannot handle \$ZTRIGGER() within a transaction.

Action: Upgrade the replicating instance to V5.4-002 or later. Alternatively adjust the application code to avoid using \$ZTRIGGER() within a transaction.

SECNOTSUPPLEMENTARY

SECNOTSUPPLEMENTARY, ssss is a **Supplementary Instance** and so cannot act as a source to non-Supplementary Instance **iiii**

MUPIP Error: Issued by a Source Server on a Supplementary Instance ssss attempted to connect to a Replicating Instance **iiii**, but found **iiii** is not configured as a Supplementary Instance.

Action: Reconfigure the instances to a supported configuration.

SECONDAHEAD

SECONDAHEAD, Secondary ahead of Primary: Secondary db possibly updated by process other than Update process. Do rollback first.

Run Time Error: The update process issues this error on finding the Secondary database containing more updates than the Primary.

Action: If you allow database updates on Secondary, no action is needed. If not, investigate the cause. Make sure the database on secondary and primary are consistent.

SEGIS

SEGIS, in xxxx segment yyyy

GDE or DSE Information: This message displays the name xxxx of the SEGMENT with which you are working.

Action: Review the accompanying message(s) for additional information.

SELECTFALSE

SELECTFALSE, No argument to \$SELECT was true

Run Time Error: This indicates that a \$SELECT function did not specify any truth value expressions that evaluated to true.

Action: Modify the \$SELECT(). The common technique is to end the selection list with -1:expr)-where expr is some default value and the integer constant 1 is always true.

SELECTSYNTAX

SELECTSYNTAX, Argument to xxxx clause is not valid

MUPIP Error: This indicates that EXTRACT encountered a qualifier with an invalid value and aborted.

Action: Review the proper syntax for EXTRACT with the qualifier SELECT. Refer to the Administration and Operations Guide or the online help for the MUPIP EXTRACT command.

SEMKEYINUSE

SEMKEYINUSE, Semaphore key xxxx is already in use (possibly by an older version)

Run Time/MUPIP Error: This indicates that GT.M failed to create a semaphore. One possible cause is that a semaphore of ID xxxx already exists in the system and the number of semaphores in that semaphore set is different than the semget has attempted to create. If a new GT.M version increases the number of semaphores in the semaphore set, collision with an older GT.M version can cause this error.

Action: Check accompanying message(s) for additional information. Use the semstat2 tool to find detailed information about the existing semaphores in the system.

SEMREMOVED

SEMREMOVED, Semaphore id xxxx removed from the system

MUPIP Information: This indicates that the orphaned semaphore with ID xxxx is removed; because it contained the signature of a GT.M semaphore, and no process was currently using it.

SEMWT2LONG

SEMWT2LONG, Process wwwww waited ssss second(s) for the llll lock for region rrrr, lock held by pid pppp

Run Time Error: This indicates that the process pppp appears to be holding the llll control semaphore for region rrrr for longer than GT.M expects.

Action: Analyze the behavior of process pppp, and terminate it if appropriate. This error may indicate that the system is under-configured for the workload.

SERVERERR

SERVERERR, Severe error on server: xxxx

GT.CM Server Error: This indicates that the GT.CM Server encountered a fatal error and terminated.

Action: Review subsequent message(s) for more information. If necessary, report the entire incident context to your GT.M support channel.

SETECODE

SETECODE, Non-empty value assigned to \$ECODE (user-defined error trap)

Run Time Error: This indicates that an error trap occurred because \$ECODE got altered to a non-null string in an M routine.

Action: Make sure that either \$ETRAP or \$ZTRAP is set to the valid value, if the error needs to be handled by GT.M.

SETINSETTRIGONLY

SETINSETTRIGONLY, ISV iiii can only be modified in a 'SET' type trigger

Run Time Error: Code invoked for a trigger other than SET (such as KILL or ZTRIGGER) attempted to modify Intrinsic Special Variable `iiii`, which applies only to a SET trigger context.

Action: Review the trigger definition and correct the types or the code to avoid the issue.

SETINTRIGONLY

SETINTRIGONLY, ISV `iiii` cannot be modified outside of the trigger environment

Run Time Trigger Error: The Intrinsic Special variable `iiii` can only be SET within the context of trigger logic (`$ZTLEVEL > 0`)

Action: Examine the application logic to determine whether code intended for use in a trigger context falls in an execution path outside of trigger logic. For code intended to execute both inside and outside triggers, use a postcondition that limits the SET to within a trigger.

SETREG2RESYNC

SETREG2RESYNC, Setting resync sequence number `xxxx` to region sequence number `yyyy` for database `zzzz`

MUPIP Information: This displays that resync sequence number `xxxx` is being set to region sequence number `yyyy` for database `zzzz`, because the journal file had crash field set and update was disabled.

SETSOCKOPTERR

SETSOCKOPTERR, Setting the socket attribute `xxxx` failed: (errno == `aaaa`) `yyyy`

Run Time Error: This indicates that an attempt to modify the `xxxx` socket attribute failed for the reason described by `yyyy`.

Action: Review the message(s) and take appropriate action.

SETZDIR

SETZDIR, Cannot change working directory to `xxxx`.

Run Time Error: This indicates that there is an invalid directory specified in the SET `$ZDIR=<xxxx>` command. The accompanying message indicates the exact cause of the failure.

Action: Make sure the specified argument conforms to the syntax of a directory specification on the host operating system. Check for the existence of the directory and access control permissions associated with the directory.

SHMPLRECOV

SHMPLRECOV, Shared memory pool block recovery invoked for region `xxxx`

Run time Information: GT.M carves out a portion of shared memory/global section allocated for each database region to use for ONLINE BACKUP - this portion is called "shared memory pool". This portion is also used by GT.M on OpenVMS while the region is being downgraded dynamically. In the unlikely event of corruption of shared memory pool, or if the blocks are "lost" due to stopped/killed or failed processes, GT.M detects the corruption or lost blocks and runs a recovery procedure to fix these errors. Such an occurrence is logged in the operator log (syslog on UNIX) with SHMLRECOV message.

Action: Report the occurrence to your GT.M support channel. No user action required. GT.M will continue to operate normally.

SHMREMOVED

SHMREMOVED, Removed Shared Memory id mmmm corresponding to file ffff

MUPIP Information: MUPIP RUNDOWN removed shared memory segment mmmm, corresponding to the file ffff, which could be a database file or a replication instance file with a GT.M signature because the resource was not actively in use.

Action: No action required.

SHRMEMEXHAUSTED

SHRMEMEXHAUSTED, Attempt by process to use more shared memory than currently permitted by OpenVMS

Run Time Error: A OpenVMS out-of-memory error was encountered while trying to open a shared global section for a database file.

Action: Reduce the shared global section usage by reducing the number of global buffers, or database block size, or the number of database files that the process tries to open.

SIGACCERR

SIGACCERR, Signal was caused by invalid permissions for mapped object

Run Time Error: This message is an auxiliary message and is preceeded with a primary KILLBYSIGxxx message.

Action: Refer to the accompanying message(s) and take appropriate action. Refer to the UNIX user documentation. If necessary, report the entire incident context to your GT.M support channel for further analysis.

SIGADRALN

SIGADRALN, Signal was caused by invalid address alignment

Run Time Error: This message is an auxiliary message and is preceeded with a primary KILLBYSIGxxx message.

Action: Refer to the accompanying message(s) and take appropriate action. Refer to the UNIX user documentation. If necessary, report the entire incident context to your GT.M support channel for further analysis.

SIGADRERR

SIGADRERR, Signal was caused by non-existent physical address

Run Time Error: This message is an auxiliary message and is preceeded with a primary KILLBYSIGxxx message.

Action: Refer to the accompanying message(s) and take appropriate action. Refer to the UNIX user documentation. If necessary, report the entire incident context to your GT.M support channel for further analysis.

SIGBADSTK

SIGBADSTK, Signal was caused by an internal stack error

Run Time Error: This message is an auxiliary message and is preceded with a primary KILLBYSIGxxx message.

Action: Refer to the accompanying message(s) and take appropriate action. Refer to the UNIX user documentation. If necessary, report the entire incident context to your GT.M support channel for further analysis.

SIGCOPROC

SIGCOPROC, Signal was caused by a coprocessor error

Run Time Error: This message is an auxiliary message and is preceded with a primary KILLBYSIGxxx message.

Action: Refer to the accompanying message(s) and take appropriate action. Refer to the UNIX user documentation. If necessary, report the entire incident context to your GT.M support channel for further analysis.

SIGFLTDIV

SIGFLTDIV, Signal was caused by a floating point divided by zero

Run Time Error: This message is an auxiliary message and is preceded with a primary KILLBYSIGxxx message.

Action: Refer to the accompanying message(s) and take appropriate action. Refer to the UNIX user documentation. If necessary, report the entire incident context to your GT.M support channel for further analysis.

SIGFLTINV

SIGFLTINV, Signal was caused by an invalid floating point operation

Run Time Error: This message is an auxiliary message and is preceded with a primary KILLBYSIGxxx message.

Action: Refer to the accompanying message(s) and take appropriate action. Refer to the UNIX user documentation. If necessary, report the entire incident context to your GT.M support channel for further analysis.

SIGFLTOVF

SIGFLTOVF, Signal was caused by a floating point overflow

Run Time Error: This message is an auxiliary message and is preceded with a primary KILLBYSIGxxx message.

Action: Refer to the accompanying message(s) and take appropriate action. Refer to the UNIX user documentation. If necessary, report the entire incident context to your GT.M support channel for further analysis.

SIGFLTRES

SIGFLTRES, Signal was caused by a floating point inexact result

Run Time Error: This message is an auxiliary message and is preceded with a primary KILLBYSIGxxx message.

Action: Refer to the accompanying message(s) and take appropriate action. Refer to the UNIX user documentation. If necessary, report the entire incident context to your GT.M support channel for further analysis.

SIGFLTUND

SIGFLTUND, Signal was caused by a floating point underflow

Run Time Error: This message is an auxiliary message and is preceded with a primary KILLBYSIGxxx message.

Action: Refer to the accompanying message(s) and take appropriate action. Refer to the UNIX user documentation. If necessary, report the entire incident context to your GT.M support channel for further analysis.

SIGILLADR

SIGILLADR, Signal was caused by illegal addressing mode

Run Time Error: This message is an auxiliary message and is preceded with a primary KILLBYSIGxxx message.

Action: Refer to the accompanying message(s) and take appropriate action. Refer to the UNIX user documentation. If necessary, report the entire incident context to your GT.M support channel for further analysis.

SIGILLOPC

SIGILLOPC, Signal was caused by an illegal opcode

Run Time Error: This message is an auxiliary message and is preceded with a primary KILLBYSIGxxx message.

Action: Refer to the accompanying message(s) and take appropriate action. Refer to the UNIX user documentation. If necessary, report the entire incident context to your GT.M support channel for further analysis.

SIGILLOPN

SIGILLOPN, Signal was caused by an illegal operand

Run Time Error: This message is an auxiliary message and is preceded with a primary KILLBYSIGxxx message.

Action: Refer to the accompanying message(s) and take appropriate action. Refer to the UNIX user documentation. If necessary, report the entire incident context to your GT.M support channel for further analysis.

SIGILLTRP

SIGILLTRP, Signal was caused by an illegal trap

Run Time Error: This message is an auxiliary message and is preceded with a primary KILLBYSIGxxx message.

Action: Refer to the accompanying message(s) and take appropriate action. Refer to the UNIX user documentation. If necessary, report the entire incident context to your GT.M support channel for further analysis.

SIGINTDIV

SIGINTDIV, Signal was caused by an integer divided by zero

Run Time Error: This message is an auxiliary message and is preceded with a primary KILLBYSIGxxx message.

Action: Refer to the accompanying message(s) and take appropriate action. Refer to the UNIX user documentation. If necessary, report the entire incident context to your GT.M support channel for further analysis.

SIGINTOVF

SIGINTOVF, Signal was caused by an integer overflow

Run Time Error: This message is an auxiliary message and is preceded with a primary KILLBYSIGxxx message.

Action: Refer to the accompanying message(s) and take appropriate action. Refer to the UNIX user documentation. If necessary, report the entire incident context to your GT.M support channel for further analysis.

SIGMAPERR

SIGMAPERR, Signal was caused by an address not mapped to an object

Run Time Error: This message is an auxiliary message and is preceded with a primary KILLBYSIGxxx message.

Action: Refer to the accompanying message(s) and take appropriate action. Refer to the UNIX user documentation. If necessary, report the entire incident context to your GT.M support channel for further analysis.

SIGOBJERR

SIGOBJERR, Signal was caused by an object specific hardware error

Run Time Error: This message is an auxiliary message and is preceded with a primary KILLBYSIGxxx message.

Action: Refer to the accompanying message(s) and take appropriate action. Refer to the UNIX user documentation. If necessary, report the entire incident context to your GT.M support channel for further analysis.

SIGPRVOPC

SIGPRVOPC, Signal was caused by a privileged opcode

Run Time Error: This message is an auxiliary message and is preceded with a primary KILLBYSIGxxx message.

Action: Refer to the accompanying message(s) and take appropriate action. Refer to the UNIX user documentation. If necessary, report the entire incident context to your GT.M support channel for further analysis.

SIGPRVREG

SIGPRVREG, Signal was caused by a privileged register

Run Time Error: This message is an auxiliary message and is preceded with a primary KILLBYSIGxxx message.

Action: Refer to the accompanying message(s) and take appropriate action. Refer to the UNIX user documentation. If necessary, report the entire incident context to your GT.M support channel for further analysis.

SIZENOTVALID4

SIZENOTVALID4, Size (in bytes) must be either 1, 2, or 4

Run Time error: The DSE CACHE command triggers this error when the size operand is not 1, 2, or 4.

Action: Specify 1, 2, or 4 as size (in bytes).

SIZENOTVALID8

SIZENOTVALID8, Size (in bytes) must be either 1, 2, 4, or 8

Run Time error: Both the DSE CHANGE -FILEHEADER command and MUIP REPLICATE -SOURCE -JNLPOOL -CHANGE command triggers this error when the SIZE qualifier is not set to 1, 2, 4, or 8.

Specify 1, 2, 4, or 8 as size (in bytes).

SNAPSHOTNOV4

SNAPSHOTNOV4, Cannot downgrade (to V4) while snapshots are in progress. Currently ssss snapshots are in progress for region rrrr.

Mupip error: A request to downgrade a region to V4 occurred while a snapshot is in progress.

Action: Wait for a currently active process using snapshots to complete before running the downgrade. Since a downgrade to V4 would not normally be expected, check to verify that the downgrade invocation is appropriate.

SOCKACPT

SOCKACPT, Error accepting socket connection

Run Time Error: This indicates that the process of opening a socket resulted in a device error.

Action: Review the accompanying message(s) for additional information.

SOCKACTNA

SOCKACTNA, Action not appropriate for current socket

Compile Time Error: This indicates that the socket state does not support the attempted action.

Action: Review the code for compatible socket actions.

SOCKBFNOTEMPTY

SOCKBFNOTEMPTY, Socket buffer size cannot be set to xxxx due to aaaa bytes of buffered data. Read first.

Run Time Error: This indicates that a USE command attempted to adjust the size of the socket buffer while it contained data.

Action: Make sure the buffer is empty when adjusting the size.

SOCKETEXIST

SOCKETEXIST, Socket xxxx already exists

Run Time Error: This indicates that the system tried to create a socket on a device, which already had a socket with that name.

Action: Review the names of the sockets already present on that device and specify a unique name.

SOCKINIT

SOCKINIT, Error initializing TCP socket: (errno == aaaa) xxxx

Run Time Error: This indicates that the process of opening a socket resulted in a device error. xxxx is the text description of the failure for the OS service.

Action: Review the accompanying message(s) for additional information.

SOCKLISTEN

SOCKLISTEN, Error listening on a socket

Run Time Error: This indicates that GT.M was unable to listen in the specified socket.

Action: Review accompanying messages for more information on the cause of the failure.

SOCKMAX

SOCKMAX, Attempt to exceed maximum sockets xxx for the SOCKET device

Runtime Error: Attempting to connect more than the maximum number of sockets defined for the process triggers this error. xxx is the maximum for the current process.

Action: Reduce the number of connections or use a process that has a higher maximum number of sockets defined by the gtm_max_sockets environment variable.

SOCKNOTFND

SOCKNOTFND, Socket xxxx not found

Run Time Error: This indicates that GT.M was unable to find a socket with the name specified.

Action: Make sure the socket is created before an I/O operation attempts using it.

SOCKPARAMREQ

SOCKPARAMREQ, Socket device parameter is required for TCP open

Run Time Error: This indicates that a socket deviceparameter was not defined to OPEN a TCP connection.

Action: Establish a TCP connection by specifying the SOCKET deviceparameter for the OPEN command. For a more complete description of this deviceparameter refer to the Input/Output Processing chapter of the Programmer's Guide.

SOCKWAIT

SOCKWAIT Error waiting for socket connection

Run Time Error: This indicates that the process of opening a socket resulted in a device error.

Action: Review the accompanying message(s) for additional information.

SOCKWRITE

SOCKWRITE, Write to a TCP/IP socket failed

Run Time Error: This indicates that GT.M was unable to write to the specified TCP/IP socket.

Action: Review the accompanying messages for more information on the cause of the failure.

SPOREOL

SPOREOL, Either a space or an end-of-line was expected but not found

Compile Time Error: This indicates that a command that required an argument did not specify one or an ELSE attempted to specify an argument.

Action: Look for and correct any typographical errors.

SRCFILERR

SRCFILERR, Error with source file I/O on file xxxx

Run Time Error: This indicates that a ZCOMPILE, ZLINK, or auto-ZLINK encountered an error when it attempted to access source file xxxx. An M command may also report this error at compile-time.

Action: Use host operating system commands to list the file and review accompanying messages for additional information.

SRCLIN

SRCLIN, xxxx

Compile Time Information: This message displays the source code line where an error occurred.

SRCLNNTDSP

SRCLNNTDSP, Source lines exceeding www character width are not displayed

Compile Time Error: Displayed instead of the source line when source line exceeds 1023 characters.

Action: Refer to the source code. the line number and column number in the associated messages identify the position of the problem .Consider shortening the line, at least until the error is found and corrected.

SRCLOC

SRCLOC, At column xxxx, line yyyy, source module zzzz

Compile Time Error: GT.M uses this message to display the line, offset, and routine where it encountered a compile-time error. xxxx is the column. yyyy is the line number. zzzz is the routine name.

Action: Review the accompanying message(s) for additional information.

SRCLOCUNKNOWN

SRCLOCUNKNOWN, M source location unknown

Run Time Warning: This indicates that GT.M could not locate the source line associated with the error.

Action: Find out if source code is available to you, if not this message is expected. Otherwise, check ZROUTINES and file permissions.

SRCNAM

SRCNAM, in source module xxxx

Compile Time Information: This message identifies the module xxxx, which contains some other error.

Action: Review the accompanying message(s) for additional information.

SRCSRVEXISTS

SRCSRVEXISTS, Source server for secondary instance xxxx is already running with pid yyyy

MUPIP Error: This error is issued by a source server startup command if there is already a source server up and running for the secondary instance name specified in the command.

Action: Do not start multiple source servers for the same secondary instance.

SRCSRVNOTEXIST

SRCSRVNOTEXIST, Source server for secondary instance xxxx is not alive

MUPIP Error: This error is issued by a mupip replic -source command that specifies any one of activate, changelog, checkhealth, deactivate, shutdown, showbacklog, statslog, stopsourcefilter if it finds no Source Server up and running for the replicating (secondary) instance name specified in the command.

Action: Make sure the Source Server for the specified replicating instance name is up and running to provide working replication.

SRCSRVTOOMANY

SRCSRVTOOMANY, Cannot start more than xxxx source servers in primary instance file yyyy

MUPIP Error: A maximum of 16 active and/or passive source servers are allowed at any point in time per instance. If 16 source servers are already running and another source server startup is attempted, it will issue this error.

Action: Shutdown any active or passive source server to allow the new source server to start up.

SRVLCKWT2LNG

SRVLCKWT2LNG, PID pppp is holding the source server lock. Waited for mmmm minute(s). Now exiting

MUPIP Error: Issued by MUPIP ROLLBACK -ONLINE when it finds process pppp has not released the journal pool resource for mmmm minutes.

Action: Investigate the state of process pppp and whether it should be stopped by operator action.

SSATTACHSHM

SSATTACHSHM, Error while attaching to shared memory identifier iiii

Runtime Error: A GT.M process encountered error while trying to attach to shared memory created to manage a snapshot and reports the above error in the operator log.

Action: Examine the accompanying system error message and take appropriate action.

SSFILCLNUPFAIL

SSFILCLNUPFAIL, Error while unlinking snapshot file -- xxxx

Mupip error: An attempt to terminate snapshot file maintenance by GT.M updater processes encountered a problem.

Action: Try a MUPIP RUNDOWN. If that has a similar problem, it may be prudent to shut down all access to the database in question in order to stop the burden of maintaining the snapshot file and to ensure it doesn't unnecessarily consume more space.

SSFILOPERR

SSFILOPERR, Error while doing oooo operation on file ffff

Mupip error: This operator log message indicates operation oooo on snapshot file ffff failed. Note in certain timing situations this action might be reported to the operator log after the snapshot consuming process (say MUPIP INTEG) has finished with the snapshot file, in which case it's harmless. If the consuming process issues a REGSSFAIL error, then it was definitely prevented from completing its task because of the SSFILOPERR.

Action: Analyze the operation and the file characteristics and take appropriate action to clear the problem.

SSPREMATEOF

SSPREMATEOF, Premature end of file while reading block nnnn of size: bbbb bytes at offset: oooo from zzzz

Mupip error: The action attempted access to a block beyond the end of the snapshot file. This means either the process was confused or the file is damaged

Action: Retry the action. If the problem persists, contact FIS with information on how to recreate the problem.

SSSHMCLNUPFAIL

SSSHMCLNUPFAIL, Error while doing snapshot shared memory cleanup. Operation -- ssss. Identifier -- dddd

Mupip error: There was an error while doing a snapshot cleanup. The operation ssss indicates what system call failed. The identifier dddd indicates the shared memory identifier that is being cleaned up.

Action: Analyze the failure details and take corrective measures. If appropriate carefully clear abandoned resources using the system ipcrm utility.

SSTMPCREATE

SSTMPCREATE, Cannot create the temporary file in directory dddd for the requested snapshot

Mupip error: An action requiring a snapshot file was unable to create it.

Action: Verify the directory has appropriate access permissions for the user performing the action.

SSTMPDIRSTAT

SSTMPDIRSTAT, Cannot access temporary directory dddd

Mupip error: An action requiring a snapshot file was unable to access the temporary directory.

Action: Verify the directory exists and has appropriate access permissions for the user performing the action.

SSTMPFILOPEN ●

Last used version: **V5.4-002A**

SSTMPFILOPEN, Failed to open shadow snapshot file ffff

Mupip error: An action requiring a snapshot file was unable to open it.

Action: Verify the file exists and has appropriate access permissions for the user performing the action.

SSV4NOALLOW

SSV4NOALLOW, Database snapshots are supported only on fully upgraded V5 databases. nnnn has V4 format blocks.

Mupip error: An action requiring a snapshot was attempted on a database the contains V4 format blocks.

Action: Upgrade the database to V5 and re-run the action.

STACKCRIT

STACKCRIT, Stack space critical

Run Time Error: This indicates that the process has consumed almost all of the available stack space.

Action: Look for infinite recursion. If you do not take immediate action to reduce your stack, a STACKOFLOW error will occur, which terminates your GT.M image. Examine the stack with ZSHOW. Trim the stack using QUIT, ZGOTO, or HALT.

STACKOFLOW

STACKOFLOW, Stack overflow

Run Time Fatal Error: This indicates that the process required more stack space than was available in memory.

Action: Reduce the stack when you get a STACKCRIT error. This error terminates the image.

STACKUNDERFLO

STACKUNDERFLO, Stack underflow

Run Time Error: This indicates that the process stack was corrupt.

Action: Review the accompanying messages for additional information. If necessary, report the entire incident context to your GT.M support channel for further analysis.

STARFILE

STARFILE, Star(*) argument cannot be specified with xxxx

MUPIP Error: This indicates that the qualifier xxxx, specified with the MUPIP JOURNAL command does not allow star (*) as an argument.

Action: Specify the journal file names explicitly.

STATCNT

STATCNT, xxxx: Key cnt: yyyy max subsc len: zzzz max data len: wwwww

MUPIP Information: LOAD uses this message to display status. xxxx is the name of the global being updated. yyyy is the number of nodes handled. zzzz is the largest subscript encountered. wwwww is the size of the largest node.

STOPTIMEOUT

STOPTIMEOUT, Waited too long for stopped process to release. Region: xxxx.

Run Time Error: This indicates that GT.M is ignoring and bypassing a locked process and a recovery mechanism has taken control of the listed region.

Action: Terminate the misbehaving process.

STRINGOFLOW

STRINGOFLOW, String pool overflow

Compile Time Error: This indicates that some action attempted to use the string pool when it was full.

Action: Look for very large string constants in a very large program; reduce one or both.

STRMSEQMISMATCH

STRMSEQMISMATCH, Unable to play update on Stream nnnn with seqno xxxx as receiving instance has a different stream seqno XXXX

MUPIP Error: Issued by the Update Process on a supplementary instance started with -UPDNOTOK (that is, local updates are disabled) when it finds the source and receiving instances have different values of stream sequence number for the non-Supplementary stream nnnn (can be any value from 1 through 15). This indicates the two instances are not in sync at least with respect to stream nnnn and so replication cannot proceed.

Action: Reinitialize the receiving instance from a backup of the source instance and restart replication between the two instances.

STRMNUMIS

STRMNUMIS, Stream # is ssss

MUPIP Information: Issued by a Receiver Server to designate a stream associated with the the immediately preceding message.

Action: Refer to the associated prior message.

STRMNUMMISMATCH1

STRMNUMMISMATCH1, Stream nnnn exists on the receiver instance file but is unknown on the source instance

MUPIP Error: Issued by a Source Server on a Supplementary Instance when it detects a non-Supplementary stream number nnnn (which can be any value from 1 through 15) exists on the receiving instance but not on the source instance. This indicates the two instances are not in sync at least with respect to stream nnnn, so replication cannot proceed.

Action: Reinitialize the receiving instance from a backup of the source instance and restart replication between the two instances.

STRMNUMMISMATCH2

STRMNUMMISMATCH2, Stream nnnn exists on the source instance file but is unknown on the receiver instance

MUPIP Error: Issued by a Source Server on a supplementary instance when it detects a non-Supplementary stream number nnnn (which can be any value from 1 through 15) exists on the source instance but not on the receiving instance. This indicates the two instances are not in sync at least with respect to stream nnnn and so replication cannot proceed.

Action: Reinitialize the receiving instance from a backup of the source instance and restart replication between the two instances.

STRNOTVALID

STRNOTVALID, Error: cannot convert xxxx value to valid yyyy value.

DSE Error: This error shows in DSE when there is a string input that cannot be converted to a number. For example, attempting "change -fileheader -location=0x123rt456" would apply to this error.

Action: Review and correct typographical errors.

STRUNXEOR

STRUNXEOR, xxxx unexpected end of record in string subscript

MUPIP Error: This indicates that LOAD aborted because it encountered an end of file while processing string subscript xxxx.

Action: Refer to the topic MUPIP LOAD Errors in About This Manual section in this manual.

STPEXPFAIL

STPEXPFAIL, Stringpool expansion failed. It could not expand to xxxx bytes

Run Time Error: The stringpool, an internally expanding data structure maintained by GT.M to store primarily M-local variable content, needs more memory than is available in the process virtual memory.

Action: Increase process memory quotas to increase available process virtual memory. Change application to reduce memory requirements of the stringpool by using lesser M-local variables.

STUCKACT

STUCKACT, Process stuck script invoked: rrrr : pppp

Runtime Information: This message shows the success or failure status return rrrr of the operation of invoking the script pppp pointed to by the environment variable \$gtm_procstuckexec

Action: If the result is success analyze the output of the script. If the result is failure, check the script and any output it produced up to the point of failure and rework the script or adjust the environment appropriately.

SUB2LONG

SUB2LONG, Subscript invalid, too long

MUPIP Error: This indicates that INTEG encountered a subscript that is too long for its display mechanism.

Action: Examine the subscript with DSE DUMP, and take action to eliminate the subscript if it is invalid.

SUPRCVRNEEDSSUPSRC

SUPRCVRNEEDSSUPSRC, Instance iiiii is not configured to perform local updates, so it cannot act as a receiver for non-Supplementary Instance sssss

MUPIP Error: Issued by a Receiver Server or a MUPIP JOURNAL -ROLLBACK -FETCHRESYNC on a Supplementary Instance sssss started with -UPDNOTOK attempted to connect to non-Supplementary Instance iiiii. A Supplementary Instance that does not permit local updates can only replicate from another Supplementary Instance.

Action: Reconfigure the instances to a supported configuration.

SUSPENDING

SUSPENDING, Suspending processing on user request or attempt to do terminal I/O while running in the background

Run Time Information: The message signifies that a GT.M process is suspended on user initiated ^Z (or key stroke that is set to shell "susp"). It is also displayed if the process attempts to do terminal I/O while running in the background. Before suspending itself, the process logs the SUSPENDING message to the operator facility. Suspended processes may be automatically released from that state if they hold a shared resource that blocks other processes.

SVNEXPECTED

SVNEXPECTED, Special variable expected in this context

Compile Time/Run Time Error: This indicates that GT.M encountered a dollar sign in a NEW command that was not followed by a valid special variable name.

Action: Look for misspelled special variable names or a missing \$ in an extrinsic.

SVNONEW

SVNONEW, Cannot NEW this special variable

Compile Time/Run time error: This indicates that a NEW command tried to new an intrinsic special variable that is not a valid argument for a NEW.

Action: Look for inappropriate \$ prefixes. \$ZTRAP, \$ETRAP, \$ESTACK, \$ZYERROR, \$ZGBLDIR are the only intrinsic special variables that can be NEWed.

SVNOSET

SVNOSET, Cannot SET this special variable

Compile Time Error: This indicates that a SET command specified an assignment that attempted to modify a read-only special variable.

Action: Look for inappropriate \$ prefixes and attempts to modify read-only special variables. \$DEVICE, \$ECODE, \$ETRAP, \$DEVICE, \$KEY, \$X, and \$Y are the only ANSI standard variables that can be SET.

SYSCALL

SYSCALL, Error received from system call xxxx -- called from module yyyy at line zzzz

Run Time Error: This indicates that system call failed due to some unusual error condition.

Action: Report to system administrator and if necessary report the entire incident context to your GT.M support channel for further analysis.

TCGETATTR

TCGETATTR, Error while getting terminal attributes on file descriptor xxxx

Run Time Error: This indicates that the terminal attributes are inaccessible where xxxx is the file descriptor for the device.

Action: Review and correct the OS configuration of the device.

TCOMMITDISALLOW

TCOMMITDISALLOW, TROLLBACK required after an unhandled error in trigger context

Runtime Error: This transaction did an update that invoked a trigger which in turn encountered an error that was not handled by the application error trap inside the trigger context. Because of this, the exit from trigger context was abnormal. GT.M does not commit such transactions since they would not preserve the atomicity of trigger updates (triggering update + triggered updates).

Action: Such transactions can only be rolled back. If this is a nested TSTART (subtransaction), it can optionally be rolled back incrementally, that is, only the nested TSTART needs to be rolled back while the parent TSTART can still be committed.

TCSETATTR

TCSETATTR, Error while setting terminal attributes on file descriptor xxxx

Run Time Error: This indicates that the terminal attributes are inaccessible where xxxx is the file descriptor for the device.

Action: Review and correct the OS configuration of the device.

TERMASTQUOTA

TERMASTQUOTA, Process AST quota exceeded, cannot open terminal

Run Time Error: This indicates that an OPEN command failed because it required an AST that would violate the OpenVMS process quota.

Action: Reduce the number of terminals in use by a single process or ask your system administrator about changing your AST quota.

TERMWRITE

TERMWRITE, Error writing to terminal, status:

Run Time Error: This indicates that a WRITE to a terminal failed. Such failures may be detected and reported asynchronously to the actual WRITE command.

Action: Review the accompanying message(s) for additional information.

TEXT

TEXT, xxxx

Run Time Information: GT.M uses this message with various accompanying text, xxxx, to expand on other errors.

Action: Examine the text and review any accompanying message(s).

TEXT

GT.CM server does not support global names longer than 8 characters

GT.CM GNP Client Error: The GT.M GNP client process contains a global variable name longer than 8 characters while the version running at the server does not support global names longer than 8 characters.

Action: Upgrade the server to V5.0-FT01 or later. If upgrade is not a possible option, avoid using global variable names longer than 8 characters at the client.

TEXT

GT.CM client does not support global names longer than 8 characters

GT.CM GNP Server Error: The result returned by the GT.M GNP server process contains a global variable name longer than 8 characters while the version running at the client does not support global names longer than 8 characters.

Action: Upgrade the client to V5.0-FT01 or later. If upgrade is not a possible option, avoid using global variable names longer than 8 characters at the server.

TEXTARG

TEXTARG, Invalid argument to \$TEXT function

Compile Time Error: This indicates that a \$TEXT function specified an invalid argument.

Action: Modify the \$TEXT() argument so it is in the format of an entryref.

TIMRBADVAL

TIMRBADVAL, Bad value specified. Timer not changed.

DSE Error: This indicates that a CHANGE command with the FILEHEADER qualifier specified a time value that was improperly formatted or inappropriate.

Action: Modify the time value.

TIMERHANDLER

TIMERHANDLER, Incorrect SIGALRM handler xxxx found by yyyy

Run Time Information: This indicates that an external user-supplied routine (C or other language) called from GT.M, incorrectly manipulated the system timer handler. The xxxx is the hexadecimal address of the handler installed by the external routine and yyyy is the routine within GT.M, which discovered the problem.

Action: Use the GT.M provided timer facility described in the Programmer's Guide .

TIMEROVFL

TIMEROVFL, Timer overflow; interval probably too large

Run Time Error: This indicates that a timeout or timers calculation exceeded the maximum allowable interval.

Action: Check the system maximum interval and set the interval accordingly.

TLVLZERO

TLVLZERO, Transaction is not in progress

Compile/Run Time Error: This indicates that a TCOMMIT, TROLLBACK, or TRESTART command attempted to change the state of the current transaction; however, a transaction was not in progress.

Action: Look for missing TSTARTs, extra TCOMMITs or TROLLBACKs, or an unanticipated flow of control; conditionalize the command on \$TLEVEL, if appropriate.

TMPFILENOCRE

TMPFILENOCRE, Error in MUPIP BACKUP while trying to create temporary file xxxx

MUPIP Error: This indicates that MUPIP BACKUP was not able to create the temporary file xxxx needed in the course of its processing, possibly due to earlier incomplete backups.

Action: Make sure the temporary files are removed from the backup directory, and re-issue the BACKUP.

TMPSTOREMAX

TMPSTOREMAX, Maximum space for temporary values exceeded

Compile Time Error: This indicates that the routine requires more space to store intermediate results in expression evaluation than GT.M provides.

Action: Look for excessive function nesting in a very large routine. Reduce the size of the routine or the amount of nesting.

TNTOOLARGE

TNTOOLARGE, Database file xxx has reached the transaction number limit (0xaaa). Renew database with MUPIP INTEG TN_RESET

Run Time Information: This indicates that GT.M detected that the transaction numbers in the named database have reached the maximum number. There are 0xFFFFFFFF ($[2^{32} - 1]$ or 4,294,967,295 decimal) possible transaction numbers for V4 or 0xFFFFFFFFFFFFFFFF ($[2^{64} - 1]$ or 18,446,744,073,709,551,615 decimal) possible transaction numbers for V5 version databases. Note that the actual maximum TN is less than this theoretical limit. DSE DUMP FILEHEADER shows what the limit is. The actual limit reflects some overhead used, for example, during a TN_RESET operation.

Action: Use MUPIP INTEG with the qualifier TN_RESET to reset the transaction numbers in the database. If the database is in the V4 format, consider converting it to the V5 format. The database cannot otherwise be used until the condition is removed by either a TN_RESET or, if a V4 database, changing the output mode to V5 with MUPIP SET VERSION.

TNWARN

TNWARN, Database file xxx has 0xaaa more transactions to go before reaching the transaction number limit (0xaaa). Renew database with MUPIP INTEG TN_RESET

Run Time Information: This indicates that GT.M detected that the transaction numbers in the named database are approaching the maximum number. There are 0xFFFFFFFF ([math>2^{32} - 1] or 4,294,967,295 decimal) possible transaction numbers for V4 or 0xFFFFFFFFFFFFFFFF ([math>2^{64} - 1] or 18,446,744,073,709,551,615 decimal) possible transaction numbers for V5 version databases. This message is sent to the operator log periodically at decreasing intervals as the transaction number approaches the maximum. Note that the actual maximum TN is less than this theoretical limit. DSE DUMP FILEHEADER shows what the limit is. The actual limit reflects some overhead used, for example, during a TN_RESET operation.

Action: Use MUPIP INTEG with the qualifier TN_RESET to reset the transaction numbers in the database. If the database is in the V4 format, consider converting it to the V5 format.

TOOMANYCLIENTS

TOOMANYCLIENTS, GT.M is serving the maximum number of clients. Try again later.

Run Time Error: This indicates that the process failed in accessing a region served via GT.CM server, which currently cannot accept another connection. This is unlikely to happen unless many of the servers clients have abruptly disconnected and the server has been running for a long time.

Action: Try again later. Stop and restart the server to resolve the problem. If the problem persists, contact the group responsible for database operations on your network.

TOTALBLKMAX

TOTALBLKMAX, Extension exceeds maximum total blocks, not extending

Run Time Error: This indicates that the database file extension specified implicitly or explicitly (using MUPIP EXTEND) would cause the GDS file to exceed the maximum size of 64 million blocks.

Action: Modify the extension to use a smaller size. This may indicate that you should move some contents of the database file to another file.

TPFAIL

TPFAIL, Transaction COMMIT failed. failure code: xxxx.

Run Time Error: This indicates that GT.M attempted to process this transaction four times, but encountered an error every time. xxxx contains the failure codes for the four attempts. It is very likely that the database may have integrity errors or that the process-private data structures are corrupted.

Action: Report this database error to the group responsible for database integrity at your operation.

TPLOCK

TPLOCK, Cannot release lock(s) held prior to current TSTART

Run Time Error: This indicates that a LOCK or ZDEALLOCATE command attempted to release named resources that were LOCKed or ZALLOCATED prior to the initial TSTART of the current transaction.

Action: Do not release the named resource until after the TCOMMIT, or LOCK/ZALLOCATE has reserved the resource after the TSTART has begun.

TPLOCKRESTMAX

TPLOCKRESTMAX, Transaction restarts due to unavailability of locks not allowed in a final TP retry more than nnnn times

Run Time Error: This indicated a timed LOCK within a transaction was consistently unavailable. In order to prevent the process from waiting for the LOCK while holding a database resource (critical section) the transaction has restarted nnnn times without success. This error limits the possibilities for this issue cascading into a live-lock (consuming resources trying to do something that is "not happening").

Action: Analyze the locking protocol for issues of dead lock or unexpected LOCK durations and rework appropriately. Note that FIS recommends against using LOCKs within transactions, as GT.M protects the transaction integrity independent of LOCK protocols. If you wish to impose a conventional locking strategy for a transaction, place the LOCK and unlock around (outside) the transaction. While it is possible to use a LOCK within a transaction for signaling, that technique is typically problematic as it violates transactional Isolation and should likely be restricted to testing.

TPMIXUP

TPMIXUP, xxxx transaction cannot be started within yyyy transaction

Run Time Error: This indicates that the software, function, or routine may use incompatible transaction fencing.

Action: Transaction fences and ZT transaction fences cannot be used in combination. Review the code for incompatible use of T and ZT transaction fences.

TPNOTACID

TPNOTACID, tttt at xxxx in a final TP retry violates ACID properties of a TRANSACTION; indefinite RESTARTs may occur

Run Time Information: GT.M issues this message if it is executing a TP TRANSACTION in the final retry and control gets transferred out of GT.M due to any one of three conditions (i) ZSYSTEM command (ii) Runtime error that transfers control to a user-defined \$ZTRAP/ETRAP or (iii) Entering direct mode (e.g. due to a BREAK command). The xxxx indicates the \$ZPOSITION where the transfer of control occurred and the condition that caused this is identified in tttt.

Action: Modify the program to avoid the transfer of control outside of GT.M at the M-source location identified. For example, if the M-program contains a ZSYSTEM, try placing it outside the TRANSACTION.

TPQUIT

TPQUIT, Cannot QUIT out of a routine with an active transaction

Run Time Error: This indicates that an implicit or explicit QUIT attempted to leave an invocation level where a transaction or subtransaction was TSTARTed but not matched by either a TCOMMIT or TROLLBACK.

Action: Modify the routine to TCOMMIT or TROLLBACK the transaction before QUITting, or move the TSTART to a prior invocation level.

TPRESTART

TPRESTART, Database mmmm; code: xxxx; blk: yyyy in glbl: zzzz; pvtmods: aaaa, blkmods: bbbb, blklvl: cccc, type: dddd, readset: eeee, writeset: ffff, local_tn: gggg

Severity: Information

Run Time Information: This UNIX environment variables or OpenVMS logical names GTM_TPRESTART_LOG_FIRST and GTM_TPRESTART_LOG_DELTA control the logging of TPRESTART messages. GTM_TPRESTART_LOG_FIRST indicates the number of TP restarts to log from GT.M invocation. Once tha many have been logged, for every GTM_TPRESTART_LOG_DELTA TP restarts, a restart message is logged. If GTM_TPRESTART_LOG_DELTA is undefined no operator logging occurs. The default value for GTM_TPRESTART_LOG_FIRST is 0 (zero), which leaves the control completely with GTM_TPRESTART_LOG_DELTA. This message is supposed to serve as a debugging tool in developmental environments to indicate globals of contention.

Action: Disable or adjust the frequency of these messages with the mechanism described above. To reduce the number of restarts, consider changes to the global structure, varying the time when work is scheduled, whether the business and program logic permits the use of NOISOLATION.

TPSTACKCRIT ●

Last used version: **V5.3-004A**

TPSTACKCRIT, TP Stack space critical

Run Time Error: This indicates that a transaction has consumed almost all of the available stack space.

Action: Look for missing TCOMMIT commands. Modify the code to reduce the total change content of the transaction; pay special attention to recursion and variable usage.

TPSTACKOFLOW ●

Last used version: **V5.3-004A**

TPSTACKOFLOW, TP Stack overflow

Run Time Error: This indicates that a transaction has consumed all of the available stack space. This error terminates the image.

Action: Modify the error handling for TPSTACKCRIT to prevent a subsequent TPSTACKOFLOW.

TPTIMEOUT

TPTIMEOUT, Transaction timeout

Run Time Error: This indicates that the transaction took too long to process successfully. Timeouts prevent runaway processes, damage from software bugs, and transactions so large as to be hostile to other users.

Action: Review accompanying message(s) for more information about what caused the transaction timeout. If the transaction uses custom programs, routines, or functions, they may need to be debugged.

TPTOODEEP

TPTOODEEP, \$TLEVEL cannot exceed 127

Run Time Error: This indicates that a TSTART attempted to initiate more concurrent subtransactions than GT.M permits.

Action: Determine whether the transaction is implemented as designed. Modify the routine to reduce the levels of subtransaction nesting.

TRACEON

TRACEON, Missing global name (with optional subscripts) to dump M-tracing information into

Compile Time Error: This indicates that no global variable was supplied on a VIEW "TRACE":1:"^gvn".

Action: Supply the global variable.

TRACINGON

TRACINGON, Tracing already turned on

Run Time Information: This indicates that M profiling is already turned on when a command was issued to turn tracing on.

TRANS2BIG

TRANS2BIG, Transaction exceeded available buffer space for region rrrr

Run Time Error: This indicates that a transaction updated more blocks than the global buffer could hold for a particular region rrrr or accessed more than the single transaction limit of 64K blocks.

Action: Look for missing TCOMMIT commands; modify the code to reduce the total content or change content of the transaction. If the transaction is as intended and the issue is the number of updates, increase the GLOBAL_BUFFERS for the region using MUPIP SET, or modify the Global Directory to redistribute the relevant globals to more regions.

TRANSMINUS

TRANSMINUS, Negative numbers not allowed with ZTCOMMIT

Run Time Error: This indicates that a ZTCOMMIT must have a zero (0) or positive integer argument.

Action: Modify the ZTCOMMIT argument.

TRANSNEST

TRANSNEST, Maximum transaction nesting levels exceeded

Run Time Error: This indicates that a ZTSTART failed an attempt to establish another level of nested subtransactions.

Action: Rework the application so that it requires no more than 255 levels of journal transaction nesting.

TRANSNOSTART

TRANSNOSTART, ZTCOMMIT(s) issued without corresponding ZTSTART(s)

Run Time Error: This indicates that a ZTCOMMIT specified an explicit number of ZTSTARTs to close that exceeded the number of ZSTARTs that were currently open.

Action: Use ZTCOMMIT with an argument of 0 to close all open transactions, or modify the ZTCOMMIT to use a positive integer argument that does not exceed the number of open ZTSTARTs.

TRESTLOC

TRESTLOC, Transaction start: xxxx, Transaction failure: yyyy

Run Time Error: This indicates that GT.M detected a resource conflict that attempted to RESTART a transaction that did not enable RESTART. This message follows TRESTNOT error message, xxxx is the starting location for the transaction and yyyy is the point of failure.

Action: Enable RESTART with the initial TSTART command argument, or add external LOCKs to serialize the transaction.

TRESTMAX

TRESTMAX, TRESTART not allowed in a final TP retry more than once

Run Time Error: The code contained one or more explicit TRESTART command(s) which called for a RESTART after more than once after \$TRESTART reached 3. Once \$TRESTART reaches 3, GT.M switches from an optimistic to a conventional locking strategy so such restarts are expensive and potentially disruptive. Therefore GT.M limits the ability of the application code to repeatedly demand such RESTARTs.

Action: Investigate the cause and reason(s) for the explicit restart. When combined with a subsequent TROLLBACK, explicit TRESTARTs may provide a way to manipulate the variable context and flow of control in useful ways, but other uses are unusual. If explicit TRESTARTs are appropriate, consider whether they should be conditional on the value of \$TRESTART.

TRESTNOT

TRESTNOT, Cannot TRESTART, transaction is not restartable

Run Time Error: This indicates that a TRESTART command attempted to RESTART a transaction that did not enable RESTART.

Action: Enable RESTART with the initial TSTART command argument, add external LOCKs to serialize the transaction, or eliminate the TRESTART command.

TRIG2NOTRIG

TRIG2NOTRIG, Sending transaction sequence number xxxx which used triggers to a replicator that does not support triggers

MUPIP Warning: The source server encountered a transaction that includes triggers, but its replicating node does not support triggers. Unless you are using application level filters to handle this case, your originating instance and replicating instance are no longer consistent.

Action: If this case it not handled by your application level filters, you should either enhance your filters or upgrade the replicating instance to a version of GT.M that supports triggers and load the the appropriate trigger definitions with MUPIP TRIGGER (or \$ZTRIGGER()), and then take appropriate action (such as recreating the replicating instance from a backup of the originating instance) to restore consistency.

TRIGCOMPFAIL

TRIGCOMPFAIL, Compilation of database trigger named tttt failed

Run Time Trigger Error: The -Xecute code of a trigger specification has syntax errors. Because triggers are precompiled when you define them, this error may indicate that either:

- A database upgrade was performed but the trigger code was not updated to eliminate obsolete syntax
- The portion of the database holding the trigger definitions may be corrupted

Action: Validate the definitions by a SELECT option with MUPIP TRIGGER or \$ZTRIGGER(), correct the trigger code syntax and apply a trigger update.

TRIGDATAIGNORE

TRIGDATAIGNORE, Ignoring trigger data tttt. Use MUPIP TRIGGER to load trigger definitions

MUPIP informational message : MUPIP LOAD displays this warning when it encounters trigger metadata during extract file processing (GO/ZWR extracts).

Action: Identify and remove trigger metadata information from GO/ZWR extract files and, if appropriate, process it with MUPIP TRIGGER or \$ZTRIGGER().

TRIGDEFBAD

TRIGDEFBAD, Trigger initialization failed for global ^gggg. Error while processing ^#t("xxxx",yyy[,zzz])

GT.M/MUPIP error: Missing or corrupted trigger metadata causes this error.

Action: Delete and replace defective triggers. If possible analyze the cause of the trigger damage and report the incident to your GT.M support channel.

TRIGDEFNOSYNC

TRIGDEFNOSYNC, Global ^gggg has triggers defined on the [originating/replicating] instance but none on the [replicating/originating] instance. Current journal sequence number is 0xjjjj

Update Process Warning: The Update Process detected that there is a mismatch in trigger definitions for global gggg between originating and replicating instances and sent this warning to the operator log.

Action: Differences in triggers between originating and replicating instances typically mean the replicating instance is not in a position to stand in as a system of record by becoming an originating instance. Unless the difference is intended because of a special use of the replicating instance, shutdown and resynchronize the replicating instance.

TRIGINVCHSET

TRIGINVCHSET, Trigger tttt for global gggg was created with CHSET=cccc which is different from the current \$ZCHSET of this process

Run Time Trigger Error: Trigger tttt on global gggg failed because the process that attempted to update global gggg did not have the same character set that was used to load trigger tttt. Databases with triggers can only be used by processes that are M mode or UTF-8 mode, depending on the mode of the process that loaded the triggers.

Action: Ensure that processes start with the same character set (as defined by the `gtm_chset` environment variable) that was used to load the trigger definitions with `MUIP TRIGGER` (or `$ZTRIGGER()` function).

TRIGIS

TRIGIS, Trigger name: `tttt`

Run time information: This message identifies a trigger name

Action: Refer to the accompanying message(s) for more information.

TRIGMODINTP

TRIGMODINTP, Triggers for a given global cannot be both used and modified or removed in the same transaction

Run Time Error: Once a transaction has used any trigger on any node associated with a given global name (the part of the variable designation between the up-arrow (^) and the left-parenthesis "(" (or end if there's no subscript), you cannot use that same transaction to modify definitions for any trigger associated with that global name.

Action: Rework the transaction and `$ZTRIGGER()` relationship to avoid this issue.

TRIGMODREGNOTRW

TRIGMODREGNOTRW, Trigger(s) cannot be added/changed/deleted because region `rrrr` is read-only

Runtime Error: This error occurs when `$ZTRIGGER()` or `MUIP TRIGGER` attempts to write to read-only region `rrrr`.

Action: Check for appropriate global directory mapping and appropriate permissions on the database file mapped to region `rrrr`.

TRIGNAMBAD

TRIGNAMBAD, Trigger initialization failed. Error while processing `^#t(tttt,cccc)`

Run Time or `MUIP` Error: A trigger operation encountered a trigger definition for trigger with an apparent internal inconsistency while looking for characteristic `cccc` of type `tttt`.

Action: Delete and redefine the trigger in question. Consult with the group responsible for database integrity at your operation to discuss what actions might have led to this error.

TRIGNAMENF

TRIGNAMENF, Trigger name `nnnn` not found with the current default global directory

Run Time or `MUIP` Error: This message indicates a trigger lookup by name failed. `GT.M` has a name cross reference in the default region of the current global directory. If you use multiple global directories with different default regions, trigger lookups by name, such as `$ZTRIGGER()`, `ZBREAK` and `ZPRINT`, only work when they use the same default region as the one in use at the time of the trigger definition.

Action: Consider using a `SET $ZGBLDIR` to change to an appropriate global directory when using name lookup. Also consider restructuring your global directories so they share a common default region.

TRIGNAMEUNIQ

TRIGNAMEUNIQ, Unable to make trigger name tttt unique beyond vvvv versions already loaded

Run Time Trigger Error: GT.M encountered more than vvvv different instances of the same trigger name across database regions used by the same process.

Action: Revise trigger names to prevent such a high degree of overlap.

TRIGREPLSTATE

TRIGREPLSTATE, Trigger cannot update replicated database file dddd since triggering update was not replicated

Run Time Error: A process performed an update on a global in a database region which is not currently replicated, and that update invoked a trigger that, in turn, attempted an update on a global in a database region that is replicated. This would produce a journal state with insufficient information to properly recover the replicated region.

Action: Investigate whether the global directories, journaling characteristics or trigger logic need revision.

TRIGTCOMMIT

TRIGTCOMMIT, TCOMMIT at \$ZTLEVEL=LLLL not allowed as corresponding TSTART was done at lower \$ZTLEVEL=BBBB

Run Time Trigger Error: A TCOMMIT in trigger logic attempted to complete the active transaction that was started outside of the current trigger. Because trigger actions are atomic with the update initiating them, committing a transaction started prior to or by the triggering update cannot be committed inside the trigger.

Action: Within the trigger context, review the TCOMMIT logic to ensure that it commits only those transactions that are started within the trigger. Ensure that TCOMMIT does not attempt to commit any transaction started prior to or by the triggering update.

TRIGSUBSCRANGE

TRIGSUBSCRANGE, Trigger definition for global ^gggg has one or more invalid subscript range(s) : ssss

GT.M/MUPIP error : This error indicates one or more subscript range(s) of of order given the current collation subscript ordering - for global gggg in tthe trigger definition files.

Action: Verify the validity of subscript ranges in trigger definition file for the particular global, taking its collation into account, and redefine the trigger with correct subscript ranges for the collation of the global in question.

TRIGTLVLCHNG

TRIGTLVLCHNG, Detected a net transaction level (\$TLEVEL) change during trigger tttt. Transaction level must be the same at exit as when the trigger started

Run Time Trigger Error: While the trigger logic can use balanced sub-transactions, it cannot cause a net change in \$TLEVEL.

Action: Review the transaction management (TSTART, TCOMMIT and TROLLBACK) within trigger logic to ensure that it commits or rolls back any transactions it starts and does not attempt to commit any transaction started prior to, or by, the trigger update. You can use TROLLBACK within trigger logic to block the current transaction, possibly to write error context information.

Nonetheless if you use such a TROLLBACK, GT.M subsequently signals this error when you leave the trigger context in order to notify the process that the original triggering update has been discarded.

TRIGZBREAKREM

TRIGZBREAKREM, ZBREAK in trigger tttt removed due to trigger being reloaded

Run Time Warning: This indicates your process had a ZBREAK defined within the XECUTE code for trigger tttt, but some action replaced the definition for trigger tttt so GT.M removed the ZBREAK.

Action: If appropriate examine the trigger with ZPRINT and reestablish the ZBREAK. The message is tied to BREAKMSG mask 8 (See VIEW BREAKMSG). The default message mask is 15, which includes masks 1, 2, 4, and 8. Using the VIEW command to set the BREAKMSG mask to 7 or any other pattern that excludes 8, disables this message.

TRNLOGFAIL

TRNLOGFAIL, Translation of [OpenVMS] logical name or [UNIX] environmental variable xxxx failed

Compile Time Error: This indicates that translation of the indicated environment variable (UNIX)/logical name (OpenVMS) failed. The message is accompanied with another message describing the failure type and the reason behind it. Most probable cause of the failure could be resource limitation.

Action: Report to your system administrator.

TROLLBK2DEEP

TROLLBK2DEEP, Intended rollback (xxxx) deeper than the current \$tlevel (yyyy)

Run Time Error: This indicates that an attempt to TROLLBACK more levels (as indicated by \$TLEVEL) of transaction nesting than are currently active.

Action: Review the logic and code path that led to the error and modify the code appropriately.

TRUNCATE

TRUNCATE, Error while truncating jnl-file xxxx to length aaaa

Run Time Error: This message is issued by Recovery/Rollback process when it is unable to truncate the journal file at the end of recovery. The journal file remains in the same state as it was before the processing and may contain incomplete data, in case of a GT.M crash.

Action: Look into the secondary error message and if necessary report the entire incident context to your GT.M support channel for further analysis.

TRUNCATEFAIL

TRUNCATEFAIL, Truncating xxxx from aaaa OpenVMS blocks to bbbb blocks failed

MUPIP Error: This indicates that a MUPIP BACKUP copied a database that extended during the copy and the attempt to return it to the proper size failed.

Action: Examine the subsequent message for more information. It maybe possible to make the copy usable by performing a MUPIP EXTEND to the actual (current) size take the GDS blocks from the extension size in the file header, and verify the same against the reported RMS blocks. You will need to rename the file since the file in this state will have a temporary file name rather than the expected name from the BACKUP.

TSTRTPARM

TSTRTPARM, Error parsing TSTART qualifier

Compile Time Error: This indicates that the TSTART specified an improperly formatted argument.

Action: Modify the TSTART argument.

TTINVFILTER

TTINVFILTER, Invalid FILTER argument

Run Time Information: This indicates that the FILTER= deviceparameter appeared in a device command for a terminal with an argument that evaluated to something other than "[NO]CHARACTERS" or "ESCAPE", which are the only valid arguments.

Action: Remove or rework the FILTER=.

TTLENGHTOOBIG

TTLENGHTOOBIG, Terminal LENGTH exceeds the maximum allowed limit

Run Time Error: The LENGTH specified is too large for a terminal device. The maximum page length is 255.

Action: Modify the USE to specify a smaller LENGTH.

TTWIDTHTOOBIG

TTWIDTHTOOBIG, Terminal WIDTH exceeds the maximum allowed limit

Run Time Error: The WIDTH specified is too large for a terminal device. The maximum page width is 511.

Action: Modify the USE to specify a smaller WIDTH.

TXTNEGLIN

TXTNEGLIN, A line prior to line number zero was referenced in \$TEXT

Run Time Error: This indicates that a \$TEXT function argument referenced a line that preceded program source line zero.

Action: Verify that \$TEXT() arguments evaluate to zero (0) or to a line subsequent to the beginning of the routine.

TXTSRCFMT

TXTSRCFMT, \$TEXT encountered an invalid source program file format

Run Time Error: This indicates that a \$TEXT function encountered an improperly formatted source file. This error only occurs when the source file has been corrupted.

Action: Use a host system command to examine the source file. Determine whether the source file was properly maintained.

TXTSRCMAT

TXTSRCMAT, M object module and source file do not match

Run Time Error: This indicates that a \$TEXT function referenced a routine whose source file does not correspond to the current object file. ZPRINT may deliver this message as a warning.

Action: ZLINK the source file or use host shell commands to rearrange the current source file to match the object file.

UIDMSG

UIDMSG, Unidentified message received

Run Time Error: This indicates that the process was performing a JOB command and received an unanticipated message while attempting to communicate with the JOBbed process.

Action: Look for indiscriminate mailbox use by other processes.

UIDSND

UIDSND, Unidentified sender PID

Run Time Error: This indicates that the process was performing a JOB command and received a message from an unidentified source while attempting to communicate with the JOBbed process.

Action: Look for indiscriminate mailbox use by other processes.

UNDEF

UNDEF, Undefined local variable: xxxx

Run Time Error: This indicates that an expression referenced a local variable xxxx, that was not defined.

Action: Ensure that all variables are assigned values before they are referenced; use \$GET(), or change the image or process to NOUNDEF mode.

UNIMPLOP

UNIMPLOP, Unimplemented construct encountered

Run Time Error: This indicates that GT.M encountered an unsupported data type while passing arguments between typed C and type-less M.

Action: Review the call-in table and ensure that the parameter types match the following table:

Error Messages

Directions	Allowed Parameter types
I	gtm_long_t, gtm_ulong_t, gtm_float_t, gtm_double_t, gtm_long_t*, gtm_ulong_t*, gtm_float_t*, gtm_double_t*, gtm_char_t*, gtm_string_t*
O/IO	gtm_long_t*, gtm_ulong_t*, gtm_float_t*, gtm_double_t*, gtm_char_t*, gtm_string_t*

UNKNOWNFOREX

UNKNOWNFOREX, Process halted by a forced exit from a source other than MUPIP

Run Time Warning: This indicates that a process has been terminated by a source (usually a user program) other than MUPIP.

Action: Investigate why an operator or program is stopping GT.M processes without using MUPIP STOP.

UNSDCLASS

UNSDCLASS, Unsupported descriptor class

Run Time Error: This indicates that an external call or \$ZCALL function encountered an unsupported argument-passing mechanism in the external call table.

Action: Make sure the argument-passing mechanism is spelled correctly in the external call table.

UNSDDDTYPE

UNSDDDTYPE, Unsupported descriptor data type

Run Time Error: This indicates that an external call or \$ZCALL function encountered an unsupported data type in the external call table.

Action: Review the external call table for invalid data types.

UNSOLCNTERR

UNSOLCNTERR, An unsolicited error message has been received from the network

Run Time Error: This indicates that a GT.CM component received a network message from an unknown source.

Action: Ensure that no agents outside the GT.CM environment are improperly sending messages to GT.CM.

UPDATEFILEOPEN

UPDATEFILEOPEN, Update file open error

Run Time Error: This indicates that file permissions were either set incorrectly or replication was turned off.

Action: Review accompanying message(s) for addition information. Correcting accompanying messages should correct this message.

UPDSYNC2MTINS

UPDSYNC2MTINS, Can only UPDATERESYNC with an empty instance file

MUPIP Error: Issued by a Receiver Server started with the -UPDATERESYNC qualifier on a non-supplementary instance or on a Supplementary Instance started specifying the -UPDNOTOK qualifier, when the replication instance file on the Receiver side contains at least one history record. The purpose of -UPDATERESYNC is to unconditionally declare the instance state as a valid state in the set of current and prior states of the originating instance disregarding any history. Note that the receiver server on a Supplementary Instance started with -UPDOK does not issue this error.

Action: Verify that -UPDATERESYNC is appropriate and if so, recreate the instance file to discard the history then reissue the command. If the replication state is not a valid match to some available current or prior state of the originating instance, either do a normal resync or refresh the replicating instance to an appropriate state.

UPDSYNCINSTFILE

UPDSYNCINSTFILE, Error with instance file name specified in UPDATERESYNC qualifier

MUPIP Error: Issued by a Receiver Server when started with the -UPDATERESYNC qualifier in case of any error while processing the instance file specified as a value to this qualifier.

Action: An accompanying message, usually a GTM-I-TEXT message, describes the particular error situation. Take appropriate corrective action based on that.

USRIOINIT

USRIOINIT, User-defined device driver not successfully initialized

Run Time Error: This indicates that a user-implemented mnemonicspace driver image did not call the initialization entry in GT.M.

Action: Modify the user-supplied image to adhere to the mnemonicspace image-calling convention.

UPDREPLSTATEOFF

UPDREPLSTATEOFF, Error replicating global gggg as it maps to database xxxx which has replication turned OFF.

MUPIP Error: This indicates that the update process encountered an update record in the replication pipe that is destined for a non-replicated database file. GT.M does not allow such updates because they make the journal sequence numbers go out of sync between the replication primary and secondary. After issuing the message, the update process shuts down immediately. No more updates are processed on the secondary until replication is turned ON in the mentioned database file.

Action: Shut down the secondary instance including the receiver server, passive source server and any other helper processes accessing this instance. Turn replication ON in the mentioned database. Restart the secondary instance. The update process should now be able to process the update successfully.

VALTOOBIG

VALTOOBIG, xxxx is larger than the maximum of yyyy for a zzzz

GDE Error: This indicates that GDE has encountered a qualifier value that exceeds the maximum allowed. xxxx is the value that is too big. yyyy is the maximum value accepted. zzzz is the qualifier.

Action: Specify a qualifier value that is less than the maximum allowed.

VALTOOLONG

VALTOOLONG, xxxx exceeds the maximum length of yyyy for a zzzz

GDE Error: This indicates that GDE encountered a value that exceeds the maximum length allowed for an object name or qualifier. xxxx is the value that is too long. yyyy is the maximum length. zzzz is the object-name or qualifier.

Action: Specify an object-name or qualifier that is shorter than the maximum length.

VALTOOSMALL

VALTOOSMALL, xxxx is less than the minimum of yyyy for a zzzz

GDE Error: This indicates that GDE encountered a value that is less than the minimum allowed for a qualifier. xxxx is the value that is too small. yyyy is the minimum limit. zzzz is the object-name or qualifier.

Action: Specify a value that is larger than the minimum requirement for the object-name or qualifier.

VALUEBAD

VALUEBAD, xxxx is not a valid yyyy

GDE Error: This indicates that GDE encountered something other than the valid syntax element it was expecting. xxxx is the invalid element. yyyy is the valid element type.

Action: Specify a valid element. This error occurs if GDE is expecting an element (such as a file-specification, local qualifier, or number) but receives a value that does not evaluate to the expected element.

VALUEREQD

VALUEREQD, Qualifier xxxx requires a value

GDE Error: This indicates that GDE encountered a qualifier with a missing value. xxxx is the qualifier with a missing value.

Action: Supply a value for the qualifier.

VAREXPECTED

VAREXPECTED, Variable expected in this context

Compile Time Error: This indicates that GT.M expected a variable but encountered an invalid one.

Action: Look for proper variable names. This error is reported by commands and functions that require a variable argument such as SET and KILL and \$DATA() and \$QUERY().

VARRECBLKSZ

VARRECBLKSZ, Blocksize must be at least record size + 4 bytes

Run Time Error: This indicates that an OPEN command attempted to initialize a variable-length magnetic tape or sequential disk file with a RECORDSIZE of less than 5 bytes.

Action: Modify the routine to use a larger record size or use FIXED length records. The minimum variable RECORDSIZE reflects a single byte of data and 4 bytes of overhead in every variable-length record.

VERIFY

VERIFY, Verification xxxx

GDE Information: This indicates that an EXIT or VERIFY command caused GDE to verify the GDE mappings. xxxx is "OK" if the verification was successful or "BAD" if the verification failed.

Action: If the mappings are valid, GDE displays a confirmation message. GDE terminates the GDE session if the verification is OK on an EXIT command. If the mappings are not valid, review the accompanying message(s) for additional information.

VERMISMATCH

VERMISMATCH, Attempt to access xxxx with version yyyy, while already using zzzz

Run Time Error: This indicates that two different versions of GT.M attempted to access the same database at the same time.

Action: Update GT.M so that both systems use the same version of GT.M.

VERSION

VERSION, Version mismatch - This program must be recompiled

Run Time Error: This indicates that the process attempted to activate a routine that was compiled with an incompatible version of GT.M.

Action: Recompile the routine or use a compatible version of GT.M.

VIEWAMBIG

VIEWAMBIG, View parameter xxxx is ambiguous

Run Time Error: This indicates that the argument xxxx for a VIEW command or \$VIEW function is ambiguous due to insufficient characters.

Action: Add enough characters to make the argument unambiguous.

VIEWARGCNT

VIEWARGCNT, View parameter xxxx has inappropriate number of subparameters

Run-Time Error: The argument xxxx for a VIEW command or a \$VIEW function has too many or too few sub-arguments.

Action: Modify the argument so it has the proper number of sub-arguments.

VIEWCMD

VIEWCMD, View parameter is not valid with VIEW command

Run Time Error: This indicates that the VIEW command has an argument that is only valid with the \$VIEW function.

Action: Modify the argument.

VIEWFN

VIEWFN, View parameter is not valid with \$VIEW()

Run Time Error: This indicates that the \$VIEW function has an argument that is only valid with the VIEW command.

Action: Modify the argument.

VIEWGVN

VIEWGVN, Invalid global key name used with VIEW/\$VIEW(): xxxx

Run Time Error: This indicates that \$VIEW("REGION":gvn) failed because of an invalid global variable name xxxx in a sub-argument.

Action: Modify the sub-argument to be a valid M global variable name.

VIEWLVN

VIEWLVN, Invalid local variable name used with VIEW or \$VIEW(): vvvv

Run Time Error: This indicates the argument for a VIEW command or \$VIEW() function required a local variable name, but it (vvvv) was either missing or invalid.

Action: Correct the code in or investigate the logic to determine why the local variable in question is not in the expected state.

VIEWNOTFOUND

VIEWNOTFOUND, View parameter xxxx not valid

Run Time Error: This indicates that the VIEW command or \$VIEW() function has an invalid argument xxxx.

Action: Modify the argument.

VMSMEMORY

VMSMEMORY, Central memory exhausted - check page file quota and page file size

Run Time Error: This indicates that programs are requesting more memory than the system can provide with the current PAGEFILE settings.

Action: Adjusting the pagefile size and quota can increase memory efficiency. Refer to the Guide to VAX/OpenVMS Performance Management to determine appropriate pagefile settings.

VMSMEMORY2

VMSMEMORY2, Central storage exhausted during allocation of dynamic file descriptor with !UL bytes - check page file quota and page file size

Run Time Error: In OpenVMS, dynamic file descriptors used in external routine processing have their space allocated differently from other VMS memory. If one of these special allocations fails, this new version of the VMSMEMORY error is reported.

Action: Adjusting the pagefile size and quota can increase memory efficiency. Refer to the Guide to VAX/OpenVMS Performance Management to determine appropriate pagefile settings.

WAITDSKSPACE

WAITDSKSPACE, Process xxxx will wait aaaa seconds for necessary disk space to become available for yyyy

Run Time Error: This indicates that the database yyyy is full and cannot extend due to lack of space. All updates are suspended. Failure to address this message in the specified seconds will result in an OUTOFSPACE error.

Action: Immediately make enough space for the database to extend, and if the space is still not sufficient, stop all processes until more space is available. Examine your space management procedures and take actions to prevent any reoccurrence of this error.

WCBLOCKED

WCBLOCKED, Field xxxx is set by process yyyy at transaction number aaaa for database file zzzz

MUPIP Warning: This indicates that the current state of the global buffer cache necessitates the need for cache recovery. The next process to attempt obtaining the critical lock on the database will perform the recovery.

Action: Refer to the accompanying message(s).

WCERRNOTCHG

WCERRNOTCHG, Not all specified databases were changed

MUPIP Error: This indicates that SET could not modify database characteristics for at least one database.

Action: Review the accompanying message(s) for additional information.

WIDTHTOOSMALL

WIDTHTOOSMALL, WIDTH should be at least 2 when device ICHSET or OCHSET is UTF-8 or UTF-16.

Runtime Error: This error is issued whenever the ICHSET or OCHSET of the current terminal or file device is UTF-8 or UTF-16 and the WIDTH specified is 1. The minimum width allowed is 2 for such devices.

Action: Specify a width that is 2 or greater.

WCFAIL

WCFAIL, The database cache is corrupt

Run Time Error: This indicates that a database operation failed because the cache was damaged.

Action: Report this database cache error to the group responsible for database integrity at your operation.

WCWRNNOTCHG

WCWRNNOTCHG, Not all specified databases were changed

MUPIP Warning: This indicates that a SET command could not modify database characteristics for a particular database.

Action: Review the accompanying message(s) for additional information.

WILDCARD

WILDCARD, Wild cards are prohibited: xxxx

Run Time Error: This indicates that a \$ZROUTINES function that does not allow wild cards encountered one in xxxx.

Action: Replace the wild card with a specific directory name.

WRITEERROR

WRITEERROR, Cannot exit because of write failure. Reason for failure: xxxx.

GDE Error: This indicates that GDE was unable to write a new or revised global directory. xxxx text describes the failure.

Action: Review the text for additional information. If the failure is due to file permissions, it may be possible to correct it from the shell, accessed by means of the SPAWN command or another session. If the reason is not tractable, QUIT from GDE and re-invoke it after correcting the problem.

WRITERSTUCK

WRITERSTUCK, Buffer flush stuck waiting for [xxxx] concurrent writers to finish writing to database file aaaa

Run Time Error: This indicates that GT.M timed out after waiting nearly a minute for concurrent processes to complete flushing modified global buffers to the disk.

Action: This is usually symptomatic of a stressed I/O subsystem, where disk writes take a long time. System Administration might be warranted to improve the performance.

WRITEWAITPID

WRITEWAITPID, PID wwwwww waited mmmm minute(s) for PID hhhh to finish writing block bbbb in database file ffff

Run Time Warning: This operator log message indicates process wwwwww needed access to block bbbb in database file ffff, but had waited mmmm minutes for process hhhh to finish with that block. mmmm exceeds the expected design criteria for the processing by hhhh.

Action: Investigate the state and activities of process hhhh (possibly using the gtm_prodstuckexec facility); try to identify any coincident operating system, file system or storage sub-system issues that might contribute to this unexpected behavior.

XCVOIDRET

XCVOIDRET, Attempt to return a value from function xxxx, which is declared void in external call table yyyy

Run Time Error: This indicates that the specified function was typed as void in call table yyyy, but when the function was executed it attempted to return a value.

Action: Modify either the function to make it void, or the table to declare the proper type for the function.

XKILLCNTEXC

XKILLCNTEXC, Maximum number of arguments (xxxx) to exclusive kill exceeded

Compile Time Error: This indicates that a KILL command specified a list of exclusive arguments that require more temporary storage than is available. xxxx is the number of KILL arguments.

Action: Modify the routine to use an inclusive KILL or reduce the number of "protected" variables.

XTRNRETSTR

XTRNRETSTR, Return string from extended reference translation algorithm is NULL.

Run Time Error: This indicates that NULL was returned instead of a string from the user specified global variable name environment translation routine.

Action: Correct the user environment translation algorithm to return a string (which can be empty). Restart GT.M for the changes to become effective.

XTRNRETVAL

XTRNRETVAL, Length of return value from extended reference translation algorithm is out of bound

Run Time Error: This indicates that the return string from the user specified global variable name environment translation routine is of an invalid length (i.e. greater than 32767).

Action: Correct the user environment translation algorithm to return a string of valid length (in the 0-32767 range). Restart GT.M for the changes to become effective.

XTRNTRANSDLL

XTRNTRANSDLL, Error during extended reference environment translation. Please check the above message.

Run Time Error: This indicates that the external object (dynamically linked library in UNIX /sharable image in OpenVMS), which holds the global variable name environment translation routine, or the entry point gtm_env_translate in this object, is not accessible.

Action: Check if the value of the gtm_env_translate (UNIX: environment variable, OpenVMS: logical name) points to a valid object (UNIX: dll or OpenVMS: sharable image), which has the entry point gtm_env_translate.

XTRNTRANSERR

XTRNTRANSERR, Error attempting to generate an environment using an external algorithm.

Run Time Error: This indicates that the external environment translation routine returned an error.

Action: Check the external routine and the conditions it errors on. A supplementary TEXT message is printed if more information is provided by the external environment translation routine.

ZATTACHERR

ZATTACHERR, Error attaching to xxxx

Run Time Error: This indicates that a ZATTACH command failed because the argument specified a process xxxx that did not exist or was otherwise ineligible for attachment.

Action: Verify that the process exists within the job. Verify the spelling of the process name.

ZBREAKFAIL

ZBREAKFAIL, Could not set breakpoint at xxxx due to insufficient memory

Run Time Information: This indicates that a ZBREAK command failed to set a breakpoint at the entry reference xxxx due to lack of memory. No breakpoint will be effective when control reaches the given entry reference.

Action: Check for very large local variables. The memory requirement for ZBREAK is proportional to the size of the routine containing the entry reference. If the routine is large, divide the routine into smaller routines thereby reducing the memory requirement for ZBREAK. If appropriate, increase the memory quota for the user.

ZCALLTABLE

ZCALLTABLE, External call: Table format error

Run Time Error: This indicates that an external call or \$ZCALL() function encountered an error in the external call table.

Action: Modify the external call table. This message is usually displayed with another message.

ZCARGMSMTCH

ZCARGMSMTCH, External call: Actual argument count, xxxx is greater than formal argument count, yyyy

Run Time Error: This indicates that an external call or \$ZCALL() function specified xxxx arguments, which is more than the number yyyy, which is defined by the corresponding external call table entry.

Action: Ensure that the number of arguments in the external call matches the number of arguments in the external call table.

ZCCLNUPRTNMISNG

ZCCLNUPRTNMISNG, External call: Cleanup routine name missing. Cannot continue

Runtime error: This error occurs when there is no value assigned to the GTMSLIBEXIT (shared library exit handler keyword) in the external call table.

Action: Either remove the line that contains GTMSLIBEXIT= or provide a suitable function name in the external call table.

ZCCOLON

ZCCOLON, Colon expected but not found

Run Time Error: This indicates that there is a syntax error in the table designated by the accompanying message.

Action: Refer to the accompanying message(s) and correct the syntax error.

ZCCONMSMTCH

ZCCONMSMTCH, External call: Too many input arguments

Run Time Error: This indicates that an external call or \$ZCALL() function specifies more input arguments than the matching entry in the external call table.

Action: Ensure that the number of input arguments in the external call matches the number of input arguments in the external call table.

ZCCONVERT

ZCCONVERT, External call: error converting output argument

Run Time Error: This indicates that an external call or \$ZCALL() function failed because an output argument supplied by the external routine did not match the corresponding output description in the external call table.

Action: Change the external call table or the called routine so that they correspond.

ZCCSQRBR

ZCCSQRBR, Closing Square bracket expected

Compile Time Error: Brackets specify the beginnings and ends of pre-allocations. This error indicates that a pre-allocation was begun with a bracket, but was not closed.

Action: Correct the syntax by placing a closing square bracket at the end of the pre-allocation.

ZCCTENV

ZCCTENV, Environmental variable for external package xxxx not set

Compile Time Error: This indicates that the program made an external call, but could not find the external call table as specified in the UNIX environmental variable GTMXC.

Action: If this calls the default external call table, locate the external call table and specify the correct path in the UNIX environmental variable GTMXC. Otherwise, you can point to the package table in the environmental variable or specify the package name in the program GTMXC_[PACKAGE_NAME].

ZCCTNULLF

ZCCTNULLF, External call table contains no records: xxxx

Compile Time Error: This indicates that the program found the requested call table, but could not find external call data.

Action: Verify that the program calls the correct call table. If the table name is correct, add external call data.

ZCCTOPN

ZCCTOPN, Unable to open external call table: xxxx

Compile Time Error: This indicates that the program found the external call table but did not have permission to open it.

Action: Verify that the external call table and the user have appropriate permissions.

ZCENTNAME

ZCENTNAME, No entry found in external call table

Run Time Error: This indicates that the entry point, which the program is trying to access cannot be found.

Action: Create a corresponding entry point and match that link with the interface found in the C library.

ZCINPUTREQ

ZCINPUTREQ, External call: Required input argument missing

Run Time Error: This indicates that an external call or \$ZCALL function did not specify an input argument that is defined in the external call table as required.

Action: Change the external call table or the called routine so that they correspond.

ZCINVALIDKEYWORD

ZCINVALIDKEYWORD, External call: Invalid keyword found. Cannot continue. Invalid keyword encountered in the ext call config file.

Runtime error: This error occurs when the keyword for the shared library exit handler configuration is wrongly spelled. The correct keyword is "GTMSHLIBEXIT".

Action: Make sure that GTMSHLIBEXIT is correctly spelled in the external call table.

ZCMAXPARAM

ZCMAXPARAM, Exceeded maximum number of external call parameters

Run Time Error: GT.M allows a maximum of 31 parameters for a single external call. This error message indicates that the external call has exceeded this number.

Action: Break up the external call into two or more external calls or rewrite the program to pass a valid number of parameters.

ZCMLTSTATUS

ZCMLTSTATUS, Multiple entries of xc_status in a single entry in external call table

Run Time Error: This indicates that a call definition contains more than one XC_STATUS.

Action: Check parameters for multiple occurrences of TYPE XC_STATUS.

ZCNOPREALLOUTPAR

ZCNOPREALLOUTPAR, Parameter xxxx in external call yyyy.zzzz is an output only parameter requiring pre-allocation.

Run Time Error: This indicates that a pre-allocation value was not specified for the output only parameter xxxx in package yyyy, external call zzzz.

Action: Specify a pre-allocation value for the output only parameter xxxx. A package designation of "<DEFAULT>" indicates the default package rather than an actual package name.

ZCOPT0

ZCOPT0, External call: Qualifier OPTIONAL_0 can be used only with mechanisms REFERENCE or DESCRIPTOR

Run Time Error: This indicates that an external call or \$ZCALL function encountered an external call table input line that contained an OPTIONAL_0 QUALIFIER with a MECHANISM other than REFERENCE or DESCRIPTOR.

Action: Verify the type and spelling of the specified argument-passing mechanism in the external call table entry.

ZCPOSOVR

ZCPOSOVR, External call: Invalid overlapping of arguments in table position xxxx

Compile/Run Time Error: This indicates that an external call or \$ZCALL function encountered input and output descriptions for the same position, but one or both descriptions had invalid mechanisms or types.

Action: Modify the mechanism type and/or position it to remove the conflict.

ZCPREALLNUMEX

ZCPREALLNUMEX, Pre-allocation value should be a decimal number

Compile Time Error: This indicates that GT.M can only accept pre-allocation values between zero (0) to nine (9).

Action: Specify a pre-allocation value between zero and nine.

ZCPREALLVALINV

ZCPREALLVALINV, The pre-allocation value exceeded the maximum string length

Run Time Error: Pre-allocation value of an output parameter in the external call table is greater than the maximum allowed limit (i.e. maximum string size plus one byte for terminating NULL).

Action: In the external call table, modify the pre-allocation value to less than or equal to 1,048,577.

ZCPREALLVALPAR

ZCPREALLVALPAR, Pre-allocation allowed only for variables passed by reference

Compile Time Error: This indicates that the program specified a pre-allocation for a scalar variable passed by value.

Action: Determine if the program should use a pre-allocation or should be passed by value. If it uses a pre-allocation, the variable must be passed by reference. If the variable must be passed by value, the program cannot use a pre-allocation for that variable.

ZCRCALLNAME

ZCRCALLNAME, Routine name expected but not found

Run Time Error: This indicates that the compiler encountered an open parenthesis (indicating that parameters are listed, but found no function, which the parameters could modify.

Action: Check that the function name has been specified correctly. The function may not have been created, or the program uses an incorrect function name, or the function name may be missing.

ZCRPARAMNAME

ZCRPARAMNAME, Parameter name expected but not found

Run Time Error: This indicates that the program specified a direction and included a colon indicating that parameters would follow, but no parameters were found.

Action: The compiler indicates where the error occurred. Review the code to determine if parameters are needed or the colon should be removed.

ZCRTENOTF

ZCRTENOTF, External call routine xxxx not found

Compile/Run Time Error: This indicates that GT.M could not locate routine xxxx.

Action In OpenVMS, relink your image to include the missing routine. In UNIX, relink your external call descriptor image to include the missing routine.

ZCRTNTYP

ZCRTNTYP, Unknown return type

Compile Time Error: This indicates that the program specified an unrecognized return type. The compiler indicates where the invalid return type was found.

Action: Review the external calls table documentation for valid return types.

ZCSTATUS

ZCSTATUS, External call: Unsuccessful return status

Run Time Error: This indicates that an external call or \$ZCALL() function failed because the status returned by the external routine indicates that the routine did not execute successfully.

Action: Review the external routine and the arguments passed to it. Ensure that the routine is following the OpenVMS calling standard in returning the status.

ZCSTATUSRET

ZCSTATUSRET, External call returned error status

Run Time Error: This indicates that the called program may contain a logic error.

Action: Review accompanying messages for more information about the cause of this error.

ZCUNAVAIL

ZCUNAVAIL, Package, xxxx unavailable

Run Time Error: This indicates that the shared library may not be specified in the external call table or the path may be specified incorrectly.

Action: Verify that the external call table and the program point to the correct shared library and path.

ZCUNKMECH

ZCUNKMECH, External call: Unknown parameter-passing mechanism

Run Time Error: This indicates that an external call or \$ZCALL() function encountered an unsupported argument-passing MECHANISM in the external call table.

Action: Verify the spelling of the argument-passing MECHANISM values in the external call table.

ZCUNKQUAL

ZCUNKQUAL, External call: Unknown input qualifier

Run Time Error: This indicates that an external call or \$ZCALL() function encountered an unsupported input QUALIFIER in the external call table.

Action: Verify the spelling of the input QUALIFIER value in the external call table.

ZCUNKTYPE

ZCUNKTYPE, External call: Unknown argument type

Run Time Error: This indicates that an external call or \$ZCALL() function encountered an unsupported argument TYPE in the external call table.

Action: Verify the spelling of the argument TYPE in the external call table.

ZCUNTYPE

ZCUNTYPE, Unknown type entered

Run Time Error: This indicates that the program used an invalid external call parameter.

Action: Refer to the Programmer's Guide for valid external call parameters.

ZCUSRRTN

ZCUSRRTN, External call: Run-time error in user routine

Run Time Error: This indicates that an external call or \$ZCALL() function failed because the external routine produced an OpenVMS error condition and terminated abnormally.

Action: Review and debug the external routine.

ZCVECTORINDX

ZCVECTORINDX, Invalid Vector Index xxxx

Run Time Error: This indicates that GT.M can only accept positive values for vector index.

Action: Specify a valid vector index.

ZCWRONGDESC

ZCWRONGDESC, A string longer than 65535 is passed via 32-bit descriptor

Run Time Error: A string size greater than 65,535 byte was passed while the input mechanism is descriptor.

Action: Modify the external call table's input mechanism to be descriptor64 instead of descriptor and modify the called routine to handle the new descriptor type.

ZDATEBADDATE

ZDATEBADDATE, \$ZDATE() date argument dddd is less than -365 (the \$HOROLOG value for 01-JAN-1840) or greater than 364570088 (the \$HOROLOG value for 31-DEC-999999)

Run Time Error: The value of the date portion of the argument dddd to \$ZDATE() is outside the range of values the function handles.

Action: Examine the code that created the value of dddd for errors or revise the design to create dates within the range supported by \$ZDATE().

ZDATEBADTIME

ZDATEBADTIME, \$ZDATE() time argument **tttt** is less than 0 or greater than 86399 (the \$HOROLOG value for a second before midnight)

Run Time Error: The value of the time portion of the argument **tttt** to \$ZDATE() is outside the range of values the function handles.

Action: Examine the code that created the value of **tttt** for errors or revise the design to create times within the range supported by \$ZDATE().

ZDATEFMT

ZDATEFMT, \$ZDATE format string contains invalid character

Run Time Error: This indicates that a \$ZDATE() function specified a second argument that contains one or more invalid format characters.

Action: Verify the \$ZDATE() format.

ZDIROUTOFSYNC

ZDIROUTOFSYNC, \$ZDIRECTORY **xxxx** is not the same as its cached value **yyyy**

Run Time Warning: For performance purposes, GT.M caches the value of \$ZDIRECTORY when it is modified using SET command. This cached value is passed to external environment translation routine. GT.M issues ZDIROUTOFSYNC error when \$ZDIRECTORY is referenced and its cached value differs from the current working directory. This might happen if an external routine called from GT.M modifies the current working directory and the application does not modify \$ZDIRECTORY to the modified directory.

NOTE: ZSHOW of intrinsic special variables appends ->%GTM-W-ZDIROUTOFSYNC error to the text corresponding to the \$ZDIRECTORY if the out of sync condition is detected.

Action: Avoid changing the current working directory except with SET \$ZDIRECTORY, or ensure that SET \$ZDIRECTORY is used along with the other mechanisms.

ZEDFILSPEC

ZEDFILSPEC, Illegal ZEDIT file specification: **xxxx**

Run Time Error: This indicates that a ZEDIT command argument contains a file-specification that could not be parsed or is not a valid type for ZEDIT.

Action: Look for an attempt to edit a .o/.OBJ file.

ZFF2MANY

ZFF2MANY, Number of characters specified for ZFF deviceparameter (**xxxx**) is more than the allowed (**yyyy**)

Run Time Error: This indicates that the number of characters specified for ZFF deviceparameter for the socked device being OPENd (or USEd) exceeds the maximum allowed.

Action: Modify the string specified for ZFF to have a length of at most the maximum allowed.

ZFILENMTOOLONG

ZFILENMTOOLONG, xxxx is longer than 255 characters

Run Time Error: This indicates that a \$ZFILE() function argument exceeded 255 characters. xxxx is the length of the file-specification supplied.

Action: Modify the file-specification argument.

ZFILKEYBAD

ZFILKEYBAD, xxxx is not a legal keyword for \$ZFILE()

Run Time Error: This indicates that a \$ZFILE() function specified the invalid keyword xxxx.

Action: Review the routine for correct spelling and the validity of keywords.

ZFILNMBAD

ZFILNMBAD, xxxx is not a legal file name

Run Time Error: This indicates that a \$ZFILE() function argument supplied an invalid file-specification xxxx.

Action: Look for invalid characters or improper punctuation within the file-specification.

ZGBLDIRACC

ZGBLDIRACC, Cannot access global directory xxxx. Continuing with yyyy.

Run Time Error: This indicates that a SET of a \$ZGBLDIR or external global reference specified a Global Directory (xxxx) that does not exist or cannot be accessed due to permissions. GT.M retains the previous Global Directory (yyyy).

Action: Ensure that you specified the intended Global Directory. Use a host shell command to verify that the specified directory exists and has the protections required for the desired access.

ZGOCALLOUTIN

ZGOCALLOUTIN, ZGOTO level 0 with entry ref not valid when using call-ins

Run Time Error: Code invoked by a call-in contained a ZGOTO 0:entryref. The purpose of a ZGOTO 0:entryref is to "refresh" the GT.M routine context, but that action would invalidate the interface with the calling (in) code.

Action: Refactor the code invoked by the call-in to avoid or appropriately conditionalize the ZGOTO 0:entryref.

ZGOTOINVLVL

ZGOTOINVLVL, ZGOTO in a trigger running in mmmm cannot ZGOTO level LLLL

MUPIP Error: A ZGOTO command in trigger logic attempted to specify an inappropriate destination. Currently that is a ZGOTO in a trigger context with a target level of one (1) and an entryref. GT.M does not support such ZGOTO arguments in MUPIP because there is no context outside that of the trigger.

Action: Revise the trigger logic to only use ZGOTO with an entryref within the trigger context of trigger logic. Note that you can ZGOTO out of a trigger, but doing so in MUPIP terminates the MUPIP process. FIS recommends limiting the use of ZGOTO to debugging, error handling and testing. Use of ZGOTO in production code, even for error processing, should always be thoroughly tested.

ZGOTOINVLVL2

ZGOTOINVLVL2, ZGOTO 0:entryref is not valid on VMS (UNLINK is a UNIX only feature)

Run Time Error: Because the GT.M OpenVMS environment includes links performed by the OpenVMS Linker, GT.M does not support this construct on OpenVMS.

Action: Refactor the code to avoid or appropriately conditionalize the ZGOTO 0:entryref.

ZGOTOLTZERO

ZGOTOLTZERO, Cannot ZGOTO a level less than zero

Run Time Error: This indicates that a ZGOTO specified a negative level in its argument.

Action: Ensure that the ZGOTO level is between zero (0) and the current \$ZLEVEL.

ZGOTOTOOBIG

ZGOTOTOOBIG, Cannot ZGOTO a level greater than present level

Run Time Error: This indicates that ZGOTO command specified a level greater than the current stack depth.

Action: Check the source of the level. The level argument of a ZGOTO indicates the depth the stack is to be after the ZGOTO, which cannot be greater than the current stack depth. The only way to increase stack depth is by performing DOs or XECUTEs.

ZINTDIRECT

ZINTDIRECT , Attempt to enter direct mode from \$ZINTERRUPT

Run Time Error: A \$ZINTERRUPT routine cannot break to direct mode if the current IO device is the same as \$PRINCIPAL.

Action: Modify the \$ZINTERRUPT routine to not break to direct mode.

ZINTRECURSEIO

ZINTRECURSEIO, Attempt to do IO to the active device in \$ZINTERRUPT

Run Time Error: A \$ZINTERRUPT routine cannot perform I/O to the current IO device if it was active when the interrupt was recognized and the device is a terminal, socket device, FIFO, or PIPE.

Action: Modify the \$ZINTERRUPT routine to not perform I/O to the active device.

ZLINKFILE

ZLINKFILE, Error while ZLINKing "xxxx"

Run Time Error: This indicates that ZLINK command failed while trying to include routine xxxx in the image.

Action: Use host shell commands to ensure that the file to be ZLINKed is in the proper directory and has the appropriate protection. Review the accompanying message(s) for additional information.

ZLKIDBADARG

ZLKIDBADARG, The tvexpr must be FALSE if last ZLKID not found

Run Time Error: This indicates that a \$ZLKID() function specified an argument that was TRUE when no active OpenVMS lock scan was in progress.

Action: Ensure that \$ZLKID() is initialized with a FALSE truth-valued expression. \$ZLKID() cannot be invoked with a TRUE truth-valued expression unless the last \$ZLKID() returned a lock. If the function has not been previously invoked, it must be first invoked with a FALSE truth-valued expression.

ZLMODULE

ZLMODULE, Object file name does not match module name: xxxx

Run Time Error: This indicates that GT.M did not perform a ZLINK or an auto-ZLINK because the name of the object file specified with the ZLINK xxxx is not the same as the name of the source file.

Action: Look for and correct any typographical errors in the object file specified for the ZLINK, or assign the same name to the source and object.

ZLNOOBJECT

ZLNOOBJECT, No object module was produced

Run Time Error: This indicates that a run-time compile specified a NOOBJECT qualifier. This can be accomplished with a ZLINK qualifier or by setting \$ZCOMPILE to a qualifier string that GT.M uses for auto-ZLINKs or ZLINKs with no qualifiers.

Action: Remove the NOOBJECT qualifier.

ZPARSETYPE

ZPARSETYPE, Illegal TYPE argument to \$ZPARSE(): xxxx

Run Time Error: This indicates that a \$ZPARSE() function specified a type argument xxxx that was not a null string, SYNTAX_ONLY, or NO_CONCEAL.

Action: Verify the spelling of the keyword.

ZPARSFLDBAD

ZPARSFLDBAD, Illegal \$ZPARSE() field parameter: xxxx

Run Time Error: This indicates that a \$ZPARSE() function specified a field argument xxxx that was not NODE, DEVICE, DIRECTORY, NAME, TYPE, or VERSION.

Action: Verify that the keyword is spelled correctly.

ZPIDBADARG

ZPIDBADARG, The tvexpr must be FALSE if last \$ZPID() not found

Run Time Error: This indicates that a \$ZPID() function specified an argument that was TRUE when no active PID scan was in progress.

Action: Review the routine to ensure that \$ZPID() is first initialized with a FALSE truth-valued expression. \$ZPID() cannot be invoked with a TRUE truth-valued expression unless the last \$ZPID() returned a PID. If it has not been previously invoked, it must be first invoked with a FALSE truth-valued expression.

ZPRIVARGBAD

ZPRIVARGBAD, xxxx is not a legal privilege for \$ZPRIV()

Run Time Error: This indicates that a \$ZPRIV() function specified an argument xxxx that contains a keyword that is not a valid OpenVMS privilege.

Action: Review for spelling errors or invalid privileges.

ZPRIVSYNTAXERR

ZPRIVSYNTAXERR, Privilege string cannot end with a comma

Run Time Error: This indicates that a \$ZPRIV() function specified a list of privileges that ended with a comma.

Action: Look for a missing keyword or an extra comma. The form for a \$ZPRIV() argument is a string of OpenVMS privilege keywords separated by commas. The list must begin and end with a keyword.

ZPRTLABNOTFND

ZPRTLABNOTFND, Label not found in routine

Run Time Error: This indicates that a ZPRINT command specified a label that could not be found in the routine.

Action: Verify the spelling of the label. Ensure that the current version of the program has the label by ZPRINTing the entire routine.

ZROSYNTAX

ZROSYNTAX, \$ZROUTINES syntax error: xxxx

Run Time Error: This indicates that a \$ZROUTINES related action encountered syntax error xxxx.

Action: Modify the UNIX environment variable gtmroutines/OpenVMS logical name GTM\$ROUTINES or the expression being SET into \$ZROUTINES.

ZSETPRVARGBAD

ZSETPRVARGBAD, xxxx is not a legal privilege for \$ZSETPRIV()

Run Time Error: This indicates that a \$ZSETPRIV() function specified an argument xxxx that contains a keyword that is not a valid OpenVMS privilege.

Action: Look for spelling errors or invalid privileges.

ZSETPRVSYNTAX

ZSETPRVSYNTAX, \$ZSETPRIV() privileges string cannot end with a comma

Run Time Error: This indicates that \$ZSETPRIV() function specified a list of privileges that ended with a comma.

Action: Look for a missing keyword or an extra comma. The form for a \$ZSETPRIV() argument is a string of OpenVMS privilege keywords separated by commas. The list must begin and end with a keyword.

ZSHOWBADFUNC

ZSHOWBADFUNC, An illegal function was specified for ZSHOW

Run Time Error: This indicates that ZSHOW argument code specified an invalid action.

Action: Modify the argument to use a valid code.

ZSHOWGLOSMALL

ZSHOWGLOSMALL, Global output variable is too small for ZSHOW output

Run Time Error: This indicates that SHOW could not place data in a target global variable because the record size for the database file that holds the region containing the variable was too small to accommodate the ZSHOW output.

Action: Use a global output variable in a less restrictive region, or reorganize the database and increase the value of the qualifier RECORD_SIZE= in the Global Directory.

ZSRCHSTRMCT

ZSRCHSTRMCT, Search stream identifier out of range

Run Time Error: This indicates that \$ZSEARCH() function specified a stream less than 1 or greater than 255.

Action: Change the stream value to be within the range of 1-255.

ZSTEPARG

ZSTEPARG, ZSTEP argument expected

Compile Time Error: This indicates that ZSTEP command did not specify an argument or a <SP> to hold the place of the missing argument.

Action: Modify the ZSTEP command so it has an argument or a trailing double space.

ZTRIGINVACT

ZTRIGINVACT, Missing or invalid subcode (first) parameter given to \$ZTRIGGER()

Run Time Trigger Error: The first argument to \$ZTRIGGER() is required to specify its mode of action.

Action: for the first argument of \$ZTRIGGER() use an expression that evaluates to "FILE", "ITEM" or "SELECT".

ZTRIGNOTP ●

Last used version: **V5.4-001**

ZTRIGNOTP, \$ZTRIGGER() cannot use update subcodes FILE or ITEM when a TP transaction is in progress (\$TLEVEL greater than zero)

Run Time Trigger Error: A FILE or ITEM operation of \$ZTRIGGER() failed because it attempted to apply a trigger definition inside an ongoing transaction. Both FILE and ITEM operations of \$ZTRIGGER initiate an implicit transaction to achieve trigger update atomicity, therefore, GT.M does not allow nesting them inside another transaction that potentially might use the very triggers \$ZTRIGGER() is attempting to update.

Action: Move all FILE or ITEM operations of \$ZTRIGGER() outside the scope of any open transaction.

ZTRIGNOTRW

ZTRIGNOTRW, ZTRIGGER cannot operate on read-only region rrrr

Runtime Error: This error occurs when ZTRIGGER attempts to write to read-only region rrrr.

Action: Check for appropriate global directory mapping and appropriate permissions on the database file mapped to region rrrr.

ZTWORMHOLE2BIG

ZTWORMHOLE2BIG, String length of LLLL bytes exceeds maximum length of mmmm bytes for \$ZTWORMHOLE

Run Time Trigger Error: GT.M limits \$ZTWORMHOLE length to mmmm bytes and the application attempted to use LLLL bytes.

Action: Restrict the size of the string stored in \$ZTWORMHOLE to mmmm bytes. Ensure that \$ZTWORMHOLE only holds the information that the application needs during trigger execution. If necessary, reorganize the logic to reduce the amount of local context needed during trigger execution, possibly by using global variables.

ZWRSPONE

ZWRSPONE, Subscript patterns in ZWRITE are atomic; Invalid delimiter

Compile Time Error: This indicates that ZWRITE specification contained a pattern match that held or terminated with a punctuation character that was not within a string literal.

Action: Look for missing quotes or typographical errors and make any corrections that are necessary.

Appendix A. Standardized Error Codes

This following table describes the error codes defined in the ISO/IEC-11756:1999 standard.

Error Code	Translation
M1	Naked indicator undefined
M2	Invalid combination with P_fncodatom_
M3	\$RANDOM seed less than 1
M4	No true condition in \$SELECT
M5	_lineref_ less than zero
M6	Undefined _lvn_
M7	Undefined _gvn_
M8	Undefined _svn_
M9	Divide by zero
M10	Invalid pattern match range
M11	No parameters passed
M12	Invalid _lineref_ (negative offset)
M13	Invalid _lineref_ (label not found)
M14	_line_ level not 1
M15	Undefined index variable
M16	Argumented QUIT not allowed
M17	Argumented QUIT required
M18	Fixed length READ not greater than zero
M19	Cannot copy a tree or subtree into itself
M20	_line_ must have _formallist_
M21	Algorithm specification invalid
M22	SET or KILL to ^\$GLOBAL when data in global
M23	SET or KILL to ^\$JOB for non-existent job number
M24	Change to collation algorithm while subscribed local
M25	Attempt to modify currently executing routine
M26	Non-existent _environment_
M27	Attempt to rollback a transaction that is not restartable

Standardized Error Codes

M28	Mathematical function, parameter out of range
M29	SET or KILL on <code>_ssvn_</code> not allowed by implementation
M30	Reference to <code>_glvn_</code> with different collating sequence within a collating algorithm
M31	<code>_controlmnemonic_</code> used in user-defined <code>_mnemonicspace_</code> selected
M32	<code>_controlmnemonic_</code> used in user-defined <code>_mnemonicspace_</code> which has no associated line
M33	SET or KILL to <code>^\$ROUTINE</code> when <code>_routine_</code> exists
M35	Device does not support <code>_mnemonicspace_</code>
M36	Incompatible <code>_mnemonicspace_s</code>
M37	READ from device identified by the empty string
M38	Invalid <code>_ssvn_</code> subscript
M39	Invalid <code>\$NAME</code> argument
M40	Call-by-reference in <code>JOB _actual_</code>
M41	Invalid LOCK argument within a TRANSACTION
M42	Invalid QUIT within a TRANSACTION
M43	Invalid range (<code>\$X</code> , <code>\$Y</code>)
M44	Invalid <code>_command_</code> outside of a TRANSACTION
M45	Invalid GOTO reference
M57	More than one defining occurrence of label in routine
M58	Too few formal parameters

Appendix B. Reference Implementation Error messages

The error messages listed in this appendix are generated by the reference plug-in. If you change the plug-in, the messages will be those generated by your plug-in. Note that the messages from the plug-in (incorrectly) use the term "DB keys file" for the master key file.

Cannot find DB keys file <path>

Plugin error: The plugin cannot find the master key file.

Action: Set the \$gtm_dbkeys environment variable to point to the correct master key file.

Cannot find MUMPS executable in <path>

Plugin error: The plugin cannot find the MUMPS executable.

Action: Set the \$gtm_dist environment variable to the directory containing MUMPS executable. Verify proper permissions for directory path and file.

Cannot open DB keys file - <path>

Plugin error: The plugin cannot open the master key file.

Action: Verify the master key file exists and there are appropriate authorizations on the directory path and master key file.

DB keys file of unknown file type : <path>

Plugin error: The plugin reports that the master key file is not the proper type file.

Action: Point the gtm_dbkeys environment variable to an appropriately formatted master key file.

Database file <path> missing in DB keys file or does not exist

Plugin error: The plugin reports that the master key file does not contain a valid entry pointing to the database file.

Action: Create an entry in the master key file for the specified database file, verify that the database file exists and appropriate authorizations exist on the directory and database file name.

Database file <path> not found

Plugin error: The plugin is unable to find the specified database file.

Action: Verify that the database file exists, the corresponding entry in the master key file points to the database file, and appropriate authorizations exist in the directory path and the database file.

Encryption handle corrupted

Plugin error: The plugin detected an internal error.

Action: This error indicates that there is a communication error between GT.M and the gtmcrypt plug-in. Replace the process with undamaged one. Report the entire incident context to your GT.M support channel.

Encryption key file <path> not found

Plugin error: The plugin was not able to find the key file on the specified path.

Action: Verify that the master key file entry for this key file points to the correct path. Verify that the key file itself exists. Verify proper authorizations on directory path and file.

Encryption library has not been initialized

Plugin error: A gtmcrypt function was called before gtmcrypt_init().

Action: Call gtmcrypt_init() before calling any other encryption functions.

Environment variable <environment_variable> not set

Plugin error: An environment variable needed by the plugin was not set.

Action: Set the environment variable <environment_variable> to an appropriate value.

Environment variable gtm_dbkeys set to empty string

Plugin error: The \$gtm_dbkeys environment variable was set to the empty string.

Recovery Action: Set \$gtm_dbkeys to point to the master key file.

Environment variable gtm_dbkeys undefined. Cannot find <path>/.>gtm_dbkeys

Plugin error: The plugin was unable to locate the master key file.

Action: Place the master key file in the users home directory or point the gtm_dbkeys environment variable to the master key file.

Environment variable gtm_passwd set to empty string. Password prompting not allowed for utilities

Plugin error: The plugin detected that it needed the obfuscated password but the \$gtm_passwd environment variable was set to the empty string.

Action: Use maskpass to set \$gtm_passwd to the obfuscated password prior to invoking MUIP or DSE, or wrap the utility invocation with a MUMPS process which will prompt for the password and set the obfuscated password.

Error initializing GpgME: <reason_for_error>/<specific_Gpg_ME_error>

Plugin error: libgpgme reported an error to the plugin.

Action: Consult GpgME documentation for the specific error message.

Error parsing database key file. At line <line_number>: No matching 'dat' entry found in <contents_of_line>

Plugin error: The plugin was unable to find a matching "dat" entry for a "key" entry in the master key file.

Action: Verify that each "key" entry has a corresponding "dat" entry.

Error parsing database key file. At line <line_number>: <line_contents> does not start with 'dat' / 'key'

Plugin error: The plugin detected that the master key file was not properly formatted.

Action: Verify entries in the master key file start with "dat" or "key".

Error parsing database key file. At line <line_number>: No matching 'key' entry found in <contents_of_line>

Plugin error: The plugin was not able to find a "key" entry for a "dat" entry.

Action: Verify the database file exists, that the corresponding entry in the master key file points to the database file, that appropriate authorizations exist on the directory path and the database file, and that each "dat" entry has a corresponding "key" entry.

Incorrect password

Plugin error: The plugin detected that the correct private key password was not supplied.

Action: Provide the correct password.

libgcrypt version mismatch. Expected <expected_version>, found <found_version>

Plugin error: The plugin could not locate an appropriate libgcrypt library version.

Action: Verify the <expected_version> is installed and in the library search path.

Matching encryption key <hash> not found in database key file

Plugin error: The plugin was not able to find a needed database file key.

Action: Add an entry for this encryption key. If needed, use DSE DUMP -FILE -ALL on all of the database files to find the database file that matches the <hash>. With extracts and backups, multiple database files may have contributed encrypted records.

No entries found in DB keys file

Plugin error: The plugin was unable to find any entries in the master key file.

Action: Add entries to the master key file.

Symmetric key <path> found to be empty

Plugin error: The plugin was unable to find a valid encrypted key in the specified file.

Action: Create a valid key file.