SOFTWARE RELEASE DOCUMENT (SOFTDOC)

Product: Shadowbase

Release: Gravic Version 6.220
                    HP NonStop Shadowbase:
                        T1122-AAC (SB Repl/Guardian)

Release Date: April 15, 2016

Copyright Notice: Copyright Gravic, Inc. 1995 – 2016 (www.gravic.com)

File Name: IPM6220.doc

NOTE: Release 6.220 is a full release of HP NonStop Shadowbase and includes a
            complete set of product files for all HP NonStop components. This softdoc
            applies to the HP NonStop Guardian replication components. Other
            softdocs document the releases for the HP NonStop Shadowbase Audit
            Reader, HP NonStop Compare, and the HP NonStop OSS Replication
            components.

NOTE: This softdoc covers new features and corrected problems for Shadowbase
            for H, J, and L series Guardian NonStop operating system versions. This
            softdoc is available in an Adobe PDF file (.PDF). Softdoc files are named
            IPMnnnn (where nnnn is the Shadowbase version number).

Table of Contents

Special Notes for Version 6.220

1) Due to a known problem outstanding, you should not configure a QMGR process for a Collector if it has one or more DBS's that are processing DDL operations with coordination and one or more Consumers doing TRACKTX processing. TIDFILECONSTANT ON may be used in place of TRACKTX processing.

2) If you utilize the SOLV loader related to a DBS object that has ALLPARTITIONS set ON, you should set the SOLV TARGETFNAME parameter to the DBS SOURCEFILE value. TARGETFNAME consists of the node name, volume and subvolume (no filename) of the SOURCEFILE.

3) As of version 4.091, the CONTROLFILE record size, associated with the CONS CONTROLFILE parameter, has been increased. You must purge any existing CONTROLFILE's before restarting Shadowbase if you are upgrading from a pre-4.091 version of Shadowbase.

4) Enscribe and SQL/MP CREATE, PURGE, PURGEDATA, and ALTER events are now processed directly by the Consumer. If you have the DBS setting for these events enabled (ON) and have user exit code that processes these events, you must change the code to call the SBSETIGNORE API function before returning. Otherwise, the Consumer will attempt to process the events again and this could result in errors.

5) For those that have user exits, you must use the 6.220 USRXLIBO (non-native version), or USRXLIBN or USRXLIBX (native versions) API library included in the distribution subvolume. If you have copies of the USRXLIBO, USRXLIBN, or USRXLIBX file in other subvolumes, make sure you replace them with the current version from the distribution subvolume. You should not use earlier versions of USRXLIBO, USRXLIBN, or USRXLIBX with Shadowbase version 6.220.

6) Control file changes made in Version 5.001 require that you perform an initial START AUD COLD if you are upgrading from a release prior to 5.001.

7) For bi-directional replication environments, due to enhancements made to the internal algorithms within the COLL and CONS, it is recommended that both (or all) nodes be upgraded at the same time. Shut down both, do your upgrade, and then bring each node's Shadowbase environment back up. Contact Shadowbase Support if you have bi-directional environments and you cannot upgrade them both at the same time.

8) For SQL/MX IEEE 32-bit floats, 8 digits of precision in scientific notation is the SQL/MX Consumer (new AUDCONXN and AUDCONXR programs delivered in 5.000) assigned value in the generated SQL statement data. The Tandem format 32-bit float will continue to use 6 digits of precision in the non-IEEE Consumer (AUDCONS, AUDCONSN, and AUDCONSR programs).

Additionally, for DDL related to Enscribe files, the Consumer assumes that all floats are in Tandem format and not IEEE format. A user exit is required if they are actually in IEEE format. Otherwise, a generated SQL statement could have an incorrect number.

9) If you are developing user exit code to use with the SQL/MX Consumer, make certain that you compile your code for IEEE floating point format. Only native code can be used with the SQL/MX Consumer. Non-native user exits are not supported.

10) You must install Shadowbase on a physical volume if SQL/MX tables will be replicated. You should not install Shadowbase on an SMF virtual drive, otherwise errors will result during the install process.

11) If SQL/MX tables will be replicated with Shadowbase, NonStop Shadowbase must be installed on the source system(s) and Shadowbase Open Server must be installed on the target system(s) as Shadowbase's SQL/MX target support is implemented in the OSS environment. Shadowbase Open Server for OSS, version 5.000 or higher, is required.

12) Support for SQL/MX has the following limitations:

- Tables with Relational Integrity (RI) constraints. Ordering of I/Os within a transaction for multiple tables cannot be guaranteed in the audit trail due to parallel nature of transaction processing on NonStop systems. For example, consider the case where there are parent and child tables, with a constraint to force a corresponding parent row to exist when a child row is added. If you add both the parent and child row under the same transaction when a child is added with a new parent, Shadowbase may not be able to add those rows on the target. Depending on which disk flushed to the audit trail first, the audit trail may return the rows with the child row insert first and the parent row insert second. In this case, Shadowbase will fail to insert either row due to the RI constraint being violated on replay.

- Tables with large rows and Cascading Deletes configured. We have seen that TMF may save updates of large rows as a delete event of the old row followed by an insert of the revised row. Shadowbase will replay this event as a delete followed by an insert as well. If cascading deletes are configured, the delete may cause all child rows to be deleted on the target. Note that this behavior may also run afoul of RI constraints noted above, if a child row is inserted between the delete and insert of the parent row.

13) Shadowbase NonStop Version 6.220 is only supported on H06.25 and later releases (or the equivalent J series RVUs) for NonStop Integrity Systems. For HP NonStop Integrity L Series, Shadowbase NonStop Version 6.220 is supported for L15.02 and later.

14) If you are upgrading from a release prior to Version 6.100, note that the Shadowbase EMS subsystem owner and version (the EMS SSID token) changed in Version 6.100. The subsystem owner changed from ITI to GRAVIC, and the subsystem version is now

composed of VNN, where NN are the two most significant digits of the Shadowbase release version id. As a result, the subsystem version has changed from D42 to V61. Any EMS filters using these attributes for filtering must likewise be updated. Alternatively, the SBEMSOLDSSID TACL parameter can be enabled to have Shadowbase use the old values for the subsystem owner and version.

15) The Shadowbase NonStop release is now included in a single self-extracting file, SBPAK. The files that were previously included separately in the self-extracting files SBSLPAK and SBLDPAK are now included in SBPAK. See **ERROR! REFERENCE SOURCE NOT FOUND.** for updated installation instructions.

16) Errors 8401, 8405, 8406, 8410, 8411, 8412, 8417, 8418, 8423, 8424, 8425, 8426, 8427, and 8428 will no longer be logged and ignored by the Consumer by default. The user exit stubs included with Shadowbase, USRXC, USRXCOB, and USRXCOBA, have been modified to remove the call to SBSETEXCEPTIONCODES. The code to setup the structure and make the call remains as a comment in the stub files. To use the structure and make the call to SBSETEXCEPTIONCODES, uncomment the code from the stub file being used.

17) Some customers experienced a rare issue which caused the ARLIB2 library to hang when there was no activity on one of the auxiliary audit trails. The Collector now links with the ALV version of the ARLIB2 library which incorporates a fix for this issue.

18) When using a multi-ported DOC Writer, it is recommended as a best practice not to use Queue Managers for the related threads in the Shadowbase NonStop configuration. Furthermore, it is recommended not to use the SUSPENDUPD/RESUMEUPD commands when replicating to a multi-ported DOC Writer.

Changes in Release 6.220 (since Version 6.200)

New Features

1) IPC and STP message sizes have been increased, adding support for increased record, row, and key size limits for replication. See INCREASED RECORD, ROW, AND KEY SIZE LIMITS for details.

2) An increased maximum number of partitions for Enhanced Key-Sequenced (EKS) files is now supported. The new maximum is the current maximum allowed for EKS files of 128 partitions. See INCREASED MAXIMUM NUMBER OF PARTITIONS FOR ENHANCED KEY-SEQUENCED (EKS) FILES for further information.

3) A new DBS parameter, USEREXITPARAM, has been added to allow for specifying a parameter string for user exit parameters. See NEW AND MODIFIED SET DBS PARAMETERS for details.

4) A new utility, SBQFILE, has been added to the release package. This utility allows the user to convert between queue files and key sequenced files and allows for the implementation of unidirectional queue file replication. See APPENDIX A – QUEUE FILE REPLICATION WITH SBQFILE for details. Note, this section was added to the HPE NonStop Shadowbase Installation and Planning Manual.

5) A new user exit API function, SBGETSRCFILEINFO, has been added for retrieving information about the source file. See SBGETSRCFILEINFO for details.

6) Views that include aliases, define names, and joins are now supported for use with the DBS VIEWNAME parameter. Note, however, that a define name cannot be specified for the VIEWNAME parameter value itself (as with other AUDCOM paramaters).

Problems Fixed

1) Some of the HPE Shadowbase program files for TNS-X were missing VPROC information. The missing VPROCs have been added.

2) The Queue Manager did not validate message sequence numbers received over Expand. This has been corrected, and message sequence numbers are checked for duplicate or out of order messages received over Expand.

3) The Queue Manager was not able to handle queue files configured greater than 4GB in size. The Queue Manager would abend with an error 45 when exceeding this limit due to 32 bit positioning being used. This issue has been corrected, and larger file sizes are now supported correctly.

4) Transactions were incorrectly age cleared by the Queue Manager when the connected Consumer was stopped with transactions active. This occurred if the Queue Manager

remained running for twelve hours after this point, and the Consumer remained stopped. This has been corrected, and transactions will no longer be age cleared in this scenario.

5) When replicating out to HPE Shadowbase for Other Servers across multiple threads to a multi-ported DOC Writer using SBCOLLMULTIENDS, commit events were not being sent for transactions when they should have been if a Queue Manager object was configured for the thread. As a result, the expected number of end events were not received by the DOC Writer, preventing the transaction from being committed. End events are now properly sent down threads with Queue Manager objects configured. While Queue Manager objects can be configured for threads feeding a multi-ported DOC Writer, however, it is still recommended as a best practice not to use Queue Managers with multi-ported DOC Writers.

6) The AUDCOM help text has been restored to alphabetical order. Some of the subcommand options for the SET command, SET QMGR and SET SOLVMGR, were missing from the help text, and have been added for consistency.

7) When using the Queue Manager to queue data while the Consumer is stopped, the Queue Manager incorrectly calculated that a cold start was occurring on a restart if no data had been sent to the Consumer. As a result, the queue files would be purged. This will no longer occur, and the data in the queue files will be preserved even if no data has yet been sent to the Consumer.

8) For a DBS configured with MAPFROMSOURCE ON and PURGES ON, if the first event replicated was a purge event, the correct file was not purged. Instead of purging the file on the target system corresponding to the source file which was purged, the target system template file would be purged. The correct file will now be purged even if a purge event is the first event replicated.

9) In some cases, SBCOLL sent commit events down threads with no data for the transaction even when SBCOLLMULTIENDS was sent to 0. This resulted in extra empty transactions being sent to the target.

10) There was a typo in the LINKUSMX script, where it said it was linking AUDCONSN instead of AUDCONXN. This has been corrected to properly convey that it is AUDCONXN that is being linked.

11) AUDMON performs a check for the setting of the INSERTNOTFOUND DBS parameter when audit compression is enabled for the source file. For replication from HPE NonStop Shadowbase to HPE Shadowbase for Other Servers, this setting has no impact. Despite this, it was still being checked and could prevent replication from starting if INSERTNOTFOUND was set on. This check has been removed for this scenario.

12) The Collector would intermittently fail to enter into the SUSPENDUPD state following a SUSPENDUPD command. The workaround was to reissue the SUSPENDUPD command. This issue has now been resolved.

13) RESET DBS ENTRYSEQEXACT did not work, and resulted in an invalid parameter for command error.

14) When using SBCOLL for SQL/MX replication, a STATUS AUD command returned incorrect status information regarding the number of consumers in a SUSPENDUPD state. This occurred if Queue Manager objects were configured and a SUSPENDUPD command had been entered for the related thread. Both the Queue Manager and Consumer objects were counted in the number of Consumers in a SUSPENDUPD state.

15) An issue existed with SOLV loading for SQL/MP tables defined with a clustering key. When a clustering key existed for a table, SOLV was skipping the SYSKEY values. This issue has now been corrected.

16) A performance bug was discovered in the Consumer RESOLVELOCKS processing. RESOLVELOCKS processing is used for resolving both lock contention issues and lock resource issues. Lock resource issues do not require waiting after performing the RESOLVELOCKS precommit; lock contention issues must wait the configured time. The RESOLVELOCKS processing was delaying in either case. This extra delay has been removed for lock resource errors.

17) A couple of bugs existed in the Consumer schema handling cleanup logic. Under certain circumstances when schema swapping occurred or following replication of a purge event, the Consumer would abend on a logic error in the schema handling cleanup logic. These issues have been fixed, and the Consumer will no longer abend.

18) In a SQL/MX environment, attempting to replicate events for records or rows exceeding the maximum size supported by HPE Shadowbase replication could result in events being skipped, the Consumer abending, or a combination of the two. The new long record and row support in this release allows for successful replication of these events. See the new features section for details on long record and row support. Additional validation logic has also been added to the Collector to detect and prevent attempting to replicate events exceeding the maximum IPC size.

19) When displaying restart information in EMS messages, SBCOLL and Queue Manager objects used the MAT audit trail sequence number when formatting the AUX file name. The correct AUX audit trail sequence number is now used in formatting the AUX file name.

20) The LINKUSMX script was missing ZCLIDLL needed for the second linker step. This resulted in a number of undefined externals when attempting to perform the linking. The script has been updated to add the missing DLL.

21) SBCOLL was not starting reading in the proper position when there is no restart file when using ADTSTARTEOF ON in a bi-directional environment or when using ADTSTARTTIME. The start point was being flagged as reached before it should have been, resulting in SBCOLL starting to process events prior to the specified time. Both issues have been corrected, so that SBCOLL begins processing events read from the expected position in audit.

22) When replicating a SQL/MX table with a SYSKEY as part of a clustering key, the Consumer did not generate the full correct where clause. The where clause generated consisted only of the SYSKEY column in this case. This has been corrected to generate the proper where clause based on the full clustering key.

23) Dynamically adding and capturing a DBS on a thread with a Queue Manager while the HPE Shadowbase environment was running caused the Collector process to abend. This will no longer cause the Collector to abend.

24) In a SQL/MX bi-direction replication environment, the Collector process was unresponsive while pre-reading audit when starting up without a restart file. During this period, responses to commands which query the Collector for data, such as STATS COLL, were held up until the audit pre-read completed. The behavior of SBCOLL has been changed to respond to such messages during audit pre-read processing.

25) When performing a SOLV load using SOLVCONSISTENCYLEVEL TRANSACTION in a HPE Shadowbase configuration using Queue Managers, the SOLV load would timeout. This issue has been corrected.

26) For replication of audit compressed updates with user exit processing performing "puts" for columns present in the table but not in the compressed update, the user exit put columns were not being properly added to the statement. The workaround for this issue was to set the TACL parameter SBCONSORDERCOLS to 0. The user exit put columns will now be properly added to the statement without needing to set this parameter.

27) The ADTEFLDUPDBUSY measure counter was using the wrong name. ADTEFLDUPDUSED was being used instead (the same as the measure counter of that name). AFTEFLDUPDBUSY now uses the correct name.

28) PIC and PICTURE were not treated as reserved words for SQL/MX column names, and were not being surrounded by double quotes in the generated statements. PIC and PICTURE will now be surrounded by double quotes if used as column names.

29) When replicating DDL events with coordination enabled and TURBOMODE OFF, AUDCOLLN would abend on a duplicate transid on AVL insert error after replicating the event. The workaround was to enable the COLL TURBOMODE

parameter, or to disable DDL coordination. AUDCOLLN will no longer abend in this scenario even if TURBOMODE is OFF.

30) The Consumer abended intermittently under certain circumstances while processing Enscribe data due to uninitialized data. The data is now initialized, and the abend no longer occurs.

31) When using ADTSTARTTIME with SBCOLL, the Collector starting EMS message indicated an invalid RBA (-1). This has been corrected to output the appropriate RBA of where the Collector starts reading audit.

32) Fetchsource processing was attempting to retrieve records from the target system instead of the source system when using a Queue Manager object configured on the target system. Records are now retrieved from the source system as expected in this scenario.

33) SBCOLL was mistakenly age clearing transactions when audit trail data that was at least twelve hours behind the audit trail timestamp where the Collector was currently reading had yet to be acknowledged as processed by at least one thread. This scenario was encountered when specifying an audit trail starting point in the past for reading pre-generated audit on a lightly loaded audit trail. As a result of these transactions being age cleared, the Collector's restart point could be incorrectly advanced. This issue has been corrected, and transactions will no longer be evaluated for age clearing while they are busy being processed for one or more threads.

New Features Description

Increased Record, Row, and Key Size Limits

Support has been added to HPE Shadowbase replication for increased record and row sizes, as well as increased key sizes. In order to handle this, the maximum IPC message size and the STP message size have been increased. The new maximum IPC message size is 57000 bytes, and the maximum STP message size is 64KB. HPE Shadowbase replication is configured to use these new sizes by default, and they must be used in order to support the increased limits. HPE Shadowbase can be configured to use the old message sizes for backwards compatibility. This can be done through the use of the COLL TURBOMODE parameter and SBCOLLTURBOSIZE TACL parameter for IPC message sizes. For STP message sizes, this can be controlled through the CONS STPVERSIONMODE parameter. See the documentation on these parameters for further details.

With the increased message sizes, the current maximum record, row, and key sizes for Enscribe files, SQL/MP tables, and SQL/MX tables are supported as indicated in the table below:

Attribute	Enscribe Limits	SQL/MP Limits	SQL/MX Limits
Record or Row Size	Up to 27,648	Up to ~4096	Up to 32,708[1]
Key Size	Up to 2048[2]	Up to 255	Up to 2048[2]

[1]Note: For rows above a certain size, update events are recorded in the audit trail as an insert followed by a delete. HPE Shadowbase will replicate the events as recorded in the audit trail.

[2]Note: SOLV, SOLVNV, and SOLVMX do not yet support increased key length limits. The maximum key length supported by these programs is 255 bytes.

Considerations

- There are additional restrictions depending on the objects configured in the HPE Shadowbase for Other Servers target environment. For instance, maximum column limits of 999 and maximum statement text size limits of approximately 53,000 bytes exist for DOC Writer objects. See HPE Shadowbase for Other Servers documentation for further details on potential limits.
- Regardless of the target object type, the maximum statement text size for replication is just under 64KB due to the limits of the STP message size.

Increased Maximum Number of Partitions for Enhanced Key-Sequenced (EKS) Files

The maximum number of partitions supported by HPE NonStop Shadowbase replication has been increased to 128 to support the current maximum number of partitions for EKS files.

While much of replication configuration is identical for these files, there are special considerations for replication of purgedata events on EKS files. The primary partition of each EKS file is reserved for storing a portion of the file label, and cannot be used to store user data. As a result, when a purgedata is performed on the file, no purgedata event is generated in the audit trail for the primary partition. To replicate purgedata events for these files, either the PARTONLY or MAPTOPRIPTN option should be specified for the DBS PURGEDATAOPTIONS parameter.

New and Modified Commands

No commands were added or modified since version 6.200.

New and Modified SET CONS Parameters

STPVERSIONMODE   { USELATEST }
                        { LEGACY30K }

Initial Version	Changed Version	Values Default	Value Specification	Alter
6.220		USELATEST	USELATEST, LEGACY30K	No
	Client ☒	Server ☒	Extended ☒ Open Target ☒	

Controls which STP version mode is used by the Consumer for TCP/IP communication. Typically, the same version must be configured on the source and target. Some automatic version downgrading is supported on the NSK side, but automatic downgrading is not supported from a STP version supporting 64KB messages to one that does not for NSK to NSK replication.

The available STPVERSIONMODE options indicate the following:

USELATEST – Use the latest STP version mode available. This includes support for 64KB STP messages. For replication to or from HPE Shadowbase for Other Servers, this parameter value should be used in conjunction with a SHAD_STP_VERSION parameter setting of 6000.

LEGACY30K – Uses the legacy STP version logic supporting a maximum message size of 30000 bytes. This logic is the equivalent of that used prior to HPE NonStop Shadowbase version 6.220. In most cases for replication to or from HPE Shadowbase for Other Servers, this is the equivalent of the SHAD_STP_VERSION parameter setting of 2600. It is the equivalent of 3900 if encryption or compression is enabled. For NSK to NSK replication, the COLL TURBOMODE parameter must be set ON, and the SBCOLLTURBOSIZE TACL parameter must be set to 29964 (or less) in order to use this parameter setting.

NETBUFFERS   num

Initial Version	Changed Version	Values	Default	Minimum	Maximum	Alter
2.500	6.220	2	1	64	No	
	Client ☒		Server ☒	Extended ☒	Open Target ☒	

The number of message blocks to be initially allocated by the Consumer for use in communication with the target process (target Consumer, HPE Shadowbase for Other Servers, etc). Each message from the Consumer to the target process is replied to with an acknowledgement, up to NETBUFFERS messages can be waiting for an acknowledgement, or "outstanding", at a time. The minimum value is 1. The maximum is 64 as of version 6.220. In prior versions, the maximum value is 32. The default value is 2.

Setting the value to 1 defines synchronous communication and the Consumer will wait for an acknowledgement from the target process before sending the next message. A value of 2 or more defines asynchronous communication and the Consumer will continue to send messages to the target process up to the NETBUFFERS value even if acknowledgements are outstanding. If you set NETBUFFERS greater than 1, it is recommended that you set the Collector parameter, MAXCONSWRITES, to a similar value.

When replicating over TCP/IP to a cached target, it is recommended that NETBUFFERS be set to 2x MAXCONSWRITES. For replication to a non-cached target, the recommended setting is 4x MAXCONSWRITES.

Note: As of version 4.000, if the Consumer runs out of available NETBUFFERS, it will attempt to allocate additional buffers dynamically rather than abending as it would previously in this scenario. If the Consumer fails to allocate memory for the additional buffer, however, then it will log a message and abend.

New and Modified SET DBS Parameters

USEREXITPARAM   { value-with-no-spaces }
                         { (value with spaces)   }

Initial Version	Changed Version	Values Default	Value Specification	Alter
6.220		None	Parameter string	No

Specifies a parameter string retrievable during user exit processing. To specify a parameter string that includes spaces, the parameter value must be surrounding in parenthesis. The parameter value can be retrieved from a user exit via the SBGETDBSPARM user exit API function.

New and Modified TACL Parameters

SBCOLLTURBOSIZE TACL Parameter

Initial Version	Changed Version	Values Default	Minimum	Maximum	Alter
2.300	6.220	57000	10000	57000	No

Determines the Collector turbomode buffer size. This buffer size determines the maximum size of IPC messages that can be sent. In addition, this parameter is used by Queue Manager and Consumer processes to determine the maximum buffer size necessary for receiving messages. With version 6.220, the maximum and default values were changed from 29964 to 57000 with the addition of support for large IPCs and long row replication. For cross-version replication with a version of HPE Shadowbase prior to 6.220, SBCOLLTURBOSIZE must be set to 29964. Otherwise, the default value of 57000 is highly recommended. When setting SBCOLLTURBOSIZE, the Collector should also have the COLL TURBOMODE parameter set to ON.

New and Modified User Exit API Functions

SBGETSRCFILEINFO

This API is used in USRXPROCESS or USRXEXCEPTION to retrieve information about the source file.

Product	Availability	Calling Module Usage
HP NonStop Server	Yes	USRXPROCESS USRXEXCEPTION
Other Servers	No	

Syntax for C Programmers

```
short SBGETSRCFILEINFO(char *src_file_name, short *file_type, short
*file_subtype, short *record_length, short *key_offset, short *key_length);
```

Parameters

RETURN-CODE   output
- indicates whether call completed successfully or encountered an error:
    -1 = Parameter content is not available.
     0 = Successful.

src_file_name   output
- Returns source file name if available
- Returns an empty string otherwise

file_type   output
- -1 = Unknown
- 0 = Enscribe file
- 1 = SQL table

file_subtype   output
- -1 = Unknown
- 0 = Unstructured file
- 1 = Relative file
- 2 = Entry-sequenced file
- 3 = Key-sequenced file

record_length   output
- Returns the record length if the source is an Enscribe file.
- Returns -1 otherwise.

key_offset   output
- Returns the key offset if the source is an Enscribe key-sequenced file.
- Returns -1 otherwise.

key_length   output
- Returns the key length if the source is an Enscribe key-sequenced file.
- Returns -1 otherwise.

New or Modified EMS Messages

| 2007 | IF TARGET CONSUMER IS RUNNING A VERSION OF SHADOWBASE PRIOR TO 6201, THE COLL TURBOMODE PARAM MUST BE ON AND THE SBCOLLTURBOSIZE TACL PARAM MUST BE SET TO 29964 TO INTEROPERATE |

Cause: An error 2 was returned to the Collector on a writeread to the Consumer. This may occur when attempted to communicate between incompatible versions of HPE Shadowbase.

Effect: Logged as additional informational specific to error 2 in addition to the normal error message. The Collector abends.

Recovery: If the Consumer is running an earlier version of HPE Shadowbase, upgrade the Consumer HPE Shadowbase version, or attempt to set the COLL TURBOMODE parameter ON and the SBCOLLTURBOSIZE TACL parameter to 29964 and restart.

| 2026 | EVENT DATA SIZE (\<size\>) EXCEEDS MAXIMUM SIZE (\<size\>) FOR FILE/TABLE \<name\>. REMOVE FILE/TABLE FROM CONFIGURATION, DISABLE BEFORE IMAGES ON UPDATES, OR REMOVE THE SBCOLLTURBOSIZE DEFINE FROM THE CONFIGURATION AND RESTART TO CORRECT. EVENT LOCATED AT \<audit-trail-file\> RBA \<audit-rba\> (MAT AA\<mat-seqno\> RBA \<mat-rba\>) |

Cause: An event was detected with a data size that exceeds the maximum length for the HPE Shadowbase environment as configured.

Effect: The Collector abends.

Recovery:	Depending on the file/table involved, remove the file/table from the configuration, disable before images on updates, or remove the SBCOLLTURBOSIZE parameter from the configuration if defined with a smaller value. Then, restart the Collector.

2017	CONSUMER PROCESS ABENDING DUE TO FAILURE TO ALLOCATE STP BUFFER; CONSUMER MAY BE OUT OF MEMORY

Cause:	The Consumer process was unable to retrieve or allocate an additional network buffer. Typically, the Consumer will allocate an additional network buffer if it runs out. If it runs out of memory, however, the allocation may fail.
Effect:	The Consumer abends.
Recovery:	Increase the value of the CONS NETBUFFERS parameter, or reduce COLL MAXCONSWRITES, then restart. If problem persists after these measures, consider setting the COLL TURBOMAXEVENTS parameter.

2123	STP VERSION RETURNED FROM SERVER DOES NOT SUPPORT 64K MESSAGES, AUTOMATIC DOWNGRADE TO 30K SIZE STP VERSIONS NOT SUPPORTED FOR NSK TO NSK; MUST SET STPVERSIONMODE TO LEGACY30K OR CHANGE TARGET TO A SUPPORTING VERSION (e.g. STPVERSIONMODE USELATEST)

Cause:	In NSK to NSK replication, the client Consumer was configured to use 64K STP messages, but the STP version returned from the server Consumer does not support 64K STP messages. Automatically downgrading from a 64K message supporting STP version to a version only supporting smaller messages is not supported for NSK to NSK replication.
Effect:	The client Consumer stops.
Recovery:	Set the client CONS STPVERIONMODE parameter to LEGACY30K, or update the target server CONS to a supporting version and set CONS STPVERSIONMODE to USELATEST. Then, restart.

4515	MESSAGE WITH DUPLICATE OR PROCESSED SEQUENCE NUMBER RECEIVED FROM COLLECTOR, EXPECTED SEQUENCE NUMBER <seqno>, RECEIVED <seqno>. <number> MESSAGES QUEUED LOOKING FOR EXPECTED SEQUENCE NUMBER

Cause: A message was received from the Collector with a sequence number of a message that had already been received or processed.

Effect: The Queue Manager abends.

Recovery: Restart the Queue Manager. If the problem persists, contact Shadowbase support.

4515	MISSING MESSAGE FROM AUDCOLL DETECTED. EXPECTED SEQUENCE NUMBER <seqno>, RECEIVED <seqno>. <number> MESSAGES QUEUED LOOKING FOR EXPECTED SEQUENCE NUMBER

Cause: A message with the expected sequence number was not received by the Queue Manager from the Collector. The specified number of messages were queued while looking for the message with the expected sequence number.

Effect: The Queue Manager abends.

Recovery: Restart the Queue Manager. If the problem persists, contact Shadowbase support.

2017	FILEMNT: FATAL ERROR, ENCOUNTERED A NUMBER OF SECONDARY PARTITIONS (<num>) GREATER THEN THE EXPECTED MAXIMUM OF <num>

Cause: A file was encountered for DDL replication with the number of
                 partitions specified in the message. This number of partitions
                 exceeds the maximum number supported.

Effect: The Consumer abends.

Recovery: Remove the file from DDL replication and restart, or reconfigure
                 the file to within the supported number of partitions.

New and Modified User Information Messages

ERR 2169:   COLL TUROBMODE MUST BE ON AND
                 SBCOLLTURBOSIZE TACL PARAM MUST BE SET TO 29964
                 (OR SMALLER) TO USE STPVERSIONMODE LEGACY30K
                 FOR NSK TO NSK

Cause: The CONS STPVERSIONMODE was set to LEGACY30K for a
                 Consumer configured for NSK to NSK replication, but other
                 parameters were not set as required to allow this. To use
                 STPVERSIONMODE LEGACY30K in this case, the COLL
                 TURBOMODE parameter must be set ON, and the
                 SBCOLLTURBOSIZE TACL parameter must be set to 29964 or
                 less.

Effect: The ADD CONS command fails.

Recovery: Correct the parameters noted to either set STPVERSIONMODE to
                 USELATEST, or to set COLL TURBOMODE ON and
                 SBCOLLTURBOSIZE to 29964.

ERR 1429:   UNABLE TO POSITION TO EVENT SPECIFIED BY
                 ADTSTARTSEQLO AND ADTSTARTRBA

Cause: The Collector was unable to position to the audit trail location specified by the ADTSTARTSEQLO and ADTSTARTRBA COLL parameters.

Effect: The START COLL command fails.

Recovery: Adjust the Collector audit start location parameters or restore the related audit trails. Then, reissue the START COLL command.

New or Modified SOLVCOM User Messages

No SOLVCOM messages have been modified since version 6.200.

Installation Instructions (HPE NonStop Shadowbase)

Shadowbase is now built for HPE NonStop installations using the HPE PAK and UNPAK utilities – refer to the G06.16 and above operating system documentation (or contact Shadowbase Support) for more information. The SBPAK file described below is delivered as a self-extracting file.

1) Binary FTP the SBPAK from your PC (if you obtained the files from a Gravic FTP site) or DVD. If FTP was used, enter the following commands to alter the file code.

> For G-based systems:
> FUP ALTER SBPAK, CODE 700

> For H-based and J-based systems:
> FUP ALTER SBPAK, CODE 800

2) To "unpak" the files from the self-extracting SBPAK, enter the following command:

RUN SBPAK,   *.*.*, VOL <restore volume and subvolume>, LISTALL, MYID

Where <restore volume and subvolume> is the location you want the Shadowbase files installed into, e.g. $DATA.SHADBASE

3) Follow instructions in the Installing Shadowbase on a NonStop System section of the *HPE Shadowbase NonStop Installation and Planning Manual*. You must now run either the INSTALL or ENSINST script.

IMPORTANT NOTES ABOUT USER EXITS:

IF YOU HAVE USER EXITS DEVELOPED WITH COBOL, YOU MUST
RECOMPILE THE USER EXIT "PROGRAMS" SO THAT THE LATEST VERSION
OF USRXLIBO OR USRXLIBN (the native library) IS ADDED TO THE RESULTING
OBJECT FILE THAT WILL SUBSEQUENTLY BE INCLUDED IN THE FINAL
BIND OR LINK.

AFTER COMPILING THE USER EXITS (IF NECESSARY AS DISCUSSED
ABOVE), RUN BINDUSRX, BINDXYPR OR LINKUSRX TO GENERATE THE
CUSTOM CONSUMER (AUDCONS OR AUDCONSN). NOTE THAT THIS
SHOULD BE DONE AFTER THE STANDARD INSTALL IS RUN.

Known Problems Remaining

1) The DOC Roll command is not supported in configurations that use a QMGR out to either the OSS SQL/MX target or an Other Servers target.

2) For SQL/MX, the documented maximum value for the 32-bit float is 3.402823<mark>47e</mark>+38. However, the HPE Shadowbase for Other Servers ODBC Server gets a range error starting with 3.402823<mark>37e</mark>+38 (<mark>difference highlighted and underlined</mark>). The cause of this issue remains under investigation.

3) With a NonStop AUDMON configured for auto-restart and backup CPUs are configured for the Collectors and Consumers, you may have to issue an AUDCOM RUN command for replication to continue after a CPU failure if you have previously entered a SUSPEND COLL command followed by a RESUME COLL command (prior to the failover of AUDMON). A workaround for this is to enter an AUDCOM run command immediately after the RESUME COLL command.

4) During SOLV testing at Gravic, the consumer received an SQL error -1031 for an insert to one of its complex test tables. HP's SQL error description for the error is:

> *A supplied numeric value exceeds the declared precision of the column, some of the supplied values for DECIMAL or VARCHAR columns are invalid, or the supplied row is too long. Also, the disk process might have encountered a bad column in a stored row or a value in an update on a row that would change the length of a VARCHAR column in an entry-sequenced table.*

After investigation of the problem, the error was subsequently reproduced with SQLCI and submitted to HP for further investigation. At this time, it appears that this is caused by a problem in SQL/MP. A fix by HP has yet to be released. HP development is working on the problem.

It has not been determined as to the exact cause of the error -1031. It could be related to the order of certain data types in the schema definition for the SQL table, resulting in an alignment issue that is not handled properly in SQL/MP. The error appeared to be related to SQL processing on the column named C63 in the following series of columns in the schema:

```
,c62      SMALLINT     DEFAULT NULL
,c63      varchar(1)     default NULL
,c64      DATETIME     year to fraction
```

HP has indicated that a workaround could resolve the problem. Add the following define before starting AUDMON.

```
ADD DEFINE = _SQL_CMP_FORCE_EXE_EXPR,
FILE <any Shadowbase TARGETFILE file name>
```

Contact Shadowbase support if the SQL error -1031 continues after you have attempted the workaround.

5) The new SOLVMGR process will abend if configured with an INACTIVITYDELAY greater than 2147 seconds (approximately 35 minutes).

6) In certain conditions, the Consumer may continuously fail (with the Collector restarting at the same audit position) replaying audit and not be able to advance. This problem may occur under the following conditions:

- The Consumer is replicating to a file (table) with a unique alternate key (index).
- The Consumer receives a Guardian error 40 (timeout) on an insert operation due to a lock collision between two active transactions on the alternate key (index).
- The Consumer receives a Guardian error 59 (disk file is bad) on its subsequent delete of the record (row) to 'reverse' the failed insert.

If this issue occurs, either set the Consumer parameter TRACKTX to enable the TRACKTX processing or set the Consumer parameter TIDFILECONSTANT to ON to enable TIDFILE processing to allow the Consumer to continue after the initial failure.

7) A failed insert into a file (table) with alternate keys (index) may leave the record inserted in the base file (table) with no corresponding alternate key records (index entries) under the following conditions:

- The maximum number of active transactions in a Consumer has been reached so the Consumer is doing the "RESOLVETRANS" processing (pre-committing transactions to allow additional simultaneous transactions).
- The Consumer receives a Guardian error 40 (timeout) on an insert operation due to a timeout due to a lock collision on the alternate key (index).

In this case, the record (row) inserted into the base file (table) may not be properly removed.

This condition can be avoided either by turning RESOLVETRANS off or by ensuring that enough Consumers are configured to avoid RESOLVETRANS processing.

8) SOLV has the potential of missing records at the end of a file in a sequence when used with SOLVMGR (solv manager) for 'file chasing'. This problem is more likely to occur with low values of SOLVMGRDELAY, SOLVEOFDELAY, INACTIVITYDELAY and ACTIVITYDELAY are set. By default these values are set to 60 seconds. The minimum values for these configurations are 1 second, however, Gravic does not recommend using values lower than the default to avoid the potential for data loss.

9) For SQL/MX replication, running the Consumer process on the target system is problematical. In several cases, the Consumer needs to access file / schema information from the catalog that is only available on the source system. To prevent issues, the Consumer process should be run on the source system. This is the normal configuration, and will be configured by default when using the SBUNDROS and SBBIDROS scripts.

10) The maximum key length supported by SOLV, SOLVMX, and SOLVNV remains 255 bytes.

11) SOLVMX cannot handle fully qualified SQL/MX table names more than 48 characters in length.

12) For SQL/MX replication, the maximum column name length that can be successfully replicated is 76 characters.

13) When performing a SOLV load of a partitioned file or table from NonStop to NonStop, no warning message is output if the TARGETFNAME is not specified. A warning is output when replicating to a HPE Shadowbase for Other Servers target when configured with the basefile as the source with the ALLPARTITIONS DBS parameter ON. This warning indicates the number of events for which no associated DBS was found in the Consumer during the SOLV load. This warning message should be output in this case if performing a SOLV load to a NonStop target as well.

14) The Consumer does not properly handle statement text that is generated in excess of 64KB for a single statement. This can occur in certain extreme cases with SQL/MX replication or with excessive user exit "put" function calls. To generate statement text exceeding this limit requires the equivalent of a table with approximately 1000 columns with an average column name length in excess of 60 bytes.

15) For DDL replication, certain DDL events require multi-stage processing by HPE Shadowbase replication. Create events, for example, are processed by creating the base file, then performing an alter of the file, and then creating alternate key files (if any). In the event of a failure in the middle of such multi-stage processing, the remaining stages may not be properly applied. If the base file was created, but the Consumer was stopped before the alternate key files were created, for example, on restart the Consumer will attempt to reprocess the event. Since the base file already exists, however, attempting to recreate it fails with an error 10. After receiving this error, the remaining stages are not attempted, and the alternate key files are not created.

16) When replicating from an Enscribe source file using a DDL definition out to a HPE Shadowbase for Other Servers target, the proper where clause is not generated if no dummy TARGETFILE is specified. This results in an empty where clause, causing a SQL error in the Consumer process. The workaround is to specify a dummy TARGETFILE.

17) A problem was introduced with HPE NonStop Shadowbase version 6.220 where the non-native Consumer can trap with an illegal address reference due to misaligned memory when attempting to store a SQL statement. This can occur when loading or replicating to SQL/MP or to a HPE Shadowbase for Other Servers target.

Appendix A – Queue File Replication With SBQFILE

Introduction

Overview

This guide provides information on how to unidirectionally replicate Enscribe queue files to normal Enscribe key-sequenced target files and instructions on how to run the Shadowbase SBQFILE utility to convert these normal key-sequenced files to queue files in the event of a fail-over to the standby node.

SBQFILE Utility

SBQFILE is a Shadowbase utility program that can quickly turn an Encribe queue file into a normal key-sequenced Enscribe file or vice-versa. On a failover situation to a standby node, one would run SBQFILE to turn on the queuefile attribute for the original target files before making use of them by application programs running on the now-primary node. If replication is needed back to the original primary node (now the standby node), on would run SBQFILE against the original source queue files to turn off the queuefile attribute prior to bringing up Shadowbase replication. SBQFILE supports wild-carding so that multiple files can be converted with a single command. Note that SBQFILE's prompting works similar to FUP (HP File Utility Program).

Configuration

Overview

The following sections describe what is necessary to setup the target files and how to configure Shadowbase to replicate your Enscribe queue files using basic Shadowbase (without a user exit).

Target File Preparation

As mentioned previously, Shadowbase can replicate Enscribe queue files to normal Enscribe key-sequenced files and maintain the exact primary key from the queue file (the 8 byte system assigned key, as well as those that have an additional user-defined key prefix). In this method, no special user exit is required. DML (insert, delete) and DDL (create, purge, purgedata, alter) operations are supported. When a create event is replicated, the target will be created without the queuefile attribute being set (normal key-sequenced file).

If the target files are to be pre-created before starting replication, they would be created without the queuefile attribute set. For example, when a source queue file is created with the following FUP commands:

```
SET TYPE K
SET AUDIT
SET KEYLEN 8
SET QUEUEFILE
SET REC 100
CREATE $DATA9.QFILESRC.QFIL001
```
The target would then be created with the following FUP commands (no SET QUEUEFILE):

```
SET TYPE K
SET AUDIT
SET KEYLEN 8
SET REC 100
CREATE $DATA9.QFILETRG.QFIL001
```

If a source is created with an additional user key prefix as in the following FUP commands:

```
SET TYPE K
SET AUDIT
SET KEYLEN 16
SET QUEUEFILE
SET REC 100
CREATE $DATA9.QFILESRC.QFIL001
```

The target would then be created with the following FUP commands (no SET QUEUEFILE):

        SET TYPE K
        SET AUDIT
        SET KEYLEN 16
        SET REC 100
        CREATE $DATA9.QFILETRG.QFIL001

FUP DUP can also be used to create the target files from the source files. Prior to starting Shadowbase, the SBQFILE utility would be run against the target files to convert them from Enscribe queue files to normal key-sequenced Enscribe files. Note that, in order to preserve the exact key value in the target as is in the source, it is important that the target files not have the queuefile attribute enabled. FUP INFO <filename>, DETAIL can show that. SBQFILE also has the list capability.

DBS Object Definition

As with any Shadowbase configuration, one adds DBS objects to identify the source files/tables and the related target files/tables. With queue files, the ALLOWQUEUEFILE DBS parameter must be set ON to enable Shadowbase to pick up and replicate queue file I/O events. If not set to ON, the I/O events will be skipped. Wild-carding of file names is supported.

Note that prior to Shadowbase version 6.201, when the Consumer replicated a create of a queue file, it was created as a normal key-sequenced file (i.e. the queuefile attribute was not set on). Starting with 6.201, the Consumer will now create the target as a queue file. You must set QUEUEASKEYSEQ for the CREATEOPTIONS DBS parameter.

The following is a DBS example:
        ASSUME DBS
        RESET DBS
        SET CONSNAME CONS01
        SET ALLOWQUEUEFILE ON
        SET CREATE ON
        SET CREATEOPTIONS (QUEUEASKEYSEQ)
        SET PURGES ON
        SET PURGEDATAS ON
        SET SOURCEFILE $DATA9.QFILESRC.QFIL*
        SET TARGETFILE $DATA9.QFILETRG.*
        ADD DBS DBS-QFILE-001

SBQFILE Utility Operation

Installation

The SBQFILE utility program runs as a NonStop process. It requires that it be progid'd as a super.super owner, as this level of security is necessary to change the queuefile attribute for a file. The following steps are used to accomplish this:
1. LOGON as super.super.
2. FUP GIVE SBQFILE,255,255
3. FUP SECURE SBQFILE, ,PROGID

Note that starting with the 6.201 release of Shadowbase, SBQFILE is installed using the above steps in the standard product install.

Running SBQFILE / Processing

SBQFILE is initiated from a TACL prompt. To see the command syntax, simply enter SBQFILE without any parameters (i.e. [RUN] SBQFILE). The following is displayed:

> SYNTAX: [RUN] [$VOL.SUBVOL.]SBQFILE {ON | OFF | LIST} <FILESETLIST> [!]

SBQFILE supports 3 basic operations:
- ON is used to cause SBQFILE to enable the queuefile attribute for the files included in the entered *FILESETLIST*.
- OFF is used to cause SBQFILE to disable the queuefile attribute for the files included in the entered *FILESETLIST*.
- LIST is used to cause SBQFILE to output the file type information (queue file or normal key-sequenced file) for the files included in the entered *FILESETLIST*.

Unless "!" (just do it) is entered, SBQFILE will prompt the user to continue for each specific file found in the entered *FILESETLIST*. The following is an example of a prompt:

> SBQFILE: SET QFILE OFF FOR $DATA9.QFILESRC.QFIL001?

"Y" (yes) or "N" (no) are valid responses. "Y" will cause SBQFILE to apply the operation. "N" will cause SBQFILE to skip applying the operation to the related file. "Q" (quit) can also be entered if a wild-carded *FILESETLIST* is entered (e.g. $DATA9.QFILESRC.*). Entering "Q" at this point will cause SBQFILE to discontinue the search and end.

If a wild-carded *FILESETLIST* is entered, SBQFILE will prompt the user to enter how to handle the list of found files. Note that SBQFILE supports the common NonStop wild-carding format similar to what FUP supports (i.e. it handles * and ?). An error will occur if the *FILESETLIST* entered is not supported. The following is an example of a prompt:

> SBQFILE: SET QFILE OFF FOR THE ENTIRE FILESET $DATA9.QFILESRC.*
> (Y[ES], N[ONE], S[ELECT], Q[UIT])?

Entering "Y" (yes) will cause SBQFILE to apply the operation against all of the files found for the wild-carded *FILESETLIST* without any additional prompting. Entering "N" (none) will cause SBQFILE to simply end without affecting any files. Entering "S" (select) will cause SBQFILE to prompt for each file found in the entered *FILESETLIST*. Entering "Q" (quit) will cause SBQFILE to end immediately.

As SBQFILE applies the operation to a given file for ON or OFF, it displays status messages indicating what it is doing as it moves forward. Because it is necessary to turn audit off, apply the change, and then turn audit back on, it outputs messages about these steps. Additionally, it outputs a line to suggest that a TMFCOM DUMP FILES be performed for the related file.

The following are sample SBQFILE runs for ON, OFF, and LIST.

Altering a set of files to enable the queuefile attribute (ON):

> $DATA1 QFILE 31> SBQFILE ON $DATA9.QFILESRC.QFIL1* !
> SBQFILE: BEGIN RUN 2015-04-12 13:46:51
>
> SBQFILE: AUDIT ALTERED TO OFF FOR $DATA9.QFILESRC.QFIL100
> SBQFILE: $DATA9.QFILESRC.QFIL100 IS NOW A QUEUE FILE
> SBQFILE: AUDIT ALTERED TO ON FOR $DATA9.QFILESRC.QFIL100
> SBQFILE: NOTE->TMFCOM DUMP FILES RECOMMENDED FOR
> $DATA9.QFILESRC.QFIL100
>
> SBQFILE: AUDIT ALTERED TO OFF FOR $DATA9.QFILESRC.QFIL101
> SBQFILE: $DATA9.QFILESRC.QFIL101 IS NOW A QUEUE FILE
> SBQFILE: AUDIT ALTERED TO ON FOR $DATA9.QFILESRC.QFIL101
> SBQFILE: NOTE->TMFCOM DUMP FILES RECOMMENDED FOR
> $DATA9.QFILESRC.QFIL101
>
> SBQFILE: END RUN 2015-04-12 13:46:51

List of queue files:

> $DATA1 QFILE 34> SBQFILE LIST $DATA9.QFILESRC.QFIL1*
> SBQFILE: BEGIN RUN 2015-04-12 13:49:51
>
> SBQFILE: $DATA9.QFILESRC.QFIL100 IS A QUEUE FILE
> SBQFILE: $DATA9.QFILESRC.QFIL101 IS A QUEUE FILE
>
> SBQFILE: END RUN 2015-04-12 13:49:51

Altering a set of files to disable the queuefile attribute (OFF):

        $DATA1 QFILE 36> SBQFILE OFF $DATA9.QFILESRC.QFIL1* !
        SBQFILE: BEGIN RUN 2015-04-12 13:59:12

        SBQFILE: AUDIT ALTERED TO OFF FOR $DATA9.QFILESRC.QFIL100
        SBQFILE: $DATA9.QFILESRC.QFIL100 IS NOW A REGULAR KEY-SEQUENCED
        FILE
        SBQFILE: AUDIT ALTERED TO ON FOR $DATA9.QFILESRC.QFIL100
        SBQFILE: NOTE->TMFCOM DUMP FILES RECOMMENDED FOR
        $DATA9.QFILESRC.QFIL100

        SBQFILE: AUDIT ALTERED TO OFF FOR $DATA9.QFILESRC.QFIL101
        SBQFILE: $DATA9.QFILESRC.QFIL101 IS NOW A REGULAR KEY-SEQUENCED
        FILE
        SBQFILE: AUDIT ALTERED TO ON FOR $DATA9.QFILESRC.QFIL101
        SBQFILE: NOTE->TMFCOM DUMP FILES RECOMMENDED FOR
        $DATA9.QFILESRC.QFIL101

        SBQFILE: END RUN 2015-04-12 13:59:12

List of normal key-sequenced files:

        $DATA1 QFILE 28> SBQFILE LIST $DATA9.QFILESRC.QFIL1*
        SBQFILE: BEGIN RUN 2015-04-12 13:44:20

        SBQFILE: $DATA9.QFILESRC.QFIL100 IS A REGULAR KEY-SEQUENCED FILE
        SBQFILE: $DATA9.QFILESRC.QFIL101 IS A REGULAR KEY-SEQUENCED FILE

        SBQFILE: END RUN 2015-04-12 13:44:20

Failover to the Standby Node

In the event that it becomes necessary to switch the application to the standby
node, the original target files should not be used/processed until after SBQFILE is
run with the ON option to make the files queue files.

If unidirectional replication needs to be started going from the original standby
node (now the primary following a failover) to the original primary node (now the
standby node), SBQFILE should be run for the files on the target node using the
OFF option to turn off the queuefile attribute for the related files.

SBQFILE Error Messages

ERROR->INVALID FILESETLIST ARGUMENT SPECIFIED OR MISSING

        Cause: The FILESETLIST entered is either invalid or is missing.

Effect: SBQFILE ends.

Recovery: Correct the FILESETLIST input and try again.

ERROR->FILE GETINFO ERROR <error number> OCCURRED FOR FILE
<file name>, RUN INCOMPLETE

Cause: SBQFILE was unable to obtain file information.

Effect: SBQFILE ends and the run is incomplete.

Recovery: Correct the cause of the reported file error and try again.

ERROR->FILE_OPEN_ ERROR <error number> OCCURRED FOR
DIRECTORY <file name>

Cause: SBQFILE could not open the identified directory.

Effect: SBQFILE ends and the run is incomplete.

Recovery: Correct the cause of the reported file error and try again.

ERROR->FILE_SETKEY_ ERROR <error number> OCCURRED FOR FILE
<file name> ON DIRECTORY <file name>

Cause: SBQFILE could not position to the record it needs in the
           identified directory.

Effect: SBQFILE ends and the run is incomplete.

Recovery: Correct the cause of the reported file error and try again.

ERROR->READUPDATELOCKX ERROR <error number> OCCURRED FOR
FILE <file name> ON DIRECTORY <file name>

Cause: SBQFILE count not read the record it needs from the
           identified directory.

Effect: SBQFILE ends and the run is incomplete.

Recovery: Correct the cause of the reported file error and try again.

ERROR->FILE LABEL RECORD FOR <file name> NOT FOUND IN
DIRECTORY <file name>

Cause: SBQFILE could not find the record it needs in the
identified directory.

Effect: SBQFILE ends and the run is incomplete.

Recovery: Correct the cause of the reported file error and try again.

ERROR->WRITEUPDATEUNLOCKX ERROR <error number> OCCURRED
FOR FILE <file name> ON DIRECTORY <file name>

Cause: SBQFILE could not update the record it needs to in the
identified directory.

Effect: SBQFILE ends and the run is incomplete.

Recovery: Correct the cause of the reported file error and try again.

ERROR->CAN'T ALTER AUDIT ON FOR FILE <file name> DUE TO ERROR
<error number>

Cause: SBQFILE could not turn audit on for the identified file.

Effect: SBQFILE ends and the run is incomplete.

Recovery: Correct the cause of the reported file error and try again.
Note that you may have to use FUP to turn audit ON.

ERROR->CAN'T ALTER AUDIT OFF FOR FILE <file name> DUE TO
ERROR <error number>

Cause: SBQFILE could not turn audit off for the identified file.

Effect: SBQFILE ends and the run is incomplete.

Recovery: Correct the cause of the reported file error and try again.

ERROR->FILEFIND ERROR <error number> OCCURRED, RUN
INCOMPLETE

Cause: SBQFILE detected a "file find" error associated with the
FILESETLIST entered.

Effect: SBQFILE ends and the run is incomplete.

Recovery: Correct the cause of the reported file error and try again.

ERROR->SBQFILE MUST RUN ON THE SAME NODE AS SPECIFIED
FILESET NODE

> Cause: You attempted to run SBQFILE on a different node than the
> node name entered as a part of the FILESETLIST.

> Effect: SBQFILE ends.

> Recovery: Run SBQFILE on the node where the files reside.

ERROR->INVALID INPUT, RE-ENTER RESPONSE?

> Cause: The response entered for a prompt is invalid.

> Effect: SBQFILE prompts again for the input.

> Recovery: Enter a valid response.

ERROR->ABENDING DUE TO STDIN ERROR, ERRNO=<error number>

> Cause: An error occurred on the STDIN file which is typically the
> terminal you run SBQFILE from.

> Effect: SBQFILE ends.

> Recovery: Correct the cause of the reported file error and try again.

ERROR->CAN'T GET <STDIN or STDOUT or HOMETERM> FILE NAME
DUE TO ERROR <error number>

> Cause: SBQFILE is unable to obtain the filename associated with
> STDIN, STDOUT, or the HOMETERM.

> Effect: SBQFILE ends.

> Recovery: Correct the cause of the reported file error and try again.

ERROR->TERMINAL NEEDED FOR USER INTERACTION

> Cause: SBQFILE was not started from a terminal.

> Effect: SBQFILE ends.

> Recovery: Run SBQFILE from a terminal.