

## SOFTWARE RELEASE DOCUMENT (SOFTDOC)

Product:	Shadowbase
Release:	Gravic Version 6.700 HPE NonStop Shadowbase: T1122-AAW (SB Repl/Guardian)
Release Date:	June 30, 2022
Copyright Notice:	Copyright Gravic, Inc. 1995 – 2022( <a href="http://www.gravic.com">www.gravic.com</a> )
File Name:	IPM6700.pdf

**VERY IMPORTANT: Due to licensing changes, existing Shadowbase installations will require a new license file in order to install and run Version 6.700. This is true for any Shadowbase upgrade when the prior release is before Version 6.700 and the new release you plan to install is version 6.700 or later...**

**Contact the HPE License Manager to request a new license file <[license.manager@hpe.com](mailto:license.manager@hpe.com)>. DO NOT INSTALL Shadowbase Version 6.700 (or later) software until a new license file has been received.**

NOTE: If this is a TCD delivery, please see [NOTE FOR TCDs](#) for TCD delivery information.

NOTE: This softdoc covers new features and corrected problems for Shadowbase for 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).

NOTE: This softdoc applies to the HPE NonStop Shadowbase Replication for Guardian component. Other softdocs document the releases of the other components in T1122J06, including:

- HPE Shadowbase Audit Reader,
- HPE Shadowbase Enterprise Manager, and
- HPE Shadowbase Compare for SQL.

You are advised to reference those other softdocs for the changes related to those specific components of T1122J06.

NOTE: **Supported Release Version Updates (RVUS):** This softdoc applies to all currently supported database and operating system versions as described by the [Shadowbase Supported Versions Cross-Reference](#). In addition, visit the <http://www.ShadowbaseSoftware.com> web site for the list of supported platforms and databases, and the Shadowbase Software Support Policy for Software Versions by clicking here <https://shadowbasesoftware.com/support/shadowbase-software-product-release-and-support-policies/>.

## Table of Contents

SOFTWARE RELEASE DOCUMENT (SOFTDOC).....	1
Table of Contents .....	1
Disclaimer .....	2
Note for TCDs.....	3
Mature and Deprecated Features .....	3
Supported Database and Operating System Versions.....	5
Special Notes for Version 6.700 .....	6
Changes in Release 6.700 (since Version 6.610).....	12
New Features .....	12
Problems Fixed .....	24
New and Modified Commands .....	26
LINKUSRX script command (modified).....	26
LINKUSMX script command (modified).....	27
SBBIDIR SB_ADD QMGR command (modified) .....	27
SBBIDIR SB_SET SB_AUDCOM_NAME command (new) .....	28
SBBIDIR SB_ADD PHANTOM_NODE command (new) .....	28
SBBIDROS SB_ADD CONSOS_PARAM command (modified) .....	29
SBBIDROS SB_ADD SBMON_PARAM command (new).....	30
SBBIDROS SB_LIST command (modified).....	31
SBBIDIROS SB_CLEAR command (modified).....	31
SBBIDROS SB_SET SB_AUDCOM_NAME command (new) .....	32
SBUNIDIR SB_SET SB_AUDCOM_NAME command (new) .....	32
SBUNDROS SB_SET SB_AUDCOM_NAME command (new).....	32
New and Modified SET AUD Parameters.....	32
New and Modified SET COLL Parameters .....	34
New and Modified SET CONS Parameters.....	35
New and Modified SET DBS Parameters.....	39
New and Modified SET QMGR Parameters .....	42
New and Modified SOLV AND SOLVNV Parameters .....	43
New and Modified TACL Parameters .....	49
New or Modified EMS Messages .....	51
New and Modified SOLV and SOLVNV User/EMS Messages .....	55
New and Modified User Information Messages .....	65
Installation Instructions (HPE NonStop Shadowbase) .....	68
HPE Release File Structure.....	70
Known Problems Remaining .....	71
[***** End of Document *****].....	82

## ***Disclaimer***

We are distributing this communication in an effort to bring important information to the attention of users of the affected products. We recommend that all users determine the applicability of this information to their individual situations and take appropriate action. We do not represent or warrant that this information is necessarily accurate or complete for all user situations and, consequently, we will not be responsible for any damages resulting from the user's use or disregard of the information provided. To the extent permitted by law, we disclaim all representations and warranties, whether express, implied, statutory, or otherwise, including the warranties of the merchantability, fitness for a particular purpose, title, and non-infringement.

We expect customers of the Shadowbase product suite to “stay current” on Shadowbase releases. This means that you, the customer, should periodically upgrade your Shadowbase software to a newer release that is under support before support ends on your current release. For most customers, this means that you will want to upgrade while your release is in ‘ACTIVE’ support. Otherwise, you run the risk of not being able to get full (or even any if the release has gone ‘OBSOLETE’ end-of-service-life) support for the version you are running.

The Shadowbase Software Policy for Software Versions is described here: <https://shadowbasesoftware.com/support/shadowbase-software-product-release-and-support-policies/>. We encourage all customers to periodically review this material and plan for periodic upgrades to their Shadowbase software. Contact Support if you need additional information.

## **Note for TCDs**

**TCD (Temporary Code Delivery)** – A software update delivered via an SPR downloadable from an FTP dropbox. A TCD is an early version, intended for customer testing only (not production usage). A TCD by definition is restricted to certain customers. Note that a “Gravic TCD” is delivered directly from Gravic, not via HPE, but otherwise has the same attributes.

A TCD is provided only to the specified customer for the purposes agreed between the customer and Gravic as to how it will be used. A TCD is provided subject to the following terms and conditions in addition to the existing written license governing the use of Shadowbase:

- A TCD is provided for evaluation and test purposes only for no more than ninety (90) days use, and is not to be used in production systems
- A TCD may not have been fully tested by Gravic, no warranties are implied as to its behavior
- A TCD is delivered directly from Gravic to the customer, it is not available from HPE/SCOUT
- As testing proceeds, iterative TCD deliveries may be necessary as issues are identified/resolved
- A TCD is temporary, after evaluation it is to be withdrawn from use by the customer
- After testing completes, a TCD may or may not subsequently be released as a Shadowbase TCF or otherwise be included in the Shadowbase product line

Please see <https://www.shadowbasesoftware.com/support/shadowbase-software-product-release-and-support-policies/shadowbase-software-release-glossary/> for additional information.

## **Mature and Deprecated Features**

This section provides a list of mature features (“*mature features*”) and deprecated features (“*deprecated features*”) for HPE NonStop Shadowbase. Gravic will continue to support *mature features* and will continue to include them in future releases; however, Gravic will not enhance those features further or qualify them on newer versions of operating systems and databases as they become available. If you are using a *mature feature*, Gravic highly recommends moving off the *mature feature* to the suggested replacement approach as soon as possible. New installations should avoid using *mature features*; please discuss this issue with Gravic if you feel you must use a *mature feature*.

Gravic will also continue to support *deprecated features* in the current release. However, Gravic plans to eliminate the deprecated features in a future release and strongly

recommends that customers begin the planning and implementation to move off the feature immediately. Contact Gravic if you need assistance in planning your migration.

[Table 1 - Mature Features in HPE NonStop Shadowbase](#), below, lists the features that are currently considered ‘mature’ by Gravic, along with the release/date where they were declared mature. Again, please note that the *mature* designation means the feature will not be enhanced and will eventually be deprecated (retired).

[Table 2 - Deprecated Features in HPE NonStop Shadowbase](#), below, lists the deprecated features. The deprecated features are no longer being enhanced, and are on a schedule to be retired. The table also contains a column indicating the date of the last release that will contain the deprecated functionality (the last release before this end date will be the last release to contain this functionality). Please note that release plans are subject to change without notice and delivery dates are not guaranteed. However, Gravic currently expects that all general availability releases up to and including the date of the last release will contain the deprecated feature(s).

Please see [Shadowbase Software Product Release and Support Policies](#) on [www.shadowbasesoftware.com](http://www.shadowbasesoftware.com) for additional information regarding the Shadowbase Software Support Policy.

**Table 1 - Mature Features in HPE NonStop Shadowbase**

Feature	Description	Mature Release
<b>AUDCOLLN</b>	The original native mode collector, AUDCOLLN is now mature and is being replaced by the new SBCOLL collector. AUDCOLLN will continued to be maintained for critical issues and tested, but may no longer be enhanced with new features and capabilities.	NSB 6.410
<b>CONSTXBQ</b>	A user exit-bound consumer that will process queue file events by use of a mapping file and batches transactions. Consider replicating queue files using the ALLOWQUEUEFILE REPLICATEKEY option along with the CONSTXB user exit-bound consumer instead if transaction batching is needed.	NSB 6.600
<b>Format 1 TMF Audit Trails</b>	While still supported, Gravic recommends that any customers still using Format 1 Audit Trails migrate to Format 2 TMF Audit Trails.	NSB 6.410
<b>ENS2SQL</b>	Tool used to convert Enscribe DDL records to SQL/MP tables, effectively replaced by the SBDDLUTL utility	NSB 6.600
<b>Loader</b>	The original Shadowbase Loader (LOADER) is considered mature for usage with KREP (the key replay feature). Outside of KREP, the Shadowbase Online Loader (SOLV) should be used for any loading of files or tables.	NSB 6.300 May, 2017

**Table 2 - Deprecated Features in HPE NonStop Shadowbase**

Feature	Description	Deprecated Release	Last Release Date
---------	-------------	--------------------	-------------------

<b>Non-native Consumer</b>	The non-native version of the consumer (AUDCONS) is deprecated. Customers still using the non-native version of the consumer should immediately begin planning to move to the native version (AUDCONSN).	NSB 6.410 April, 2018	April, 2020
<b>SBLINK</b>	The original program used to connect SEM to a NonStop environment, replaced by listener AUDCOM process.	NSB 6.220 May, 2016	NSB 6.320 May, 2017
<b>SBQFILE</b>	Program used in conjunction with replicating Enscribe queue files to target non-queue key-sequenced files, replaced with exact queue file replication.	NSB 6.220 May, 2016	NSB 6.320 May, 2017
<b>SQL92</b>	Protocol used by Shadowbase components between the NonStop and Other Servers, replaced by SQLSBV1.	NSB 6.320 May, 2017	NSB 6.410 April, 2018

### ***Supported Database and Operating System Versions***

The database and operating systems versions supported by HPE Shadowbase can be found on the Gravic Shadowbase website. Please see [Supported Databases and Platforms](#) on [www.shadowbasesoftware.com](http://www.shadowbasesoftware.com) for additional information regarding Shadowbase support for databases and operating systems.

## **Special Notes for Version 6.700**

- 1) As of Version 6.700, customers are required to update their license / SHADPASS files. Shadowbase will not start if an old SHADPASS is used. Contact your HPE Shadowbase account representative for a renewed license.
- 2) The Version 6.700 message format has been modified in preparation for future enhancements and is incompatible with prior versions of Shadowbase for NonStop servers. Because of this change, **you must upgrade all nodes at the same time**. Shadowbase for NonStop 6.700 cannot interoperate with prior versions.

If you are replicating from NonStop to NonStop, the recommended upgrade procedure is as follows:

### *Prior to upgrading:*

- Install Shadowbase 6.700 on each node involved in replication in a different location from your existing Shadowbase installation.
- Create a new configuration based upon your existing configuration, using the new location for the Shadowbase programs.
- **Do not** modify the locations of the Shadowbase files required for restarting. These files are compatible across versions and will be used to insure no data loss during the upgrade. These include:
  - The Collector's restart file location
  - The Queue Manager's restart file location
  - The Queue Manager's queue file locations
  - The Consumer's TIDFILE, TRANSLOG, TRACKTXFILE, and OVFLQFILE file locations.

### *Upgrading:*

- Select a time window for performing the upgrade, preferably with little or no activity.
- Gracefully shutdown the existing replications threads using the AUDCOM SHUTDOWN command.
- Restart replication using the new configuration files.
- Shadowbase will resume replication from where it was shutdown.

*QMGR Note:* This version (Version 6.700) of the Queue Manager has been modified to allow it to read pre-V6.700 queue data and to convert it to the new format prior to sending to the Consumer. This facility allows Shadowbase to be upgraded with data in the queue files without data loss, as long as the queue file location is not changed during the upgrade.

If you would like assistance in planning your upgrade of your NonStop Shadowbase environment, please contact HPE support.

3) Please note that this release of Shadowbase includes “SBCREATP” a new utility for converting SQL/MP schemas into various target SQL formats. The SBCREATP utility is being provided as a *Shadowbase TCD (Temporary Code Delivery)*. **NOTE:** This utility is made available solely subject to the terms enclosed in this SOFTDOC. Do not use the utility if you do not agree with the terms.

Please note the following:

- This TCD is an early software version, intended for specific customer/partner testing purposes only (not production usage).
- This TCD is provided only to the specified customer/partner for the purposes agreed between the customer/partner and Gravic/HPE as to how it will be used.
- This TCD is Confidential, and is not to be discussed, shared or installed outside of the specified customer/partner facilities/personnel.
- This TCD may or may not subsequently be released as a Shadowbase TCF or otherwise be included in the Shadowbase product line.
- This TCD is provided subject to the above and the following terms and conditions in addition to the existing written license governing the use of Shadowbase software:
  - TCD is provided for evaluation and test purposes, and is not to be used in production systems.
  - TCD may not have been fully tested by Gravic, no warranties are implied as to its behavior or fitness for a particular purpose.
  - TCD is delivered as part of the normal Shadowbase release from HPE/SCOUT.
  - TCD is temporary, after evaluation it is to be withdrawn from use by the customer/partner.

For additional information, please see <https://www.shadowbasesoftware.com/glossary-policy-and-versions/>.

**NOTE:** Usage of this SBCREATP documentation, software and/or validating password(s) and any other documentation, software, and password(s) for Product is only authorized under the terms of a valid end-user or trial license agreement. If you or your company have not paid license and maintenance fees to Gravic, Inc. or an authorized Shadowbase reseller for use of the software product for which this and any other documentation, software or passwords are designed, you may only use the product for evaluation purposes for a limited period of time (up to 60 days). Unless you have executed a written Product usage agreement with Gravic, Inc. or an authorized Shadowbase reseller, usage of this license file or any other documentation, software or password in this case connotes acceptance by your company of the Shadowbase Confidential Testing and Evaluation Agreement. A review copy of this agreement can be found on the Shadowbase website at <https://www.shadowbasesoftware.com/wp-content/uploads/2015/06/ShadowbaseEvalAgmt.pdf>. In the event that your company is licensing the Product for production use, has paid license and maintenance fees to Gravic, Inc. or an authorized Shadowbase reseller, and does not yet have a written end-user license agreement with Gravic, Inc. or an authorized Shadowbase reseller, usage of this

license file and any other documentation, software and/or password(s) connotes acceptance by you and your company of the terms of the shrink wrap license which can be found at <https://www.shadowbasesoftware.com/wp-content/uploads/2015/06/ShadowbaseTCAgreement.pdf>.

Please contact Shadowbase Support if you have any questions.

4) As of version 6.320, the behavior of the Queue Manager ABNORMALSHUTDOWN parameter and the default value has been changed. As a result, when upgrading from a pre-6.320 version of Shadowbase in an environment with Queue Manager objects configured, the associated consumer processes will need to have a TIDFILE configured (if not already), or the ABNORMALSHUTDOWN parameters will have to be set to STOP for the Queue Manager objects.

5) 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.

6) 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.

7) 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.

8) For those that have user exits, you must use the 6.600 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 6.600.

9) 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.

10) For bi-directional replication environments, due to enhancements made to the internal algorithms within the COLL and CONS, it is required 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.

11) 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.

12) 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.

13) 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.

14) 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.

15) 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, and the parent and child tables are on separate data disks, TMF may store them in reverse order in the audit trail and 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 on the target, 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.

16) 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. For any upgrade to Version 6.6nn, the version has thus been updated according to the VNN format to V66. Any EMS filters using these attributes for filtering must likewise be updated. Alternatively, the SBEMSOLDSSID TAACL parameter can be enabled to have Shadowbase use the old values for the subsystem owner and version.

17) The EMS subsystem version has been updated with this version from V65 to V66 to match the two most significant digits of the current Shadowbase release version id.

18) 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.

19) 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.

20) 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. Contact Support if this issue applies to you and you need to discuss options.

21) As of Version 6.400, the SBLINK and SBQFILE programs have been deprecated and will no longer be shipped alongside the rest of Shadowbase.

22) SBHADW configuration scripts now sets the TAACL parameter SBCOLLAUTOADVMODE in the AMON script to a default value of 0. This parameter used to default to 1.

23) If a Shadowbase process on another node does not respond to a STOP request from the AUDMON, the AUDMON will attempt to kill the process using the PROCESS\_STOP\_ system call. This attempt will often fail if the remote process is PROGID'd, as all of the Shadowbase processes are. Once the AUDMON has passed its STOPPROCESSMAXRETRIES threshold, it will assume the process has been closed

and continue on as usual.

When the environment is started up again, the remote process will fail to start because it still exists from before. This problem can be prevented by creating a SAFEGUARD Access Control List (ACL) for the process to allow access to the user that AUDMON is PROGID'd under. Once the user has been whitelisted with the ACL, AUDMON is able to properly kill the remote process.

24) The SHAD\_DOC\_BLOCKING=1 default setting has been removed from the SBUNDROS and SBBIDROS configuration macros. Shadowbase for Other Servers already handles setting SHAD\_DOC\_BLOCKING's default setting, and Shadowbase for Other Servers version 6.700 and later will set SHAD\_DOC\_BLOCKING to 2 instead of 1.

25) The SBCOLLMULTIENDS parameter has been changed so that the default is now 0 and not 1. This parameter was being set to 1 in the SBBIDROS macro as well as AUDCOLLN and SBCOLL. Both AUDCOLLN and SBCOLL set SBCOLLMULTIENDS to 0 by default and SBBIDROS no longer sets the parameter (i.e. it was removed from SBBIDROS).

## ***Changes in Release 6.700 (since Version 6.610)***

### **New Features**

- 1) SBCOLL has been enhanced to allow it to retry the ARGETANSINAME call when certain ARLIB error codes are encountered. This feature is controlled by a new parameter added: ARLIBRETRYERRORS. See “New and Modified SET COLL Parameters” for more information.
- 2) Support for Internal and External SQL/MX Sequence Object replication has been added to Shadowbase.

A sequence object is a SQL entity which generates sequences of unique numbers – often used to make table row entries unique. An External Sequence Generator is created in MXCI with the CREATE SEQUENCE command. An Internal Sequence Generator is created whenever a SQL/MX table is created containing an IDENTITY column.

Prior to 6.700, Shadowbase did not replicate sequence generator events. In normal processing this was fine - but in failure recovery - there was no way to guarantee that the sequence generators on the source and target systems were synchronized without adding an additional, typically manual, step to the recovery process.

Sequence generator replication allows Shadowbase to replicate both Internal and External sequence generator events from source to target – allowing Shadowbase to recover from failures and keep sequence generators synchronized automatically.

To replicate External Sequence Generators, the user must add a DBS with the source and target sequence generators as the MXSOURCETABLE and MXTARGETTABLE. Then the user must enable the ISSEQGEN parameter for this DBS – this allows Shadowbase to properly interpret the sequence generator events seen in the audit trail.

To replicate an Internal Sequence Generator, the configuration must already have a DBS for the SQL/MX source and target tables containing the IDENTITY column.. Then the user must enable the REPLINTERNALSG parameter for that DBS. This causes the underlying internal sequence generator for the IDENTITY column in that table to also be replicated.

Note that Sequence Generator replication is only supported for uni-directional Disaster Recovery and Sizzling-hot Takeover architectures. To replicate Sequence Generators in active-active bi-directional environments it is recommended to configure alternating sequences between the nodes (e.g. Node A uses even numbered sequences and Node B uses odd numbered sequences) in order to avoid data collisions.

See the New and Modified SET DBS Parameters sections for more information on the respective parameters.

- 3) Enscribe compare supports a new iteration mechanism, for aiding in compare runs while running in a live environment with active application I/O and subsequent replication occurring during the compare.

In this case, it is possible for data to be applied to the source file, but not yet replicated to the target file. If a compare occurs during this time, it is possible that a difference could be found, even if the difference disappears quickly. To prevent this issue, HPE Shadowbase Compare uses an iteration technique to catch these temporary differences caused by latency.

When iteration is enabled, SOLVNV writes any differences found to a special difference file containing the keys of the differences. After the initial compare, SOLVNV will go back and reread the records from the target file, only looking at the records that are in the difference file. If the difference has been resolved, the record in the difference file is removed. After rereading each record, the results are updated to the new values. This iteration can be repeated as many times as needed. After enough iterations occur it can be assumed that any differences that still remain are actual differences and not just temporary ones caused by latency.

It is also possible for there to have been subsequent changes to the source file record during the compare run and iteration processing, that keep the newly re-read target records from matching the value that was retrieved from the source. To account for this, on the final iteration, the source record will also be re-fetched for the comparison.

The parameter VVITERNUMBER is used to specify the maximum number of iteration steps completed. If this parameter is not set, or set to 0, iterations will not occur and the difference file will not be created.

In order to further protect against false positives, it is possible to set a delay between iterations using the parameter VVITERDELAY. This can prevent SOLVNV from quickly completing all iterations before the records have a chance to be replicated to the target file.

- 4) Enscribe Compare now supports compression at either the row or record level, or in blocks of data. Normally, when comparing files with large records, only a few records will be able to be sent in a single IPC buffer, resulting in many IPCs being sent. If communication between SOLV and SOLVNV is slow, compression may be able to increase the rates for comparing. When row or record compression is enabled, SOLV only sends the key values and a short digest of each record, increasing the number of records in each IPC and reducing the amount of data being sent between the compare processes. When block compression is enabled SOLV only sends the start and end key values for a block of records and one digest of all records in the block, increasing

the number of blocks in each IPC and reducing the amount of data being sent between the compare processes.

- 5) Enscribe compare now supports Enscribe unstructured and queue files.
- 6) The Collector (AUDCOLLN and SBCOLL), the Consumer, and the Queue Manager have been enhanced to accept an EMSCOLLECTOR parameter. This parameter allows each of the programs to specify their own EMS Collector they would like to print EMS messages to. See their corresponding "New and Modified" parameter sections for more information.
- 7) The Shadowbase Mapping Utility (SBMAP) has been added to the Shadowbase release for General Availability. See the official manual for more information on how to configure and use this product. SBMAP-related parameters are listed in the "New and Modified SET CONS Parameters" and "New and Modified SET DBS Parameters" sections of the SOFTDOC.
- 8) Support for the PostgreSQL database has been added to the Shadowbase DDL Utility (SBDDLUTL). It can now convert a flattened Enscribe DDL schema into CREATE TABLE statements for PostgreSQL. The following options for the SET command have been added to accommodate some of the extra syntax available in Postgres for CREATE TABLE:

**POSTGRESIFNOTEXISTS <ON | OFF>**

Determines whether the output CREATE TABLE statement will contain an "IF NOT EXISTS" clause. Default value is OFF.

**POSTGRESTABLETYPE "<[GLOBAL | LOCAL] TEMP[ORARY] | UNLOGGED>"**

Determines whether the output CREATE TABLE statement will contain additional table type information, such as GLOBAL or LOCAL, TEMPORARY, or UNLOGGED. The value must be between double-quotes.

**POSTGRESTABLESPACE <tablespace>**

Determines whether the output CREATE TABLE statement will contain a TABLESPACE specification.

**POSTGRESONCOMMIT "<PRESERVE ROWS | DELETE ROWS | DROP>"**

Determines whether the output CREATE TABLE statement will contain an ON COMMIT clause with any of the given values. By default no ON COMMIT clause is output.

- 9) The 'CONTINUE' value for the DBS parameter DOIFNOTRGATRUN has been replaced with new options. The SKIP, SKIPWARN, and SKIPWARN1ST values have been added. If one of these values are used, the event will be skipped if the

target is missing. The allowed values are STOP, WARN, WARN1ST, SKIP, SKIPWARN, and SKIPWARN1ST.

- 10) The LINKUSMX script has been enhanced as follows:
  - The MXCOMP argument is no longer necessary, as it now always SQL/MX compiles the AUDCONXN program regardless. LINKUSMX can still be run with MXCOMP though for backwards compatibility.
  - A new optional argument OWNER has been added. When specified, both AUDCONXN and OAUDCOXN are given and secured to the specified user via FUP. OAUDCOXN is never SQL/MX compiled.
- 11) The LINKUSRX script now accepts the optional argument OWNER. If the CATALOG argument is used with the OWNER argument, the AUDCONSN program will be SQL/MP compiled and altered via SQLCI. If OWNER is run without CATALOG, AUDCONSN will be secured via FUP. Both FUP and SQLCI will give AUDCONSN to the user specified by OWNER, secure it with the permissions of that user, and PROGID it. When using the OWNER argument, the backup Consumer OAUDCONN is always secured and PROGID'd with FUP and never SQL compiled. Both CATALOG and OWNER are optional and can be specified in any order.
- 12) The SBBIDIR configuration macro's SB\_ADD QMGR command now takes primary and backup CPU arguments.
- 13) Modified comments that get added to the \*SPINI, \*SBMON, and \*SBMCU files generated by the SBBIDROS and SBUNDROS configuration macros. The comments were modified to update jargon such as "Shadowbase open server" to "Shadowbase for Other Servers", and replacing the word 'link' with 'thread'. Comments that reference the objects were also modified to clarify if the object is a primary or back DOC Writer, Direct Writer, or DOC Cleaner.
- 14) When using the SBUNDROS or SBBIDROS configuration macros, the SHADPASS file from the Guardian space is no longer copied to the \data directory as the shadpass.ini file in the OSS space. The shadpass.ini file must be placed in the \data directory when installing the OSS side of Shadowbase.
- 15) The SBUNDROS and SBUNIDIR configuration macros will only offer to deploy the TG\* scripts if the target system is a remote system. If the source and target of the configuration is the same system, the macros will not prompt to deploy the TG\* scripts.
- 16) More validation logic (via regular replication and CHECK DBS command) was added to check for potentially problematic scenarios involving files with alternate keys.
- 17) The SB\_ADD CONSOS\_PARAM command for SBBIDROS configurations has been modified to require a primary (**P**) or backup (**B**) argument. This new argument pair is

used in conjunction with the DOC and DCL (DOC Cleaner) arguments to indicate whether the override applies to the primary or backup DOC or DCL process, should a backup process be configured. P and B does not affect overrides for the Consumer or TRS. P and B can be wildcarded with '\*'.

- 18) SBBIDROS configurations support the new SB\_ADD SBMON\_PARAM command. This command can override the settings of the SBMON ADD command written to the Guardian file \*SBMON and can override additional settings in the Guardian file \*SPINI and the OSS file shadparm.ini. Overrides can be specified for a DOC Writer, TRS, or DOC Cleaner. The override options are as follows:

DOC Writer overrides and values

- a. **NAME** – A unique 5-character name for the DOC Writer. Defaults to a name made up of a two-letter prefix combined with the associated link ID.

Modifies the following \*SPINI/shadparm.ini file parameters:

- i. SHAD\_RUN\_PARAMS\_<doc-name>\_1
- ii. SHAD\_<doc-name>\_IP
- iii. SHAD\_<doc-name>\_PORT
- iv. SHAD\_<doc-name>\_PROCESS

- b. **PROCESS\_NAME** – An existing HPE NonStop TCP/IP process name. Defaults to the TCP/IP process name of node src-node-id (see SBMON\_PARAM syntax).

Modifies the following \*SPINI/shadparm.ini file parameter:

- i. SHAD\_<doc-name>\_PROCESS

- c. **SERVER\_TYPE** – A number ranging from 1 to 10, which represents an OSB object type. For example, 1 is DOC Writer, 2 is Direct Writer, 3 is TRS, etc. Defaults to 1 for DOC Writer objects.

*Note: This override should not be used unless directed to do so because it can break the environment and requires all of the object's other parameters to be overridden!*

- d. **EXECUTABLE\_NAME** – A valid executable type for the corresponding object. Defaults to SBOCONSC for cached DOC Writers.

- e. **RECORD\_FORMAT** – A number ranging from 1 to 3. Defaults to 3 for cached SQL statements.

- f. **IP\_ADDRESS** – A valid TCP/IP address available on the system. Defaults to the TCP/IP address associated with node src-node-id (see SBMON\_PARAM

syntax).

- g. **PORT** – A valid port number on the system. Defaults to the port number associated with node src-node-id (see SBMON\_PARAM syntax).
- h. **CPU** – A valid CPU on the system. Defaults to the primary CPU specified for the DOC Writer.
- i. **PRIORITY** – A number ranging from 0 to 199. Defaults to 130.
- j. **NETWORK\_PACKET\_SIZE** – Defaults to 65534 bytes, which is the recommended size.
- k. **TRANSACTION\_PROCESSING** – An option of Y or N. Defaults to Y.
- l. **TRANSACTION\_BOUNDRIES** – An option of Y or N. Defaults to Y.
- m. **STARTING\_SEQUENCE\_NUMBER** – A number ranging from 1 to 999. Defaults to 1.
- n. **SQL\_FORMAT** – A valid SQL format type. Defaults to SQLSBV1.
- o. **STP\_USER** – A username for the Shadowbase connection between the Consumer and DOC Writer. Defaults to OPEN.
- p. **STP\_PASSWORD** – A password for the Shadowbase connection between the Consumer and DOC Writer. Defaults to OPEN.

#### TRS overrides and values

- a. **NAME** – A unique 5-character name for the TRS. Defaults to a name made up of a two-letter prefix combined with the associated link ID.

Modifies the following \*SPINI/shadparm.ini file parameters:

- i. SHAD\_RUN\_PARAMS\_<trs-name>\_1

- b. **SERVER\_TYPE** – A number ranging from 1 to 10, which represents an OSB object type. For example, 1 is DOC Writer, 2 is Direct Writer, 3 is TRS, etc. Defaults to 3 for TRS objects.

*Note: This override should not be used unless directed to do so because it can break the environment and requires all of the object's other parameters to be overridden!*

- c. **EXECUTABLE\_NAME** – A valid executable type for the corresponding object. Defaults to SBODBCTRS for TRS objects.

- d. **READ\_OPEN\_DOCS** – An option of Y or N. Defaults to Y.
- e. **CLEANUP\_DOCS** – An option of Y or N. Defaults to N.
- f. **CONVERT\_INS\_TO\_UPD** – An option of Y or N. Defaults to N.
- g. **STARTING\_SEQUENCE\_NUMBER** – A number ranging from 1 to 999. Defaults to 1.
- h. **SQL\_FORMAT** – A valid SQL format type. Defaults to SQL92.
- i. **CATALOG** – The name of the target SQL/MX catalog. Defaults to the catalog specified in the SB\_ADD CONSOS command.
- j. **SCHEMA** – The name of the target SQL/MX schema. Defaults to the schema specified in the SB\_ADD CONSOS command.
- k. **DB\_USER** – The target database user name. Defaults to the database user name specified in the SB\_ADD CONSOS command.
- l. **DB\_PASSWORD** – The target database password. Defaults to the database password specified in the SB\_ADD CONSOS command.
- m. **DSN** – The target DSN. Defaults to the DSN specified in the SB\_ADD CONSOS command.
- n. **CPU** – A valid CPU number. Defaults to the CPU specified in the SB\_ADD CONSOS command.
- o. **PRIORITY** – A valid number ranging from 0 to 199. Defaults to 130.

#### DOC Cleaner overrides and values

- a. **NAME** – A unique 5-character name for the DOC Cleaner. Defaults to a name made up of a two-letter prefix combined with the associated link ID.

Modifies the following \*SPINI/shadparm.ini file parameters:

- i. SHAD\_RUN\_PARAMS\_<dcl-name>\_1
- b. **SERVER\_TYPE** – A number ranging from 1 to 10, which represents an OSB object type. For example, 1 is DOC Writer, 2 is Direct Writer, 3 is TRS, etc. Defaults to 9 for DOC Cleaner objects.

*Note: This override should not be used unless directed to do so because it can break the environment and requires all of the object's other parameters to be overridden!*

- c. **STARTING\_SEQUENCE\_NUMBER** – A number ranging from 1 to 999. Defaults to 1.
  - d. **CPU** – A valid CPU number. Defaults to the CPU specified in the SB\_ADD CONSOS command.
  - e. **PRIORITY** – A valid number ranging from 0 to 199. Defaults to 130.
- 19) The data message format between components has been enhanced to prepare for future Shadowbase requirements, to allow interoperability between future Shadowbase releases, and to provide additional diagnostic tools for Shadowbase development to aid in identifying and correcting issues. As a result of this change, NonStop Shadowbase Version 6.700 is incompatible with previous versions. All nodes must be upgraded at the same time.
- 20) The AUDCOM process name no longer defaults to \$SRCOM when the \*AMON script starts up AUDCOM. For SBUNIDIR, SBUNDROS, SBBIDIR, and SBBIDROS configurations, the RUN AUDCOM command contained in \*AMON doesn't include a NAME for the AUDCOM by default now. The user can specify a name for the AUDCOM process with the new setting SB\_SET SB\_AUDCOM\_NAME <Audcom-name>
- 21) The SBBIDIR configuration macro now supports the use of phantom nodes. Phantom nodes can be added to a configuration in order to remove nodes from a SBBIDIR configuration while preserving the same link IDs of the previous configuration. The link IDs are used in the names of several files such as the QMGR restart file, the Consumer overflow files, etc. By preserving the link IDs after removing a node, the new configuration can be run without needing to run \*CU in order to clean up the old files with the old link IDs. If phantom nodes are not used when dropping a node in a configuration and if the \*CU script is not run before starting the new configuration, errors will occur.
- 22) The Queue Manager will now reuse the current existing queue file for further writing following a restart, so long as there is space remaining. Previously, a restart of the Queue Manager would result in rolling to the next queue file if there was any data in the current queue file.
- 23) Please note that this release of Shadowbase includes “SBCREATP,” a new utility for converting SQL/MP schemas into various target SQL formats. The SBCREATP utility is being provided as a *Shadowbase TCD (Temporary Code Delivery)*. **NOTE:** This utility is made available solely subject to the terms enclosed in this SOFTDOC. Do not use the utility if you do not agree with the terms.

Please note the following:

- This TCD is an early software version, intended for specific customer/partner testing purposes only (not production usage).
- This TCD is provided only to the specified customer/partner for the purposes agreed between the customer/partner and Gravic/HPE as to how it will be used.
- This TCD is Confidential, and is not to be discussed, shared or installed outside of the specified customer/partner facilities/personnel.
- This TCD may or may not subsequently be released as a Shadowbase TCF or otherwise be included in the Shadowbase product line.
- This TCD is provided subject to the above and the following terms and conditions in addition to the existing written license governing the use of Shadowbase software:
  - TCD is provided for evaluation and test purposes, and is not to be used in production systems.
  - TCD may not have been fully tested by Gravic, no warranties are implied as to its behavior or fitness for a particular purpose.
  - TCD is delivered as part of the normal Shadowbase release from HPE/SCOUT.
  - TCD is temporary, after evaluation it is to be withdrawn from use by the customer/partner.

For additional information, please see

<https://www.shadowbasesoftware.com/glossary-policy-and-versions/>.

**NOTE:** Usage of this SBCREATP documentation, software and/or validating password(s) and any other documentation, software, and password(s) for Product is only authorized under the terms of a valid end-user or trial license agreement. If you or your company have not paid license and maintenance fees to Gravic, Inc. or an authorized Shadowbase reseller for use of the software product for which this and any other documentation, software or passwords are designed, you may only use the product for evaluation purposes for a limited period of time (up to 60 days). Unless you have executed a written Product usage agreement with Gravic, Inc. or an authorized Shadowbase reseller, usage of this license file or any other documentation, software or password in this case connotes acceptance by your company of the Shadowbase Confidential Testing and Evaluation Agreement. A review copy of this agreement can be found on the Shadowbase website at <https://www.shadowbasesoftware.com/wp-content/uploads/2015/06/ShadowbaseEvaluAgmt.pdf>. In the event that your company is licensing the Product for production use, has paid license and maintenance fees to Gravic, Inc. or an authorized Shadowbase reseller, and does not yet have a written end-user license agreement with Gravic, Inc. or an authorized Shadowbase reseller, usage of this license file and any other documentation, software and/or password(s) connotes acceptance by you and your company of the terms of the shrink wrap license which can be found at <https://www.shadowbasesoftware.com/wp-content/uploads/2015/06/ShadowbaseTCAgreement.pdf>.

Please contact Shadowbase Support if you have any questions.

- 24) Due to the internal processing required for JULIANTIMESTAMP procedure calls, we reduced our frequency of calls to it. This may improve performance for environments with a high arrival rate of audit trail events.
- 25) Replication of Enscribe Entry-sequenced files can now be configured to use a unique alternate key to apply update events to the target file. Previously, this was only available for BASE24HANDLERID processing. To configure this, use the KEYSPECIFIER parameter. See the New and Modified SET DBS parameters section for more details.
- 26) A new program named SBLSTRAN has been added to the HPE Shadowbase installation. SBLSTRAN is a tool used to list all the transactions on a system that are currently monitored by TMF. In an optional brief mode, the program will only display the timestamp of the oldest transaction on a system. The program also has an option to start and commit n number of transactions. The purpose of this option is to assist with testing the program's functionality on a system with little or no transactional activity.

Please note that SBLSTRAN will start and communicate with a sysnn.tmfserve process. It will require the appropriate user security to do so.

The following are three examples of running the program. The first example shows how to get the program help.

```
> SBLSTRAN -h
SBLSTRAN - T1122 - V6700L06 - (30Jun22), COPYRIGHT GRAVIC, INC. 2022.
PORTIONS COPYRIGHTED BY AND LICENSED FROM THIRD PARTIES. SEE README FILE.
USAGE SUBJECT TO THE TERMS OF A WRITTEN LICENSE AGREEMENT. PATENTS: SEE
WWW.GRAVIC.COM/GRAVICLABS/PATENTS/PRODUCTS. THE SOFTWARE IS PROVIDED "AS IS"
WITHOUT WARRANTY OR GUARANTEE EITHER STATED OR IMPLIED. USE AT YOUR OWN RISK.
Started Jun 24 2022 09:43:18 for \VIV1(078466/254)

syntax: sblstran [-h] [-b] [-s<n>]
        -h      = Help
        -b      = Brief will only print the oldest start time.
        -s<n>   = Start n concurrent transactions where 0 <= n <= 1000.
                  Option used for testing functionality.
```

The second example demonstrates the standard output of the current transactions monitored by TMF. The -s3 option is added here to start three active transactions because the system has little or no transactional activity. The -s option would normally not be supplied by the user as it starts and lists the transactions as they are started. Regardless if the -s option is supplied, information about all the transactions on the system are then displayed followed by the oldest timestamp. Finally, the three transactions started as a result of the -s option are committed and logged as such.

```
> SBLSTRAN -s3
SBLSTRAN - T1122 - V6700L06 - (30Jun22), COPYRIGHT GRAVIC, INC. 2022.
PORTIONS COPYRIGHTED BY AND LICENSED FROM THIRD PARTIES. SEE README FILE.
```

```
USAGE SUBJECT TO THE TERMS OF A WRITTEN LICENSE AGREEMENT. PATENTS: SEE
WWW.GRAVIC.COM/GRAVICLABS/PATENTS/PRODUCTS. THE SOFTWARE IS PROVIDED "AS IS"
WITHOUT WARRANTY OR GUARANTEE EITHER STATED OR IMPLIED. USE AT YOUR OWN RISK.
Started Jun 24 2022 09:43:01 for \VIV1(078466/254)

Opening the tfile for 3 transactions
Starting 3 transactions
tag:-5146871 id:0xff0112eb77090000 \VIV1.1.317421321
tag:-5146870 id:0xff0112eb770a0000 \VIV1.1.317421322
tag:-5146869 id:0xff0112eb770b0000 \VIV1.1.317421323
3 transactions started

-Start-- -----Transaction----- ----Process(filename)----- --state--
09:43:01 \VIV1.1.317421323          \VIV1.$Y1MF(SBLSTRAN)      ACTIVE
09:43:01 \VIV1.1.317421322          \VIV1.$Y1MF(SBLSTRAN)      ACTIVE
09:43:01 \VIV1.1.317421321          \VIV1.$Y1MF(SBLSTRAN)      ACTIVE

Oldest of 3 transaction(s) is 2022-06-24:09:43:01

Committing 3 transactions
Committed tx tag -5146871
Committed tx tag -5146870
Committed tx tag -5146869
>
```

Only the timestamp of the oldest transaction is displayed in the final example. This is enabled with the brief mode (-b) argument. Here we also add -s3 to start three transactions on a system that currently has no transactional activity. The starting and stopping of the transactions are not logged in brief mode when transactions are enabled by the -s option.

```
> SBLSTRAN -s3 -b
SBLSTRAN - T1122 - V6700L06 - (30Jun22), COPYRIGHT GRAVIC, INC. 2022.
PORTIONS COPYRIGHTED BY AND LICENSED FROM THIRD PARTIES. SEE README FILE.
USAGE SUBJECT TO THE TERMS OF A WRITTEN LICENSE AGREEMENT. PATENTS: SEE
WWW.GRAVIC.COM/GRAVICLABS/PATENTS/PRODUCTS. THE SOFTWARE IS PROVIDED "AS IS"
WITHOUT WARRANTY OR GUARANTEE EITHER STATED OR IMPLIED. USE AT YOUR OWN RISK.
Started Jun 24 2022 09:43:10 for \VIV1(078466/254)

Oldest of 3 transaction(s) is 2022-06-24:09:43:10
>
```

27) A new program named SBLLOCK has been added to the HPE Shadowbase installation. SBLLOCK is a tool that is used to summarize all the locks on a specific file or SQL/MP table. It was intended to help identify all the processes (participants) in the same lock. It lists all the partitions, alfiles/indexes and associated locks for the specified file or table. It will also describe the processes (openers) that have the file or table opened.

The detailed key information about individual locks is not displayed by this tool. Use FUP LISTLOCKS for that if necessary. All locks that involve more than one participant are always displayed. Similar locks with only one participant are grouped together by transaction id or process id with an 'occurs' counter showing the number of similar locks held by the transaction.

Running the program without a file/table name will display a banner, the command syntax and information about what to expect the resulting output to look like along with notes.

```
>SBLSLOCK
SBLSLOCK - T1122 - V6700L06 - (Jan 6 2022), COPYRIGHT GRAVIC, INC. 2021.
PORTIONS COPYRIGHTED BY AND LICENSED FROM THIRD PARTIES. SEE README FILE.
USAGE SUBJECT TO THE TERMS OF A WRITTEN LICENSE AGREEMENT.
PATENTS: SEE WWW.GRAVIC.COM/GRAVICLABS/PATENTS/PRODUCTS.
Started Jun 24 2022 09:13:07 for \VIV1(078466/254)

Syntax: SBLSLOCK <fname>

The output of the program consists of five sections.
1) The key section lists the primary and altkeys/indexes (top left of the report
output).
2) The partition section lists all the partitions of all the keys (top right).
3) The lock section pertains to locks on each row/record (middle left).
4) The participant section lists information about each participant (holders
and waiters) in the record/row lock (middle right).
5) The opener section lists all the processes that have the file opened (bottom).

The key and part numbers in the keys section (1) map to those in the locks section
(3).

Program column abbreviations and definitions.
t lock type; 'r' = record and 'f' = file
flags is a decimal number where bit <0> indicates a generic lock
bits <1:15> are reserved
recid Record specifier (4 bytes) if the lock is a record lock on a format 1
file that is not key-sequenced
klen The length in bytes of the key if the lock is a record lock on either a
key-sequenced file or a format 2 nonkey-sequenced file.
info G - lock is granted
I - lock is intent lock
W - lock in waiting state
state LK^IS - Intent share for file locks needed.
LK^IX - Intent exclusive for file locks. Needed to lock a record using
LK^UX, LK^X.
LK^R - Used only to test for existence of KS. Record range locks
(LK^S AND LK^X).
LK^US - Share for unique record locks. For KS record lock a range is
not locked.
LK^S - Share for file and record locks. For KS record lock a range
is locked. Lines deleted.
LKSIX - Share and intent exclusive derived state for file locks.
LK^UX - Exclusive unique record locks. For KS record lock a range is
locked. Lines deleted.
LK^X - Exclusive for file and record locks. For KS record lock a range
is locked.
occurs The number of locks with a single participant that are lumped together.

For more information on locks see the fup listlocks command in the FUP Manual
and the FILE_GETLOCKINFO_ procedure in the Guardian Procedure Calls Manual.

Sample output
8> sblslock kspart
SBLSLOCK - T1122 - V6700L06 - (Nov 16 2021), COPYRIGHT GRAVIC, INC. 2021.
PORTIONS COPYRIGHTED BY AND LICENSED FROM THIRD PARTIES. SEE README FILE.
USAGE SUBJECT TO THE TERMS OF A WRITTEN LICENSE AGREEMENT.
PATENTS: SEE WWW.GRAVIC.COM/GRAVICLABS/PATENTS/PRODUCTS.
Started Nov 16 2021 16:09:00 for \VIV1(078466/254)

Keyseq Enscribe
-----Keys-----Partitions-----
-File-----File-----Prt -----File-----
0 \VIV1.$GPJ.LISTLOCK.KSPART 0 \VIV1.$GPJ.LISTLOCK.KSPART
1 \VIV1.$DATA1.LISTLOCK.KSPART
2 \VIV1.$DATA2.LISTLOCK.KSPART
```

```

1 \VIV1.$GPJ.LISTLOCK.KSPART0          0 \VIV1.$GPJ.LISTLOCK.KSPART0
                                           1 \VIV1.$DATA1.LISTLOCK.KSPART0
                                           2 \VIV1.$DATA2.LISTLOCK.KSPART0

-----locks----- -----participants-----
-Key- Prt t flags recid klen info state -----tx/process----- (occurs)
  0  0 F   0   -1   0 GI  LK^IX \VIV1.1.241075985/\VIV1.$IUD1 (1)
      R   0   0   8 G  LK^X  \VIV1.1.241075985/\VIV1.$IUD1 (2)
  0  1 R   0   0   8 G  LK^X  \VIV1.1.241075985/\VIV1.$IUD1 (1)
      F   0  -1   0 GI  LK^IX \VIV1.1.241075985/\VIV1.$IUD1 (1)
  0  2 F   0  -1   0 GI  LK^IX \VIV1.1.241075985/\VIV1.$IUD1
      GI  LK^IX \VIV1.0.283355812/\VIV1.$IUD2
      R   0   0   8 G  LK^X  \VIV1.1.241075985/\VIV1.$IUD1 (7)
      R   0   0   8 G  LK^X  \VIV1.0.283355812/\VIV1.$IUD2 (10)
  1  0 F   0  -1   0 GI  LK^IX \VIV1.1.241075985/\VIV1.$IUD1
      GI  LK^IX \VIV1.0.283355812/\VIV1.$IUD2
      R   0   0  14 G  LK^X  \VIV1.0.283355812/\VIV1.$IUD2 (10)
      R   0   0  14 G  LK^X  \VIV1.1.241075985/\VIV1.$IUD1 (10)
  1  1 no locks
  1  2 no locks

Openers
\VIV1.$IUD2 \VIV1.$QA.SBTOOLS.IUD RW-S 255,255
\VIV1.$IUD1 \VIV1.$QA.SBTOOLS.IUD RW-S 255,008
>
    
```

## Problems Fixed

- 1) A selected undoing delete event with a before length of 0 on an Enscribe Entry Sequenced file encountered during RESOLVLOCKS processing could cause the consumer process to abend. This could occur during the abort of a source transaction which included an insertion of a 0 length record into the entry sequenced file.
- 2) When purging the existing restart file in an environment using TRACKTXFILE processing with QMGRS and starting the environment with a start point (time or position) prior to the ending audit position present in the TRACKTXFILE, there was a timing issue that could cause the QMGR process to hang. Once the QMGR outputs the AUDIT DATA PREREAD COMPLETED EMS message, it is past the potential for this hang to occur. The workarounds for this issue were to either purge the TRACKTXFILE (and allow data to potentially be replayed), or to manually stop the QMGR process if it became hung, which would allow it to progress further on a restart.
- 3) The Enscribe Compare program, SOLVNV, did not compare entry sequenced or relative files correctly. Incorrect counts of matches and mismatches were returned.
- 4) The SBUNIDIR configuration macro's SRCU script wasn't purging QMGR queue files when executed.
- 5) The SBUNDROS configuration macro's CONSOS\_PARAM wasn't adding DOC Cleaner overrides to the TGSPINI file.

- 6) In the SBUNDROS configuration macro, the SB\_LIST CONSOS\_PARAM command would not display any CONSOS\_PARAMS that are set.
- 7) The SBUNDROS, SBUNIDIR, SBBIDIR, and SBBIDROS configuration macros now sets the Collector's MAXCONS parameter to the number of Consumers minus the number of QMGRs, with a minimum value of 1. Essentially, the Collector's MAXCONS parameter now counts how many threads the Collector possesses.
- 8) The SBBIDROS configuration macro wouldn't configure DBS overrides if the DBS\_PARAM command's SRC|\* option was wildcarded with the '\*'. Only the SRC option worked.
- 9) Both the LINKURSX and LINKUSXM scripts failed to link COBOL Consumers because it would never include the USRXLIBN library.
- 10) The RUNECOMP script wouldn't compile COBOL Consumers on TNS-X systems because the RUNECOMP script only called the ECOBOL compiler when the XCOBOL compiler is needed on TNS-X.
- 11) The USRXLIBH file was missing declarations for the SBISSQLMXSOURCE(void) and SBISSQLMXTARGET(void) functions. User Exits trying to use those functions would fail on compilation.
- 12) The SRCU scripts generated by the SBUNDIR configuration macro would not clean up QMGR files if more than one QMGR volume was configured. A bug in the macro's routines caused the QMGR's cleanup logic to never be entered.
- 13) The HPE Shadowbase Consumer would open target files in 32-bit mode when applying PURGEDATA events to it. This would cause an error 580 (an open failed because the file was oversized and the opener did not specify use of 64-bit primary keys) for certain types of format 2 files that were larger than 4GB.
- 14) The HPE Shadowbase Consumer would open target files in 32-bit mode when performing BASE24 processing for an event with a DBS configured with BASE24HANDLERID to perform ILF or PTDF file processing. In the case of ILF processing, this would occur for UPDATE events only. This would cause an error 580 (an open failed because the file was oversized and the opener did not specify use of 64-bit primary keys) for format 2 files that were larger than 4GB.
- 15) The collector could fail with a "DUPLICATE TRANSID DURING AVL INSERT" error message following a peer consumer failure due to failing to start a transaction (e.g. BEGINTRANSACTION error 98).
- 16) Under certain rare conditions, replicated data could be "ponged" by AUDCOLLN in a bi-directional environment following multiple successive Collector process restarts, causing data to be replicated back to the source system incorrectly.

- 17) Shadowbase was not properly checking the bounds for extremely large records when replicating out to Shadowbase for Other Servers. This error could potentially lead to lost or corrupted data in the target database. This error occurs with tables that can create single row statement sizes larger than 56K bytes. Shadowbase will now correctly check the statement size and will either stop replication or optionally skip the event. See the [SBSKIPBADSQLSTMTLEN TACL](#) parameter documentation in the [NEW AND MODIFIED TACL PARAMETERS](#) section for more information.
- 18) The SBBIDIR's \*CU script was not properly purging the indexes of the TID file when the parameter SBCONSENSTIDFILE was enabled.
- 19) The QMGR displayed negative event counts for the "COLL RESEND" column in the Queue Statistics when the QMGR is doing a cold start (deleting any queued events) with an existing QMGR restart file present. This can typically happen when:
  - a. The QMGR is stopped using the AUDCOM command "STOP QMGR", and then restarted, or
  - b. An existing environment is restarted after purging just the collector's restart file.
- 20) When using UNDO with AUDCOLLN, the collector could hang at the end of the extract phase, preventing it from completing and transitioning out of the extract phase.
- 21) Fixed an issue with SUSPENDUPD that could cause it to fail to fully complete when QMGRs were being used. The objects would suspend replication as expected, but the completion notification would not propagate upstream, and the STATUS would not update to display the SUSPUPD state.
- 22) The ALLPARTITIONS parameter did not work when used with the SBCOLL collector program. This parameter is now supported with SBCOLL for DML event replication only. The AUDCOLLN collector program still supports use of this parameter for both DML and DDL event replication.

## ***New and Modified Commands***

### **LINKUSRX script command (modified)**

The LINKUSRX script supports the new argument OWNER. Argument OWNER gives the Consumer programs AUDCONSN and OAUDCONN to the user specified by OWNER. Both Consumer programs are also secured with that user's permissions and are prog. ID'd.

If both arguments OWNER and CATALOG are used, the AUDCONSN Consumer is secured and prog. ID'd with SQLCI. If only argument OWNER is used, the AUDCONSN Consumer is secured and prog. ID'd with FUP. The OAUDCONN Consumer is always secured and prog. ID'd with FUP when argument OWNER is used.

The arguments OWNER and CATALOG are both optional and they can be specified in any order. Any arguments besides OWNER and CATALOG will be ignored and will not cause the script to break. Specifiers <user> and <catalog>'s contents are validated in the script.

The syntax for LINKUSRX is as follows:

```
LINKUSRX      [OWNER <user>] [CATALOG <catalog>]
```

### **LINKUSMX script command (modified)**

The LINKUSMX script no longer uses the MXCOMP argument and supports the new argument OWNER. Argument OWNER gives the AUDCONXN and OAUDCOXN Consumers to the user specified by OWNER. Both Consumer programs are also secured with that user's permissions and prog. ID'd via FUP.

LINKUSMX will always SQL/MX compile the AUDCONXN Consumer, hence argument MXCOMP is no longer needed. However, the OAUDCOXN Consumer is never SQL/MX compiled.

Argument OWNER is optional. Any arguments besides OWNER will be ignored and will not cause the script to break, which means the new LINKUSMX syntax is backwards compatible and can take the MXCOMP argument without failing. Specifier <user>'s contents are validated in the script.

The syntax for LINKUSMX is as follows:

```
LINKUSMX      [OWNER <user>]
```

### **SBBIDIR SB\_ADD QMGR command (modified)**

The SB\_ADD QMGR command for the SBBIDIR configuration macro has been modified to require the primary and backup CPU for the QMGR to be specified. Both the primary and backup CPU can optionally be given the value of -1, which defaults the primary and backup CPU values to 1 and 0 respectively.

This pre-6700 syntax still works and keeps SB\_ADD QMGR backwards compatible. However, for all new environments, it is advised to use the new syntax going forward. When using the old syntax, the QMGR's primary CPU defaults to 1 and the backup CPU defaults to 0.

The syntax for SBBIDIR SB\_ADD QMGR is as follows:

```
SB_ADD QMGR <qmgr-id> <pri-cpu|-1> <bk-cpu|-1> <cons-id> <q-volumes> <src-tgt>
```

where

<qmgr-id>	is a unique id given to the queue manager with 4 or less characters.
<pri-cpu>	is the primary cpu of the queue manager. Value of -1 can be specified to have the primary cpu default to use cpu 1.
<bk-cpu>	is the backup cpu of the queue manager. Value of -1 can be specified to have the backup cpu default to use cpu 0.
<cons-id>	is an existing consumer id.
<q-volumes>	is a single volume or volume list where the queue files will reside.
<src-tgt>	the node on which the queue manager process and associated files will reside. Only valid options are SRC or TGT. This parameter can only be used in expand configurations.

### **SBBIDIR SB\_SET SB\_AUDCOM\_NAME command (new)**

The `sb_audcom_name` variable is used to set the name of the Audcom process that runs upon startup of Shadowbase. If `sb_audcom_name` is not specified in the configuration, the Audcom process starts with a system-generated process name. The syntax is:

```
SB_SET SB_AUDCOM_NAME $<process-name>
```

### **SBBIDIR SB\_ADD PHANTOM\_NODE command (new)**

`SB_ADD PHANTOM_NODE` command is used when nodes need to be removed from a configuration. Every time a SBBIDIR configuration is generated, a list of link IDs is created and used to give unique names to Shadowbase objects such as the Collector and QMGR, and to some of the files such as the QMGR queue files and Consumer overflow files. The link IDs' values are based upon the number of nodes in a configuration. The `PHANTOM_NODE` command allows a configuration to be generated as if it had more nodes than it actually does and manipulates the values of the link IDs.

The `PHANTOM_NODE` command should be used when nodes are being removed from a configuration and the `*CU` cleanup script cannot be run to remove the files of the old configuration. Any node can be removed with `PHANTOM_NODE` and any amount of nodes can be removed with `PHANTOM_NODE` as long as at least two working nodes remain.

The syntax for `SB_ADD PHANTOM_NODE` is as follows:

```
SB_ADD PHANTOM_NODE <2-character-node-id>
```

The 2-character-node-id *must* match the name of the node being removed from the configuration, and the node being removed needs to either be deleted from the configuration or commented out. The order of the nodes also must be preserved.

For example, here is a three-node configuration that became a two-node configuration with the use of SB\_ADD PHANTOM\_NODE. The node being removed was commented out.

```
SB_ADD NODE n1 \system1 1.2.3.4 $zj02b - - $n1 \system1.$install.subvol &
      \system1.$data.subvol ~[#MYTERM~] &
      \system1.$install.subvol.shadpass bidir

==SB_ADD NODE n2 \system2 2.3.4.5 $zj02b - - $n2 \system2.$install.subvol &
==      \system2.$data.subvol ~[#MYTERM~] &
==      \system2.$install.subvol.shadpass bidir

SB_ADD PHANTOM_NODE n2

SB_ADD NODE n3 \system3 3.4.5.6 $zj02b - - $n3 \system3.$install.subvol &
      \system3.$data.subvol ~[#MYTERM~] &
      \system3.$install.subvol.shadpass bidir
```

### **SBBIDROS SB\_ADD CONSOS\_PARAM command (modified)**

The SBBIDROS CONSOS\_PARAM command overrides the default settings for the specified Consumer, DOC writer, TRS, or DOC Cleaner.

The overrides are added to the Guardian \*SPINI file and OSS shadparm.ini file when the DOC Writer, TRS, or DOC Cleaner is specified. Otherwise, the overrides are added to the Guardian \*CONF file when the Consumer is specified.

The syntax for SB\_ADD CONSOS\_PARAM is as follows:

SB_ADD CONSOS_PARAM	<src-node-id *> <tgt-node-id *> <cons-id *> <src-doc-trs-dcl *> <p-b *> <param-name-and-value>
where	
<src-node-id>	is an existing source node id. This id can be wildcarded with an asterisk.
<tgt-node-id>	is an existing target node id. This id can be wildcarded with an asterisk.
<cons-id>	is an existing consumer id. This id can be wildcarded with an asterisk.
<src-doc-trs-dcl>	specifies which process the override applies to. Valid options are <b>SRC</b> (Consumer), <b>DOC</b> (DOC Writer), <b>TRS</b> (Transaction Replay Server), and <b>DCL</b> (DOC Cleaner). This specifier can be wildcarded with an asterisk.
<p-b>	specifies if the override applies to the primary DOC or DCL, or the backup DOC or DCL. Valid options are <b>P</b> for primary and <b>B</b> for backup. This specifier can be wildcarded with an asterisk.
<param-name-and-value>	is the parameter name and value which to override.

### **SBIDROS SB\_ADD SBMON\_PARAM command (new)**

The SBMON\_PARAM command overrides the default settings for the specified DOC Writer, TRS, or DOC Cleaner. Overrides can be made to the SBMON's ADD command for the specified object, or overrides can be made to the default object parameters contained in the .ini file.

Alterations to the Consumer cannot be made with SBMON\_PARAM.

Depending on the override specified, the override is added to the Guardian file \*SBMON, the Guardian file \*SPINI and OSS file shadparm.ini, or both.

The syntax for SB\_ADD SBMON\_PARAM is as follows:

```
SB_ADD SBMON_PARAM <src-node-id|*> <tgt-node-id|*> <cons-id|*>
<doc-trs-dcl|*> <p-b|*> <param-name-and-value>

where

<src-node-id> is an existing source node id. This id can be
wildcarded with an asterisk.

<tgt-node-id> is an existing target node id. This id can be
wildcarded with an asterisk.

<cons-id> is an existing consumer id. This id can be wildcarded
with an asterisk.

<doc-trs-dcl> specifies which process the override applies to.
Valid options are DOC (DOC Writer), TRS (Transaction
Replay Server), and DCL (DOC Cleaner). This specifier
can be wildcarded with an asterisk.

<p-b> specifies if the override applies to the primary DOC
or DCL, or the backup DOC or DCL. Valid options are P
for primary and B for backup. This specifier can be
wildcarded with an asterisk.

<param-name-and-value> is the parameter name and value which to override.
```

### SBBIDROS SB\_LIST command (modified)

The SB\_LIST command lists the configuration settings for the nodes, links, DBSs, and all the Shadowbase objects, including any overrides for those objects. Using the 'ALL' keyword will list the whole configuration.

The following options are supported for the SB\_LIST command.

```
SB_LIST {NODE | LINK | CONSOS | QMGR | DBS | BIDIRDBS | UNIDIRDBS |
DBS_PARAM | AUDCOM_PARAM | AUD_PARAM | COLL_PARAM |
CONSOS_PARAM | QMGR_PARAM | SBMON_PARAM | AUDMON_PNAME | TACL_PARAM |
TACL_COMMAND | TACL_DEFINE | ALL}
```

### SBBIDIROS SB\_CLEAR command (modified)

The SB\_CLEAR command clears the configuration settings for the specified object. Use the 'ALL' keyword to clear all configuration settings.

The following options are supported for the SB\_CLEAR command.

```
SB_CLEAR {NODE | LINK | CONSOS | QMGR | DBS | BIDIRDBS | UNIDIRDBS |
DBS_PARAM | AUDCOM_PARAM | AUD_PARAM | COLL_PARAM |
CONSOS_PARAM | QMGR_PARAM | SBMON_PARAM | AUDMON_PNAME |
TACL_PARAM | TACL_COMMAND | TACL_DEFINE | ALL }
```

### **SBIDROS SB\_SET SB\_AUDCOM\_NAME command (new)**

The `sb_audcom_name` variable is used to set the name of the Audcom process that runs upon startup of Shadowbase. If `sb_audcom_name` is not specified in the configuration, the Audcom process starts with a system-generated process name. The syntax is:

```
SB_SET SB_AUDCOM_NAME $<process-name>
```

### **SBUNIDIR SB\_SET SB\_AUDCOM\_NAME command (new)**

The `sb_audcom_name` variable is used to set the name of the Audcom process that runs upon startup of Shadowbase. If `sb_audcom_name` is not specified in the configuration, the Audcom process starts with a system-generated process name. The syntax is:

```
SB_SET SB_AUDCOM_NAME $<process-name>
```

### **SBUNDROS SB\_SET SB\_AUDCOM\_NAME command (new)**

The `sb_audcom_name` variable is used to set the name of the Audcom process that runs upon startup of Shadowbase. If `sb_audcom_name` is not specified in the configuration, the Audcom process starts with a system-generated process name. The syntax is:

```
SB_SET SB_AUDCOM_NAME $<process-name>
```

## ***New and Modified SET AUD Parameters***

CATALOG { Guardian Subvolume }

Initial Version	Changed Version	Values	Default	Minimum	Maximum	Alter
6.700		None				NO

This parameter accepts a valid, fully-qualified or partially-qualified Guardian sub-volume path. When specified, AUDMON registers its AUDHST table within the given catalog. If a SQL/MP catalog does not exist at the given location, AUDMON will create the catalog. If any errors result from this process, AUDMON will ignore this parameter and proceed to create the AUDHST table using the catalog it was registered with upon SQL compilation. If AUDMON was SQL compiled with the NOREGISTER option set ON, then AUDMON will fail to create the AUDHST table and continue to run without audit history tracking. Note that the given SQL catalog location must reside on an audited disk, and must reside on the same system node as the AUDHST table.



## New and Modified SET COLL Parameters

ARLIBRETRYERRORS { (<error>[, error] ... ) }

Initial Version	Changed Version	Values	Default	Minimum	Maximum	Alter
6.700		None				NO

This parameter accepts a comma-separated list of ARLIB error codes enclosed in parentheses. When the collector encounters an ARLIB error that is in the list supplied, the collector will re-try the ARLIB function 2 more times, with a 1-second delay between each attempt. By default no EMS message is output in the case an ARLIB error has been encountered, but can be enabled with the LOGARLIBRETRYERRORS parameter. See ‘New or Modified EMS Messages’ for more information.

Notes:

- This parameter only applies to SBCOLL
- At this time, the # of retry attempts and the #of seconds to delay between retries are not parameterized
- This parameter only affects errors on ARGETANSINAME

LOGARLIBRETRYERRORS { ON }  
{ OFF }

Initial Version	Changed Version	Values	Default	Minimum	Maximum	Alter
6.700		None				YES

This parameter determines whether EMS messages are output in the event that the collector encounters an ARLIB error and goes into ARLIBRETRYERRORS processing. The default value is OFF, meaning no EMS messages are output. This is the recommended setting as enabling the EMS logging has the potential to produce a lot of unwanted EMS output. When enabled, the collector will log an EMS message for each retry attempt, as well as one final message if all retry attempts have been exhausted and the ARLIB error persists. See ‘New or Modified EMS Messages’ for more information.

EMSCollector { Guardian process name }

Initial Version	Changed Version	Values	Default	Minimum	Maximum	Alter
6.700		\$0				NO

Specifies an alternative EMS collector to the default system collector (\$0). All Collector (AUDCOLLN & SBCOLL) EMS messages will be sent to the specified EMS collector.

SBCOLLMULTIENDS { 1 }

{ 0 }

Initial Version	Changed Version	Values	Default	Minimum	Maximum	Alter
6.700		0				NO

Specifies whether to enable multi-ends for multi-ported DOC Writers. When enabled, the Collector coordinates commits down all threads, which is necessary for multi-ported DOC Writers for transactional consistency. Multi-ends is not necessary when each Consumer has its own DOC Writer target. The default is 0, which disables mutli-ends.

MAXCONS number

Initial Version	Changed Version	Values	Default	Minimum	Maximum	Alter
1.000		1		1	255	No

Is the maximum number of Consumer or QMGR processes with which the Collector can communicate. The number must be at least 1. The default is 1. The sum of MAXCONS for all Collectors may not exceed the value specified in the SET AUD MAXCONS command. When configuring QMGRs, the Collector's MAXCONS count is the number of Consumers minus the number of QMGRs, which means MAXCONS essentially counts the amount of threads the Collector possesses.

Depending on your implementation strategy, this parameter may not be needed.

## New and Modified SET CONS Parameters

HOMETERM { Guardian process name }

Initial Version	Changed Version	Values	Default	Minimum	Maximum	Alter
6.700		None				NO

Specifies a HOMETERM process name for the given consumer.

Prior to NSB 6700, the consumer would inherit its hometerm from the AUDMON process that started it. This would be problematic in instances where the user desired to place the consumer into debug, but the consumer ran on a different node than its AUDMON. This parameter helps avoid that issue by allowing the user to specify a hometerm for the consumer that is resting on the same node as it is.

**NOTE:** it is important to be aware of the fact that for security reasons, HPE NonStop disallows users from launching a program in debug if the program is PROGID'd and located on a different node. In order to debug a remote program,

the PROGID on the program must be temporarily removed. Please refer to GenWeb case 10-201103-9901 for more information about this topic.

**EMSCOLLECTOR** { Guardian process name }

Initial Version	Changed Version	Values Default	Minimum	Maximum	Alter
6.700		\$0			NO

Specifies an alternative EMS collector to the default system collector (\$0). All Consumer EMS messages will be sent to the specified EMS collector.

**SBMAPCONF** { Guardian File Name }

Initial Version	Changed Version	Values Default	Minimum	Maximum	Alter
6.700		None			NO

Determines the location of the mapping input file the consumer will use to process replication events and perform data mapping. This is a required parameter, and if omitted, will result in any DBS associated with this consumer with the SBMAPENABLED parameter set ON failing to be added to the Shadowbase configuration.

The Guardian file specified must be a code 101 EDIT file containing mapping file syntax described in the HPE Shadowbase Mapping Utility manual. If the consumer cannot process the mapping file specified here either because it is missing, the wrong file type, or improperly formatted, the consumer will print an error message to EMS and abend.

**SBMAPSRCSUBVOL** { Guardian Subvolume }

Initial Version	Changed Version	Values Default	Minimum	Maximum	Alter
6.700		None			NO

Determines the default location of the SQL/MP source tables specified in mapping input file map-block headers. This is a global setting, applying to all map blocks in a mapping input file. This parameter allows users to specify source tables in their map block headers using relative paths, for example:

```
[SRCTAB -> \NODE.$VOL.SUBVOL.TGTTAB]
.
.
.
```

The node, volume, and sub-volume of the table named "SRCTAB" would be derived from the value of the SBMAPSRCSUBVOL parameter.

**SBMAPTGTSUBVOL** { Guardian Subvolume }

Initial Version	Changed Version	Values	Default	Minimum	Maximum	Alter
6.700		None				NO

Determines the default location of the SQL/MP target tables specified in mapping input file map-block headers. This is a global setting, applying to all mapping blocks in a mapping input file. This parameter allows users to specify target tables in their map block headers using relative paths, for example:

```
[ \NODE.$VOL.SUBVOL.SRCTAB -> TGTTAB ]
.
.
.
```

The node, volume, and sub-volume of the table named "TGTTAB" would be derived from the value of the SBMAPTGTSUBVOL parameter.

**SBMAPTRACELEVEL** { Number }

Initial Version	Changed Version	Values	Default	Minimum	Maximum	Alter
6.700		0		0	7	YES

Allows the user to turn the consumer's SBMAP-related tracing on or off. The different values correspond to different levels of tracing shown below:

Trace Level	Meaning
0	Tracing is disabled
1	Start-up tracing information is output such as the values of configured SBMAP-related parameters, the status of parsing the mapping input file, how many map blocks were detected and parsed, etc.
2	Replication event processing information is output, such as whether a suitable map block was found for the event, whether the SWHERE filter passed or failed, how many mappings were successfully applied, etc.
3	Both start-up and event processing trace information is output.



CATALOG { Guardian Subvolume }

Initial Version	Changed Version	Values Default	Minimum	Maximum	Alter
6.700		None			NO

This parameter accepts a valid, fully-qualified or partially-qualified Guardian sub-volume path. When specified, the consumer registers its TIDFILE (if the TIDFILE is to be created as a SQL table) within the given catalog. If a SQL/MP catalog does not exist at the given location, the consumer will create the catalog. If any errors result from this process, the consumer will ignore this parameter and proceed to create the TIDFILE table using the catalog it was registered with upon SQL compilation. If the consumer was SQL compiled with the NOREGISTER option set ON, then the consumer will fail to create the TIDFILE table and fail to start.

Note that this parameter will also influence which catalog the SBMAP control file is registered under. For more information on SBMAP, please see the [HPE NonStop Shadowbase SBMAP Manual](#).

The given SQL catalog location must reside on an audited disk, and must reside on the same system node as the TIDFILE table and the SBMAP control file. If either of these conditions fail to be met, the consumer will emit an EMS message and fail to start.

QFILERETRYDELAY { centiseconds }

Initial Version	Changed Version	Values Default	Minimum	Maximum	Alter
6.400		10	1	2147483647	No

Sets the delay between retries of an Enscribe queue file IO operation following an error 11. The delay is represented as centiseconds.

Note that either a TRACKTXFILE must be configured when replicating Enscribe queue files. Failure to configure this option can result in a degradation of consumer performance on Shadowbase restarts.

**New and Modified SET DBS Parameters**

ISSEQGEN { ON }  
{ OFF }

Initial Version	Changed Version	Values Default	Minimum	Maximum	Alter
6.700		OFF			NO

This parameter marks a DBS as a special DBS designated for SQL/MX External Sequence Generators. These sequence generators are created in MXCI with the CREATE SEQUENCE command. If the user intends to replicate External Sequence Generator events, a DBS for the source and target sequence generator must be added, and this parameter must be set to ON. This parameter allows Shadowbase to interpret External Sequence Generator audit events properly.

The default value is OFF, implying the DBS is replicating regular SQL/MX tables.

REPLINTERNALSG                    { ON }  
    { OFF }

Initial Version	Changed Version	Values	Default	Minimum	Maximum	Alter
6.700		OFF				NO

This parameter determines whether Shadowbase replicates Internal Sequence Generator events for the given “owner DBS”. An Internal Sequence Generator is created whenever a SQL/MX table is created with an IDENTITY column. An “owner DBS” is one which is configured to replicate between a source SQL/MX table containing an IDENTITY column and a target SQL/MX table containing an IDENTITY column. Internal Sequence Generator replication is responsible for replicating changes on the source table identity column’s associated sequence generator table to the target table identity column’s associated sequence generator table.

This parameter is OFF by default, meaning when an IUD event is done on the owner DBS’s source, the associated Internal Sequence Generator audit event will be ignored by Shadowbase. When set to ON, such events will be replicated.

SEQOFFSET                            { number }

Initial Version	Changed Version	Values	Default	Minimum	Maximum	Alter
6.700		0		-10000000000	10000000000	NO

This parameter determines whether to increment or decrement the current value of a sequence generator event being replicated. A parameter value of zero disables the offset, and the current value of the sequence generator is replicated as-is.

This parameter helps in cases when Shadowbase is recovering from a failure, and there is no guarantee that the source and target sequence generators are still synchronized. Applying an offset to all sequence generator events reduces the possibility of value collisions on the target.

SBMAPENABLED                    { ON }  
    { OFF }

Initial Version	Changed Version	Values	Default	Minimum	Maximum	Alter
6.700		OFF				YES

Determines whether SBMAP replication is enabled for the given DBS. If set to ON, Shadowbase requires that the SBMAPCONF Consumer parameter be set for the consumer pertaining to the given DBS.

EXCLUSIONMODE { SHARED }  
{ EXCLUSIVE }  
{ PROCESSEXCLUSIVE }  
{ PROTECTED }

Initial Version	Changed Version	Values	Default	Minimum	Maximum	Alter
6.700		SHARED				YES

Determines what exclusion mode a Consumer will use when opening a target file associated with this DBS. The default mode is SHARED. For more information about the concept of exclusion modes and how they impact file I/O, see the [Guardian Programmer's Guide](#), and the FILE\_OPEN\_ description in the [Guardian Procedure Calls Reference Manual](#).

Note: when configuring Shadowbase replication to partitioned target files, if an exclusion mode other than SHARED is desired for a given DBS, make sure to enable ALLPARTITIONS for that DBS. Otherwise, make sure that all Consumer(s) replicating to different partitions of the same target file have their DBS's EXCLUSIONMODE set to SHARED, or one consumer may end up locking other consumers out of the file.

The CHECK DBS command will validate this parameter and warn of potential scenarios described above.

DOIFNOTRGATRUN { STOP }  
{ WARN }  
{ WARN1ST }  
{ SKIP }  
{ SKIPWARN }  
{ SKIPWARN1ST }

Initial Version	Changed Version	Values	Default	Value Specification	Alter
4.091	6.700	STOP		Valid Action	No

This parameter is used to enable run-time checking in the CONS to ensure that the target file/table, associated with the current event for the DBS, exists. Note that this run-time checking is only performed if the CONS CHECKFORMISSING

parameter is set to ON. WARN causes the CONS to report the condition for every event. WARN1ST causes the CONS to report the condition for the first event only associated with the target file/table. SKIP causes the CONS to skip any event if the target is missing. SKIPWARN causes the CONS to skip and report the condition for every event. SKIPWARN1ST causes the CONS to skip and report the condition the first time an event is skipped for the target file/table. The CONS abends if STOP is set and it does not find the target file/table. The default is STOP.

**Note:** If DOIFNOTRGATRUN is set to WARN or WARN1ST, the Consumer will still fail if it tries to apply an event to a target file that does not exist. If DOIFNOTRGATRUN is set to WARN or WARN1ST, it does not override that default Consumer behavior.

KEYSPECIFIER <key-specifier>

Initial Version	Changed Version	Values	Default	Minimum	Maximum	Alter
4.090	6.700			-32768	32767	No

This parameter is used to identify the key-specifier associated with a target file alternate key file to be used when applying events. This will be made available to BASE24 handler code in the Consumer to aid in identifying a unique record in a target file. As of version 6.700, this can also be used with a unique alternate keys on Enscribe Entry-sequenced files without the need for BASE24 processing to be enabled.

The value can be entered as a 2 byte string delimited by double quotes (e.g., “A” or “I1” or “IX”, etc.) or can be specified as a number in the range -32768 to 32767 similar to how the key-specifier can be defined within the NonStop FUP utility.

**Note:** By default, HPE Shadowbase uses the primary key value of the source record/row to locate the correct target record/row to process.

### ***New and Modified SET QMGR Parameters***

EMSCollector { Guardian process name }

Initial Version	Changed Version	Values	Default	Minimum	Maximum	Alter
6.700		\$0				NO

Specifies an alternative EMS collector to the default system collector (\$0). All Queue Manager EMS messages will be sent to the specified EMS collector.

## New and Modified SOLV AND SOLVNV Parameters

VVCOMPARELEVEL { FULL }  
 { COMPRESSROWREC }  
 { COMPRESSBLOCK }

Initial Version	Changed Version	Values Default	Minimum	Maximum	Alter
6.700		FULL			NO

This parameter is used to control the level of compression used. Setting this parameter to FULL disables compression. Setting this parameter to COMPRESSROWREC will cause SOLV to compress each row or record individually before sending it along with the key and/or alt key. If compression would not reduce the size of the record, a message will be logged, and VVCOMPARELEVEL will be set to FULL. If VVCOMPARELEVEL is set to COMPRESSROWREC and VVREPORTMODE is set to DETAIL, the report will not contain the data values of the source records when applicable. Setting this parameter to COMPRESSBLOCK will cause SOLV to compress a number of rows or records based on NUMTRANSRECS into a block before sending the block with the start key and end key.

VVITERNUMBER { <count> }

Initial Version	Changed Version	Values Default	Minimum	Maximum	Alter
6.700		0	0	32767	YES

This parameter is used to determine the maximum number of additional iterations occur after the primary run. If equal to 0, iteration is disabled. If iteration is enabled, during the primary run, a set-aside file will be generated. The keys of any differences will be stored in this file. After completing the initial compare, the records in the difference file will be checked again to see if the difference is still there. Iterations will be done until all differences are gone or the maximum number of iterations is reached. This parameter cannot be altered to be less than the current iteration.

VVITERFILENAMEPREFIX { <file-name-prefix> }

Initial Version	Changed Version	Values Default	Minimum	Maximum	Alter
6.700		IT + <VVProcessName>			NO

The prefix name of the difference file that SOLVnV will use for iterations. The difference file will be named based on VVITERFILENAMEPREFIX + R/B where R represents COMPRESSROWREC and B represents COMPRESSBLOCK compare. If this parameter is not set while

VVITERNUMBER is greater than 0 then by default it will be set as “IT” + VVProcessName.

VVITEREXTENTS { (x, y, x )

Initial Version	Changed Version	Values Default	Minimum	Maximum	Alter
6.700		(64, 128, 160)			NO

VVITEREXTENTS can be used to specify the extent definition to be used for the difference file used for iterations.

The following values are accepted for x, y, and z.

x: Set x between 2 and 65535 to define the number of pages for the primary extent.

y: Set y between 2 and 65535 to define the number of pages for the secondary extent.

z: Set z between 16 and 959 to define the maximum number of extents to be allocated.

If you do not specify a VVITEREXTENT, the default primary extent size is 64 pages, the secondary extent size is 128 pages, and the maximum extents allocated is 160.

VVITERDELAY { <delay> }

Initial Version	Changed Version	Values Default	Minimum	Maximum	Alter
6.700		0	0	2 <sup>31</sup> -1	YES

This optional parameter sets the delay between iterations. It is specified in 0.01-seconds. Default is 0 (no delay). Valid values are between 0 (no delay) and 2,147,483,647 (2<sup>31</sup> - 1).

VVITERTIMEOUT { <timeout> }

Initial Version	Changed Version	Values Default	Minimum	Maximum	Alter
6.700		360,000	0	2 <sup>31</sup> -1	YES

This optional parameter sets the timeout for how long SOLV will wait for a response from SOLVNV during iterations. It is specified in 0.01-seconds. Default



VVREPORTEXCLUDEBLOCKS { ON }  
 { OFF }

Initial Version	Changed Version	Values	Default	Minimum	Maximum	Alter
6.700		OFF				NO

VVREPORTEXCLUDEBLOCKS can be used to eliminate block mismatches from the report output. A block mismatch occurs when a block sent from the source does not match the block generated on the target. Block mismatches tell us very little about why the blocks mismatched just that they are different in some way such as a record is missing from source/target, a source/target record have different values.

VVREPORTEXCLUDETIMEOUTS { ON }  
 { OFF }

Initial Version	Changed Version	Values	Default	Minimum	Maximum	Alter
6.700		OFF				NO

VVREPORTEXCLUDETIMEOUTS can be used to eliminate timeouts from the report output. A timeout occurs when a read on the target file hits a lock on a record or when the read takes too long to complete due to any number of reasons. When you get a timeout you can tell a bit about why depending on a few variables. If VVTARGETENSREADTHROUGHLOCKS is set to OFF then it is most likely that the timeout was due to a locked record, otherwise, it is due to the read taking too long. Check your report file for the primary key value of the record that was trying to be read.

VVLOCKRETRIES <count>

Initial Version	Changed Version	Values	Default	Minimum	Maximum	Alter
6.700		2		0	2 <sup>31</sup> -1	NO

This is an optional parameter. Valid values are between 0 and 2<sup>31</sup>-1. The default is the value set to LOCKRETRIES which is 2. This sets the number of times a locked record is retried before the program halts. This setting works in conjunction with VVRETRYDELAY, i.e., SOLVNV will delay VVRETRYDELAY seconds after a lock error occurs before it retries a lock (or read timeout) error.

VVLOCKWARNINGS { ON }

{ OFF }

Initial Version	Changed Version	Values	Default	Minimum	Maximum	Alter
6.700		OFF				NO

This is an optional parameter. If other applications are updating the target file being read by SOLVNV, records may become locked at the same time SOLVNV is attempting to read them. You can enable or disable warning messages when a lock (or read timeout) error is detected. Lock warning messages are OFF (disabled) by default. To enable lock warning messages, set value to ON.

VVRETRYDELAY <retry delay time>

Initial Version	Changed Version	Values	Default	Minimum	Maximum	Alter
6.700		30		1	32,767	NO

This optional parameter defines the number of seconds that SOLVNV waits before retrying a "record lock" error. The minimum is 0 (no delay, retry immediately) and the maximum is 32,767. Note that SOLVNV attempts to read/fetch a record up to VVLOCKRETRIES times, delaying VVRETRYDELAY seconds in between each lock attempt, before terminating. The default is to use the value set for RETRYDELAY which defaults 30 seconds.

GWSTCPNODELAY { ON }  
 { OFF }

Initial Version	Changed Version	Values	Default	Minimum	Maximum	Alter
6.700		ON				NO

This optional parameter defines if the gateway server will set the socket TCP\_NODELAY which disables the Nagle algorithm. The default setting is ON.

GWCTCPNODELAY { ON }  
 { OFF }

Initial Version	Changed Version	Values	Default	Minimum	Maximum	Alter
6.700		ON				NO

This optional parameter defines if the gateway client will set the socket TCP\_NODELAY which disables the Nagle algorithm. The default setting is ON.

GWCNETWINDOWSIZE <send buffer size>

Initial Version	Changed Version	Values	Default	Minimum	Maximum	Alter
-----------------	-----------------	--------	---------	---------	---------	-------

<u>Version</u>	<u>Version</u>	<u>Values Default</u>	<u>Minimum</u>	<u>Maximum</u>	<u>Alter</u>
6.700		245000	128	1048576	NO

This optional parameter defines the gateway client's socket send buffer size. The default value is 245000.

GWSNETWINDOWSIZE <receive buffer size>

<u>Initial Version</u>	<u>Changed Version</u>	<u>Values Default</u>	<u>Minimum</u>	<u>Maximum</u>	<u>Alter</u>
6.700		245000	128	1048576	NO

This optional parameter defines the gateway server's socket receive buffer size. The default value is 245000.

GWCREUSEADDR { ON }  
 { OFF }

<u>Initial Version</u>	<u>Changed Version</u>	<u>Values Default</u>	<u>Minimum</u>	<u>Maximum</u>	<u>Alter</u>
6.700		ON			NO

This optional parameter defines if the gateway client will set the socket reuse address option. The default setting is ON.

GWSREUSEADDR { ON }  
 { OFF }

<u>Initial Version</u>	<u>Changed Version</u>	<u>Values Default</u>	<u>Minimum</u>	<u>Maximum</u>	<u>Alter</u>
6.700		ON			NO

This optional parameter defines if the gateway server will set the socket reuse address option. The default setting is ON.

GWCKEEPALIVE { ON }  
 { OFF }

<u>Initial Version</u>	<u>Changed Version</u>	<u>Values Default</u>	<u>Minimum</u>	<u>Maximum</u>	<u>Alter</u>
6.700		ON			NO

This optional parameter defines if the gateway client will set the keep alive option. The default setting is ON.

GWSKEEPALIVE { ON }  
 { OFF }

<u>Initial Version</u>	<u>Changed Version</u>	<u>Values Default</u>	<u>Minimum</u>	<u>Maximum</u>	<u>Alter</u>
6.700		ON			NO

6.700	ON	NO
-------	----	----

This optional parameter defines if the gateway server will set the keep alive option. The default setting is ON.

VVPURGEITERATIONFILE { ON }  
{ OFF }

Initial Version	Changed Version	Values	Default	Minimum	Maximum	Alter
6.700		ON				NO

This optional parameter defines if SOLVNV purges its iteration file at the end of a compare. The default setting is ON.

VVTARGETENSSEQUENTIALBBLEN <length-in-bytes>

Initial Version	Changed Version	Values	Default	Minimum	Maximum	Alter
6.700		varies		0	32,767	NO

This is an optional parameter. This parameter is used to specify whether sequential block buffering is to be requested during an Enscribe file open. For an overview on sequential block buffering see the ‘Using Cache Buffering or Sequential Block Buffering’ section in the Enscribe Programmer’s Guide. For details on the seq-block-buffer-len go to the FILE\_OPEN\_() procedure in the Guardian Procedure Calls Reference Manual.

The default value is set to the value of ENSSEQUENTIALBBLEN. ENSSEQUENTIALBBLEN’s default value is zero when the Enscribe file type is unstructured or the VVSOLVCONSISTENCYLEVEL is set to TRANSACTION or ROWORRECORD. Note that VVSOLVCONSISTENCYLEVEL is only supported for the value NONE as of 6700. Otherwise the default value is 32,767.

## New and Modified TACL Parameters

SBSKIPBADSQLSTMNTLEN

Initial Version	Changed Version	Values	Default	Minimum	Maximum	Alter
6.700		0		0	1	No

Controls how the consumer handles generated SQL statements for open that exceed Shadowbase’s limit for statement size. If set to 0 or omitted, the consumer

will stop replicating. If set to 1, the consumer will skip the event that generated the large statement but will continue replicating.

```
PARAM SBSKIPBADSQLSTMNTLEN 1
```

## ***New or Modified EMS Messages***

5753	COORDINATOR TRACE FILE MAY ONLY BE CONFIGURED ON PHYSICAL DISKS – FILE <file name> IS CONFIGURED ON A VIRTUAL DISK. TRACING IS DISABLED.
------	--

Cause: The TRACE\_FILE coordinator parameter was configured to point to a virtual drive.

Effect: No trace file is created and the coordinator does not trace.

Recovery: Stop sync, set the TRACE\_FILE parameter to another destination which is on a physical disk, re-start sync.

5753	COORDINATOR TRANSACTION TRACING SUBVOL MAY ONLY BE CONFIGURED ON A PHYSICAL DISK – SUBVOL <subvol> IS CONFIGURED ON A VIRTUAL DISK. TRANSACTION TRACING IS DISABLED.
------	--

Cause: The TRACE\_TX\_SUBVOL coordinator parameter was configured to point to a virtual drive.

Effect: No transaction trace file is created and the coordinator does not do transaction tracing.

Recovery: Stop sync, set the TRACE\_TX\_SUBVOL parameter to another destination which is on a physical disk, re-start sync.

5753	UNABLE TO DETERMINE THE PHYSICALITY OF THE VOLUME THE COORDINATOR TRACE FILE <file name> RESIDES ON. TRACING IS DISABLED.
------	---

Cause: An error occurred when attempting to determine whether the coordinator trace file exists on a physical disk or a virtual disk.

Effect: Tracing is disabled.

Recovery: Check the trace file location, attempt to determine factors that may cause issues; try changing the location of the trace file and re-starting sync.

5753	UNABLE TO DETERMINE THE PHYSICALITY OF THE VOLUME OF THE COORDINATOR TRANSACTION TRACING SUBVOL <subvolume>. TRANSACTION TRACING IS DISABLED.
------	---

- Cause:** An error occurred when attempting to determine whether the coordinator transaction trace file exists on a physical disk or a virtual disk.
- Effect:** Transaction tracing is disabled.
- Recovery:** Check the transaction trace volume, attempt to determine factors that may cause issues; try changing the location of the transaction trace files and re-starting sync.

2111	SOLVNV: UNKNOWN END KEY MODE <end-key-mode> FOUND
------	--

- Cause:** Solvnm encountered an end key mode that it was not expecting while comparing unstructured files.
- Effect:** Solvnm logs to EMS and abends.
- Recovery:** Check your configurations and program file versions and fix any errors found, and then retry.

2127	COLL ENCOUNTERED ERROR (<error>) FOR <ARLIB function> ON FILE <file name>. ATTEMPTING RETRY <count> of <max retry count>...
------	---

- Cause:** SBCOLL encountered an error when calling an ARLIB function (currently, ARGETANSINAME is the only supported function).
- Effect:** SBCOLL logs the EMS message. Additional affects depend on the error code and the presence of certain TAACL parameters, for instance SBCOLLANSINAMEERROK.
- Recovery:** Depending on the error code, SBCOLL may continue reading and processing audit normally, and no operator intervention is required. In other circumstances, the operator should look at the error code and determine the best course of action depending on the significance of the error.

2127	COLL EXHAUSTED ARLIB RETRIES COUNT OF <max retry count>; ARLIB ERROR (<error code>) PERSISTS FOR FILE <file name>. SKIPPING EVENT AT <ADT file name> RBA <adt rba>.
------	---

- Cause: SBCOLL encountered an error when calling an ARLIB function and attempted to retry the given ARLIB function per the ARLIBRETRYERRORS parameter. All retry attempts yielded an error and SBCOLL was not able to execute the given ARLIB function successfully.
- Effect: SBCOLL logs the EMS message. Additional affects depend on the error code and the presence of certain TACL parameters, for instance SBCOLLANSINAMEERROK.
- Recovery: Depending on the error code, SBCOLL may continue reading and processing audit normally, and no operator intervention is required. In other circumstances, the operator should look at the error code and determine the best course of action depending on the significance of the error.

5502	UNABLE TO OPEN ANY FILE(S) ON CONFIGURED DISK(S) – SHUTTING DOWN
------	--

- Cause: SBMETRO attempted to purge, create, and open its tick file(s) (based on the FILE\_NAME and DISK parameters). Upon doing so, SBMETRO encountered some kind of file system error on all such file(s), as a result, SBMETRO has no tick files to process.
- Effect: SBMETRO logs the given error message to EMS and shuts down.
- Recovery: Check EMS – the relevant file system error messages should be present, allowing the user to figure out a way to fix the underlying issue.

2002	COLL USING REPLICATION ACTION SETTING FROM RESTART FILE. RESTART FILE SETTINGS ARE <restart file setting>; CONFIGURED SETTINGS ARE <configured setting>; RESULTING SETTINGS ARE <in use setting>
------	--

Cause: The Collector's REPLICATION parameter's ACTION component was modified using the ALTER command to something other than its configured value. This change is persisted using the configured RESTART file to avoid problems associated with changing the REPLICATION ACTION while a transaction is in progress.

Effect: The Collector will continue using <in use setting>.

Recovery: If <in use setting> is not desired, the REPLICATION parameter can be modified using the ALTER command at a time when there are no outstanding transactions that the Collector is tracking.

2108	DUPLICATE TRANSID DURING AVL INSERT; TRANSID: <trans-id>, TRANS STATUS: <trans-status>; PREFIX: <adt-prefix>, SEQNO: <sequence-number>, MAT <mat>, RBA <rba>
------	--

Cause: The same transaction id was used twice during an AVL insert. This is a symptom of a different problem

Effect: The consumer abends.

Recovery: Contact HPE support.

2099	ALTERNATE KEY [<key>] FOUND IN TERGET FILE [<file>] IS NON-UNIQUE. CONSIDER SETTING DBS PARAMETER 'BASE24HANDLERID' TO <bit4> TO PREVENT POSSIBLE DATA CORRUPTION.
------	--

Cause: The consumer performed validation of a target Enscribe file containing non-unique alternate keys and found a potential configuration issue that could lead to data corruption.

Effect: The consumer prints this EMS message and continues running.

Recovery: None needed, but setting the BASE24HANDLERID parameter properly is strongly recommended. To change frequency of validation and logging, see the ALTKEYVALIDATIONFREQ parameter.

2099	DETECTED DBS PARAMETER [BEFOREVALUES] IS SET TO [OFF] FOR DBS [<db name>], WHICH ALSO HAS [KEYSPECIFIER] CONFIGURED TO [<key>]. THIS SHOULD BE CONFIGURED ONLY IF THE ALTERNATE KEY VALUE IS NEVER UPDATED, OTHERWISE DATA CORRUPTION CAN OCCUR.
------	--

Cause: The consumer performed validation of the DBS configuration during the first replication event for a file where the DBS KEYSPECIFIER is configured and found a potential issue in the configuration.

Effect: The consumer prints this EMS message and continues running.

Recovery: None needed, but turning on BEFOREVALUES is strongly recommended for BASE24 replication, or replication via alternate keys in general.

2027	An incompatible license file version was detected - file must be renewed to expected version. Detected SHADPASS version <version number>. Expected SHADPASS version <version number>. Contact your Shadowbase account representative (for HPE Shadowbase, contact the HPE GNSC or your HPE account manager; your local Shadowbase reseller; or if a direct Gravic Shadowbase customer, follow your Gravic support contract directions or SBSUPPORT@GRAVIC.COM)
------	--

Cause: The user attempted to start a program, but the version in the provided SHADPASS file did not match the expected SHADPASS version.

Effect: The program abends.

Recovery: Acquire a license with the correct SHADPASS version and restart the program.

### ***New and Modified SOLV and SOLVNV User/EMS Messages***

2084	Compressing records would not decrease record length. Setting VVCOMPARELEVEL to FULL
------	--

Cause: Compressing the data would not decrease the amount of data sent per record.

Effect: VVCOMPARELEVEL is set to FULL

Recovery: Set VVCOMPARELEVEL to FULL or use a file that will reduce in size from compression.

2084	SOLV: IPCMAXBUFFERSIZE TOO LARGE, CHANGED FROM <ipcmaxbuffersize> TO <ipc_maxbuf>
------	---

Cause: IPCMAXBUFFERSIZE was set as too large of a value

Effect: IPCMAXBUFFERSIZE gets adjusted down to the maximum value

Recovery: Either set the IPCMAXBUFFERSIZE to a valid value or allow SOLV to do so for you

2017	SOLV: INVALID ITERATION MESSAGE REPLY LENGTH FROM SOLVNV
------	--

Cause: SOLVNV responded to an iteration request with a message that was the incorrect size.

Effect: SOLV abends

Recovery: Contact HPE support.

2017	SOLV: FATAL REPLY CODE FROM SOLVNV DURING ITERATION. SOLV API REPLY CODE(<code>) - <Error>
------	--

Cause: SOLVNV responded to an iteration request with an error.

Effect: SOLV abends

Recovery: Fix the problem causing the error to occur and retry.

2017	SOLV: CANNOT CONTINUE, CURRENT ITERATION GREATER THAN VVITERNUMBER
------	--

Cause: The current iteration that SOLV read from the marker file is greater than the maximum number of iterations allowed.

Effect: SOLV abends

Recovery: Increase VVITERNUMBER or remove the marker record.

2084	SOLV: STARTING COMPARE AFTER ITERATION #
------	--

Cause: Informational message stating from what iteration SOLV is starting at.

Effect: N/A

Recovery: N/A

2111	SOLVNV: ERROR <error-no> WRITING TO FILE <fname>
------	--

Cause: An error was returned while writing to <fname>

Effect: SOLV abends

Recovery: Fix the problem causing the error to occur and retry.

2111	SOLVNV: ERROR <error-no> PURGING FILE <fname>
------	---

Cause: An error was returned while trying to purge <fname>

Effect: SOLV abends

Recovery: Fix the problem causing the error to occur and retry.

2111	SOLVNV: ERROR <error-no> CREATING FILE <fname>
------	--

Cause: An error was returned while trying to create <fname>

Effect: SOLV abends

Recovery: Fix the problem causing the error to occur and retry.

2111 SOLVNV: ERROR <error-no> GETTING FILEINFO FOR FILE  
<fname>

Cause: An error was returned while trying to call  
FILE\_GETINFOLISTBYNAME\_ on <fname>

Effect: SOLV abends

Recovery: Fix the problem causing the error to occur and retry.

2111 SOLVNV: EXISTING FILE <fname> HAS FILE CODE <bad  
code>, EXPECTING <good code> OR <good code>

Cause: SOLVNV will only replace an existing file if it has the expected  
file code.

Effect: SOLV abends

Recovery: Change the file name, or move or purge the existing file.

2110 SOLVNV: FILE <fname> ALREADY EXISTS, PURGING

Cause: Informational message to show that a file a being purged.

Effect: <fname> is purged

Recovery: None

2111 SOLVNV: UNABLE TO ALLOCATE SPACE FOR  
DIFFERENCE RECORD

Cause: SOLVNV was unable to allocate space for a record being written  
to the set-aside file.

Effect: SOLV abends

Recovery: Retry.

2111	SOLVNV: UNABLE TO ALLOCATE SPACE FOR DIFFERENCE RECORD
------	--

Cause: SOLVNV was unable to allocate space for a record being written to the set-aside file.

Effect: SOLV abends

Recovery: Retry.

2111	SOLVNV: ERROR <error> UPDATING RECORD IN FILE <fname>
2111	SOLVNV: ERROR <error> DELETING RECORD IN FILE <fname>

Cause: SOLVNV returned an error trying to either update or delete a record from <fname>.

Effect: SOLV abends

Recovery: Fix the problem causing the error to occur and retry.

2111	SOLVNV: MISSING DIFFERENCE FILE <fname> WAS EXPECTED TO EXIST
------	---

Cause: SOLVNV expected to find the difference file with the name <fname>, but it was unable to find it.

Effect: SOLV abends

Recovery: Retry with a fresh compare or validate that VVITERFILENAME is correct.

2111	SOLVNV: LAST MODIFIED TIMESTAMP FOR FILE <fname> IS <timestamp>, EXPECTED <timestamp>.
------	---

Cause: The difference file SOLVNV has open was modified after the last time SOLV received a reply

Effect: A warning is logged immediately and whenever the compare results are output.

Recovery: Retry with a fresh compare or validate that VVITERFILENAME is correct.

2017	SOLV: INVALID ITERATIONS FILE NAME
------	------------------------------------

Cause: SOLV was unable to create the iterations file with the given VVITERFILENAMEPREFIX.

Effect: SOLV cleanly exits.

Recovery: Make sure VVITERFILENAMEPREFIX is no longer than 7 characters long.

2017	SOLV: LOCK TIMEOUT OCCURRED ON SOLVNV'S SOURCE FILE
------	--

Cause: SOLVNV was unable to gain access to a locked record within the specified source file.

Effect: SOLVNV attempts to access the record again LOCKRETRIES times.

Recovery: Resolve the lock on the source file's record or set ENSREADTHROUGHLOCKS to ON.

2017	SOLV: LOCK TIMEOUT OCCURRED ON SOLVNV'S TARGET FILE
------	--

- Cause:** SOLVNV was unable to gain access to a locked record within the specified target file.
- Effect:** SOLVNV attempts to access the record again VVLOCKRETRIES times.
- Recovery:** Resolve the lock on the target file's record or set VVTARGETENSREADTHROUGHLOCKS to ON.

2110	SOLVNV : READ TIMEOUT OCCURRED ON FILE <FileName>. RETRYING READ. RETRY <VVLockTimeoutOccurrences> / <VVLockRetries>
------	--

- Cause:** SOLVNV was unable to gain access to a locked record within the specified file.
- Effect:** SOLVNV attempts to access the record again LOCKRETRIES times.
- Recovery:** Resolve the lock on the file's record or set VVTARGETENSREADTHROUGHLOCKS to ON.

2110	SOLVNV : READ TIMEOUT OCCURRED ON FILE <FileName> DURING ITERATION <iter>. SKIPPING RECORD.
------	---

- Cause:** SOLVNV was unable to gain access to a locked record within the specified file during the specified iteration.
- Effect:** SOLVNV skips the record and will try to read it again during the next iteration.
- Recovery:** Resolve the lock on the file's record or set VVTARGETENSREADTHROUGHLOCKS to ON.

2111	SOLVNV : READ TIMEOUT OCCURRED ON FILE <FileName>. MAX LOCKRETRIES HIT. CANNOT CONTINUE. TRY ADJUSTING VVLOCKRETIRES
------	--

- Cause:** SOLVNV hit the maximum number of retries while attempting to gain access to one of the file's records.
- Effect:** SOLVNV stores its current run's information in the marker files and SOLV output the run information and shutdown.
- Recovery:** Resolve the lock on the file's record or set VVTARGETENSREADTHROUGHLOCKS to ON.

2017	SOLV: SOLVNV ITERATION SIGNAL TIMEOUT VVITERTIMEOUT EXCEEDED TIME <VVIterTimeout> MINUTES
------	---

- Cause:** SOLV was waiting for a reply from SOLVNV but received nothing before VVIterTimeout was exceeded during an iteration.
- Effect:** SOLV cleanly shutdown.
- Recovery:** Running SOLV again will pick up from the iteration that failed. Either increase VVIterTimeout or resolve SOLV not receiving a response from SOLVNV.

2111	SOLVNV: ERROR INVALID VVCOMPARELEVEL <VVCompareLevel>
------	--

- Cause:** VVCompareLevel was not set as a valid value.
- Effect:** SOLV & SOLVNV shutdown.
- Recovery:** Check VVCompareLevel's documentation in the Enscribe Compare Manual for valid VVCompareLevel values.

2017	ABENDING: ERROR VALIDATING VVCOMPARELEVEL
------	---

- Cause:** VVCompareLevel was not set as a valid value.

Effect: SOLV & SOLVNV shutdown.

Recovery: Check VVCompareLevel's documentation in the Enscribe Compare Manual for valid VVCompareLevel values.

2084	SOLV: ADJUSTING ENSSEQUENTIALBBLen FROM <value> TO 0
------	--

Cause: EnsSequentialBBLen was set to a non 0 value while either solv consistency level was set to transaction/roworrecord or the filetype is unstructured.

Effect: SOLV auto-adjusts EnsSequentialBBLen to 0.

Recovery: If you want to use sequential block buffering then make sure your file is not unstructured and that consistency level is not set to transaction or roworrecord.

2084	SOLV: ADJUSTING VVTARGETENSSEQUENTIALBBLen FROM <value> TO 0
------	--

Cause: VVTargetEnsSequentialBBLen was set to a non 0 value while either solv consistency level was set to transaction/roworrecord or the filetype is unstructured.

Effect: SOLV auto-adjusts VVTargetEnsSequentialBBLen to 0.

Recovery: If you want to use sequential block buffering then make sure your file is not unstructured and that consistency level is not set to transaction or roworrecord.

2017	SOLV: FILE NAME <fname> DOES NOT CONTAIN THE CHARACTER '.'
------	--

Cause: The given file does not contain the character '.' when it was expected.

Effect: SOLV & SOLVNV shutdown.

Recovery: Make sure that the given file contain the character '.' in its file path.

2110	SOLVNV: ERROR <error> WHILE TRYING TO PURGE ITERATION FILE <fname>
------	--

Cause: There was an error with purging the iteration file.

Effect: The iteration file is not purged.

Recovery: Check the error to see why the purge failed.

2137	SQL CATALOG LOCATION <Guardian path> SPECIFIED IN CATALOG PARAMETER RESIDES ON A DIFFERENT NODE THAN THE CONS TIDFILE (<Tidfile path>). CATALOG PARAMETER WILL BE IGNORED.
------	--

Cause: The consumer CATALOG parameter was set to a location on a system node different than the consumer TIDFILE.

Effect: The EMS message is output and the CATALOG parameter is cleared. The consumer tries to use the SQL catalog it was registered with upon SQL compilation during installation for the TIDFILE and SBMAP control file creation. If the consumer was not registered into a catalog during SQL compilation, creation of these files will fail and the consumer will not be able to start.

Recovery: Reset the consumer CATALOG parameter and set it to something residing on the same node as the consumer's TIDFILE. Re-start the consumer with the new parameter value.

2030	UNABLE TO CREATE SQL <object>; sqlcode: <code>, fs error: <error>
------	---

Cause: A Shadowbase program attempted to create a SQL object (e.g. a TABLE, INDEX, CATALOG, etc.) but failed to do so.

Effect: The EMS message is output. Additional effects depend upon which object was attempting to create which SQL entity.

Recovery: Assess proper recovery actions depending upon the listed sqlcode and file system error code.

2137	CATALOG PARAMETER VALUE <Guardian path> REFERS TO A NON-AUDITED DISK. SQL CATALOGS MUST RESIDE ON AUDITED DISKS.
------	--

Cause: During validation AUDMON determined that the either its own, or a consumer's CATALOG parameter were referring to a non-audited disk.

Effect: If AUDMON's CATALOG parameter were in question, AUDMON would disable the catalog parameter and continue running. If the consumer's CATALOG parameter were in question, the consumer would fail to be added to the Shadowbase configuration on start-up.

Recovery: Update the Shadowbase configuration to fix the CATALOG parameter in question.

2001	CLOSED CONSUMER <name>(<process name>)
------	--

Cause: The QMGRs open of the Consumer process was closed. This will occur when the Consumer process fails or is stopped while the QMGR is running.

Effect: Informational. Data will be queued until an open of the Consumer process can be reestablished.

Recovery: None needed specific to this message. The QMGR will reopen the Consumer process automatically when it is started again. However, if the Consumer stopping was not expected, check EMS for other Shadowbase messages for further details, and take recovery action as appropriate for those messages.

## ***New and Modified User Information Messages***

The new user information messages are:

1646	REQUIRED CONSUMER PARAMETER 'SBMAPCONF' MUST BE SET PRIOR TO ENABLING SBMAP PROCESSING FOR AN ASSOCIATED DBS.
------	---

Cause: The user attempted to enable SBMAP processing for a DBS by altering the SBMAPENABLED parameter ON prior to configuring a mapping input file for the associated consumer via the SBMAPCONF parameter.

Effect: AUDMON disallows the altering of the SBMAPENABLED DBS parameter.

Recovery: Alter the SBMAPCONF parameter for the consumer associated with the intended DBS so that it points to a valid SBMAP input file; then re-attempt the altering of the SBMAPENABLED parameter.

1647	ERROR ENCOUNTERED WHILE LOADING MAPPING INPUT FILE <file name> - SEE EMS MESSAGES FOR MORE DETAILS
------	--

Cause: The user attempted to load an SBMAP input file either by altering the SBMAPCONF consumer parameter, or turning on an SBMAPENABLED parameter for a DBS. At this time SBMAP encountered an error loading the mapping input file.

Effect: The error message is produced and SBMAP replication is not enabled for the given consumer.

Recovery: The error could be many different things; an error 11 due to the file in question not existing, or the file name being mis-typed; a syntax error within the mapping input file; an empty mapping input file; etc. See the EMS log for the given consumer to determine the exact cause of the error.

1648	UNEXPECTED ALTER DBS COMMAND RECEIVED – CONSUMER SHOULD BE STOPPED AND DBS RELEASED
------	---

Cause: The user attempted to execute an ALTER DBS command for a parameter that is not alterable while the DBS is captured and the associated consumer is running.

- Effect:** The DBS parameter is not altered.
- Recovery:** This is an unexpected error – please contact Shadowbase support to get the issue resolved.

2017	SQL STATEMENT SIZE EXCEEDS MAXIMUM SIZE (65490) ON <source table name> TO <target table name>
	followed by
2017	SET "SBSKIPBADSQLSTMNTLEN=1" TACL PARAM TO CONTINUE AND SKIP THE STATEMENT

- Cause:** The consumer generated a statement that exceeds the maximum size (65490 bytes) supported by Shadowbase.
- Effect:** The consumer abends and the replication thread stops.
- Recovery:** There are several solutions for resuming replication. You can set the SBSKIPBADSQLSTMNTLEN TACL parameter to 1, as suggested by the second message. In this case, any statements that are too long will be skipped (with an EMS message) and not replicated.
- You can also remove the offending table from replication to allow replication for all other tables.
- You can also create a user exit to remove some of the columns prior to replicating to reduce the size of the replicated data.

2184	COORDINATION CANNOT BE ENABLED ON A DBS OR CONSUMER UNDER THE SAME COLLECTOR AS A CONSUMER WITH A QMGR AND TRACKTX
------	--

- Cause:** The user attempted to start a consumer or capture a DBS in an environment where a consumer is configured with a QMGR and a TRACKTXFILE, and coordination is enabled. Coordination can be caused by MP DCR or Audit DDL.
- Effect:** The consumer does not start or the DBS is not captured.

**Recovery:** This issue can be resolved in one of two ways. You can disable one of the offending parameters, or you can disable coordination without disabling MP DCR or Audit DDL.

An incompatible license file version was detected - file must be renewed to expected version. Detected SHADPASS version <version number>. Expected SHADPASS version <version number>. Contact your Shadowbase account representative (for HPE Shadowbase, contact the HPE GNSC or your HPE account manager; your local Shadowbase reseller; or if a direct Gravic Shadowbase customer, follow your Gravic support contract directions or SBSUPPORT@GRAVIC.COM)

**Cause:** The user attempted to start a program, but the version in the provided SHADPASS file did not match the expected SHADPASS version.

**Effect:** The program abends.

**Recovery:** Acquire a license with the correct SHADPASS version and restart the program.

## ***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

For L-based systems:

FUP ALTER SBPAK, CODE 500

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

Note: Using UNPAK to extract the files instead can result in errors and fail to unpack the files with older versions of UNPAK. Using the RUN command as listed above does not have this issue.

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.

## HPE Release File Structure

If this release is obtained through HPE Scout, the downloaded file is a single self-extracting zip file that is intended to be run on a Windows PC. When the zip file has extracted itself, it will create a folder structure on the PC with all of the various pieces of Shadowbase in appropriate folders. This folder structure is as follows:

### CONTENTS

Each component's installation files are in a separate directory in the self-extracting ZIP file or on the product DVD under the directory \T1122HO6-<SPR ID>. This file set contains the following files:

File or Directory	Description
\readme.txt	This file.
\SBEnterpriseManager	Directory containing the files required to install SEM on a PC.
\TNS-E	Directory containing the installation files for the HPE Integrity NonStop i versions of NS Repl, SAR, and Compare.
SBAuditReader	Directory containing the installation files for SAR.
SBCompare	Directory containing the installation files for Compare.
SBReplication	Directory containing the installation files for NonStop Guardian replication.
\TNS-X	Directory containing the installation files for the HPE Integrity NonStop X versions of NS Repl, SAR, and Compare.
SBAuditReader	Directory containing the installation files for SAR.
SBCompare	Directory containing the installation files for Compare.
SBReplication	Directory containing the installation files for NonStop Guardian replication.

## 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 3.40282347e+38. However, the HPE Shadowbase for Other Servers ODBC Server gets a range error starting with 3.40282337e+38 (difference highlighted and underlined). 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 HPE for further investigation. At this time, it appears that this is caused by a problem in SQL/MP. A fix by HPE has yet to be released. HPE 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

HPE 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 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) 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.

15) 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.

16) SQL/MX tables may not be replicated if the SQL/MX catalog they are located in has multiple definition schema versions defined. This can occur if a new schema is added to a catalog following an upgrade to a newer version of SQL/MX. This can result in a consumer process replicating MX data to log errors similar to the following:

```
\NODE1.$DATA1.ZSDQAG7S.G2CS4Q00 / MXCAT_V.MXSCH_T.TABL_1  
(DBS=DBS-SB-01) -> TARGET IS MISSING. CONSUMER ABENDING.
```

In order to get around this issue, the MX catalog metadata must be upgraded so that there is only one version of the schemas define. The following command can be run to upgrade the catalog metadata:

```
UPGRADE ALL METADATA IN CATALOG catalog
  [ RESTRICT | CASCADE ]
  [ output-spec ]

output-spec is:
  [ log-to-spec ] [ REPORTONLY ]

log-to-spec is:
  { [ LOG TO ] OUTFILE oss-file [ CLEAR ] | LOG TO HOMETERM }
```

- 17) After a QMGR restart with the CONS in a suspended (SUSPEND CONS command issued) state, you will have to issue another SUSPEND CONS command prior to the RESUME CONS command. Issuing just a RESUME CONS command will result in an erroneous “ERR 1244: DIDN'T FIND ANY OBJECTS TO RESUME” message to be issued. Issuing another SUSPEND CONS command followed by the RESUME CONS command will resume the replication flow.
- 18) Enabling the DBS ALLPARTITIONS parameter does not work for DDL replication if using the SBCOLL Collector program. The workaround for this is to use either a DBS SOURCEFILE parameter value with a wildcarded volume, or to use multiple DBSes for the individual partitions. In either case, the DBS TARGETFILE should be set to the primary partition of the target file.
- 19) SQL/MX bidirectional replication sometimes sends events to the target even if those events are part of a transaction that will ultimately be aborted. These events can trigger the Collector to change from SUSPNEXT to REPLSUSP mode, which can seem misleading since the events will not be applied to the target.
- 20) The DUMPTX COLL command does not display information for any tracked peer consumer transactions when using bidirectional Enscribe or SQL/MP replication with SBCOLL. The peer consumer transactions are counted in the number of active transactions displayed by the command, but they are not output.
- 21) The AUD level parameters for global translog support, PEERTRANSLOG and TRANSLOG, do not work for bidirectional replication with SBCOLL.
- 22) Replicating a created table using the new CREATEOPTIONS AUDITED parameter setting can lead to inconsistency between the Collector stats for the source and target tables.
- 23) The checking for the DOIFNOTRGATRUN DBS is only performed when an event is encountered for a target file for which the consumer does not have the schema information in memory. For SQL/MP tables, if the target table is dropped after one or

more events have been replicated, and the schema is already loaded, the consumer will ultimately receive an error 11 for the event, ignore it, and continue, even if DOIFNOTRGATRAN is set to STOP.

- 24) When replicating ALTER events between multiple tables in succession (e.g. Table A to Table B, then Table B to Table C), extra ALTER and COMMIT events can be logged in the STATS.
- 25) Consumer processes that are on a different node from their corresponding AUDMON process are not automatically killed when a non-responsive state is detected.
- 26) The HADW script enables the SBCOLLAUTOADV MODE parameter by default. If configuring an HADW config, be sure to explicitly disable this parameter to ensure that the Shadowbase collector does not automatically advance when the needed audit trail is no longer on the system.
- 27) The Consumer's I/O Tracing output can contain garbage characters when the Consumer encounters an error 73 (table or record is locked).
- 28) Regarding SOLV/SOLVnV shadparm input processing, Shadowbase currently only supports KEYVALUESTART and KEYVALUESTOP values up to 45 bytes long. With other methods SOLV can be pushed to use a KEYVALUESTART value above 239 or 240 bytes, but goes into an infinite loop.
- 29) SBGETREALTARGET will cause the Consumer to trap with an illegal address reference if called for an "end trans" (commit or abort) event, and tracing is enabled. SBGETREALTARGET should never be called for an end trans event.
- 30) The CONS SOLVMAXSESSIONS parameter cannot be set to 0 if a QMGR is configured. Attempting to do so will result in error 1235: REQUIRED PARAMETER NOT SET -> COLLNAME when attempting to add the Consumer object.
- 31) No validation occurs when the DBS VIEWNAME parameter is set to confirm that the DBS SOURCEFILE is a SQL/MP table. This parameter is only valid for SQL/MP source tables.
- 32) Internal Gravic testing revealed that setting the VVMKRHSTEXTENT overwrites the SOLV VVMKREXTENT value. The workaround is to create the SOLVMKRH file with the desired extents via one of the steps described above.
- 33) When attempting to SOLV load a SQL/MP or SQL/MX table with more than 3 primary key columns with a mix of ascending and descending columns in the key, the SQL optimizers for both SQL/MP and SQL/MX will choose a poor access path to the data which results in SOLV or SOLVMX eventually encountering an error 35 or slowing to a crawl. A different method must be used to load this table other than SOLV.

34) Attempting to use UNPAK to unpack the shadowbase self-extracting PAK files will fail with older versions of UNPAK up to T1255H01^AAI. The workaround is to run the self-extracting PAK files instead of using UNPAK, using the method outlined in the installation instructions.

35) Modifications to a DDL record definition during replication will not be picked up by the consumer when either performing a RELEASE DBS followed by a CAPTURE DBS, or removing the existing DBS and adding a new DBS. The consumer process must be restarted in order for the change to be picked up.

36) The Queue Manager will abend with the following error if MAXCONSWRITES is set to 1 and the Consumer process is on a remote system:

```
QMGR QMGR01(\GRAVIC1.$GSQM1): ERROR 28 CALLING PROCEDURE  
SETMODENOWAIT ON FILE \VIV1.$GSCS1 , OPERATION 71, PARAM1 0,  
PARAM2 115
```

```
QMGR QMGR01(\GRAVIC1.$GSQM1): ABENDING: UNABLE TO SET  
EXPAND MESSAGE TRANSMISSION PRIORITY
```

The workaround for this issue is the set MAXCONSWRITES to a value of 2 or higher. The default value for the MAXCONSWRITE parameter is 15.

37) The Shadowbase Consumer sporadically encounters SQL error -8204 / FS error 1061 when doing inserts into partitioned entry-sequenced SQL/MP tables. Based on internal testing, a work-around for this issue is to do a SET DBS CONTROLTABLEEXT1 (SEQUENTIAL INSERT OFF ) for the related DBS object. Note that the CONTROLTABLEEXT parameter has superseded CONTROLTABLEEXT<n>, and if present, CONTROLTABLEEXT<n> will be ignored (CONTROLTABLEEXT1, in this case).

38) Shadowbase BASE24 handling has a potential efficiency issue when processing events associated with a DBS where the source or target file definitions could be associated with multiple files. This is especially apparent when using "true dynamic" selection mapping (i.e. no MAPFROMSOURCE and SOURCEFILE/TARGET is used only for mapping) and when using MAPFROMSOURCE ON and the SOURCEFILE has a wild-card (\*) in the volume or sub-volume. For now a workaround would be to try and limit the use of these types of DBS specifications for BASE24.

39) Enscribe DDL definitions with data types containing negative scale causes SBDDLUTL to generate invalid SQL/MP output. For instance, an Enscribe DDL data type of TYPE BINARY 64,-16 will generate a corresponding SQL/MP type of NUMERIC(18, -16).

40) SBDDLUTL traps if supplied an input file it cannot open (e.g. if the input file does not exist, or if SBDDLUTL does not have permission to open it)

- 41) When VARCHAR fields are included in the primary key of a SQL/MP table with AUDITCOMPRESS on, SQL/MP will pad VARCHAR columns in the WHERE clause of UPDATE statements with spaces, up to the configured size of the VARCHAR. This can result in data corruption.
- 42) MP DCR name mapping does not currently have awareness of the relationship between physical and virtual drives, i.e. when performing name mapping, SBSQLCI does not know ahead of time if the mapping performed for the PHYSVOL clause will generate a command with a physical volume that is in the same storage pool as the virtual volume used in the rest of the command. For example:

Given the following name mapping definitions:

```
MAP NAME $VDRV1.*.* TO $VDRV1B.*.*
MAP NAME $QA TO $QAX
```

The following CREATE TABLE statement will cause the target MPSEVER consumer to fail:

```
CREATE TABLE $VDRV1.TEST.NEWTAB (
    ...
)
CATALOG $VOL.CAT
PHYSVOL $QA;
```

The failure will occur if the physical drive \$QAX is not within the same storage pool as the virtual drive \$VDRV1B.

The work-around here is to just be conscientious about what name mappings are configured for the MP DCR environment if you're going to be running commands with the PHYSVOL clause. Check the storage pools on the target node ahead of time to make sure you properly configure name mapping. To do so:

```
$VOL SUBVOL 10> SCF
SCF - T9082H01 - (01MAY18) (02MAR18) - 08/02/2019 17:56:25 System \GRAVIC1
(c) Copyright 1986-2016 Hewlett Packard Enterprise Development LP
(Invoking \GRAVIC1.$SYSTEM.SYSTEM.SCFCSTM)
1- INFO POOL $*, DETAIL
```

- 43) When generating an explain plan using SOLVMX, certain conditions can cause the explain output to contain this error: '\*\*\* ERROR[2105] This query could not be compiled because of incompatible Control Query Shape (CQS) specifications. Inspect the CQS in effect.' This is the result of the sql compiler selecting an

invalid SPLIT\_TOP plan from the supplied select statements and CQS generated by SOLVMX. This problem does not seem to affect the SOLVMX load.

- 44) The MPSEVER consumer will not back out target-side transactions following a PAUSE that happens within the transaction/session.

For now, it is recommended to either avoid running multiple commands under a single explicit transaction on the source, or avoid using MPERRORACTION PAUSE until this issue is resolved.

- 45) Running an environment with some form of coordination (DCR coordination / DDL coordination), a Queue Manager on any thread, and TRACKTXFILE(S) has the potential to lock up the environment in some scenarios. It is recommended to avoid this by doing one of the following things:

- Not using a QMGR in the Shadowbase environment if MP DCR or DDL Replication with coordination will occur under the same AUDMON
- Turning MPCOORDINATION OFF for MP DCR or turning DDL coordination off (SET DBS CREATES NOCOORDINATION) for DDL replication
- Using TID files with TIDFILECONSTANT ON instead of TRACKTXFILE

- 46) SBSQLCI currently has a problem parsing SQL DDL commands that contain SQL/MP reserved/key words as part of column/file names. For instance, any place in a command that could take any user-specified data value cannot be a reserved/key word. Example:

```
SBSQLCI 1> CREATE TABLE DCRSRC.PARTTB (
mul(1, 2)> ID INT NOT NULL,
mul(1, 3)> NAME CHAR(30),
mul(1, 4)> PRIMARY KEY(ID))
mul(1, 5)> CATALOG $DATA9.VPCAT
mul(1, 6)> PARTITION (
mul(1, 7)> $VDV001.DCRSRC.PARTTB FIRST KEY(1001) CATALOG $DATA9.VPCAT,
mul(1, 8)> $VDV002.DCRSRC.PARTTB FIRST KEY(2001) CATALOG $DATA9.VPCAT,
mul(1, 9)> $VDV003.DCRSRC.PARTTB FIRST KEY(3001) CATALOG $DATA9.VPCAT
mul(1,10)> );
-- SQL operation complete.

SBSQLCI 2> CREATE INDEX DCRSRC.PARTIND1 ON DCRSRC.PARTTB (ID, NAME) PARTITION (
mul(2, 2)> $VDV001.DCRSRC.PARTIND1 FIRST KEY(1001) CATALOG $DATA9.VPCAT,
mul(2, 3)> $VDV002.DCRSRC.PARTIND1 FIRST KEY(2001) CATALOG $DATA9.VPCAT,
mul(2, 4)> $VDV003.DCRSRC.PARTIND1 FIRST KEY(3001) CATALOG $DATA9.VPCAT);
...DCRSRC.PARTTB (ID, NAME) PARTITION ( $VDV...
^
SBSQLCI ERROR: syntax error (unexpected "NAME")
```

In the preceding example, the column name in an index column list cannot be NAME because NAME is a SQL/MP reserved/key word (in this case in a WITH SHARED ACCESS clause).

- 47) The QMGR occasionally abends on a Shadowbase restart with a message similar to the following:

```
19-06-26 10:56:03 \H2.$CJCQM          *GRAVIC.100.V66          004530 QMGR
QMGR-SRC-CJCQM(\H2.$CJCQM): COLDSTART REQUIRED: UNABLE TO FIND FIRST
EXPECTED EVENT IN THE RESTART IPC; ADT POSITION (AA000001,282391212); MAT
POSITION (1, 282423416); MATCH MAT ONLY (FALSE); FIRST MAT POSITION IN IPC
(1, 282423416); LAST MAT POSITION IN IPC (, )
```

Allowing Shadowbase to restart the QMGR will usually clear up this issue.

- 48) AUDMON will not allow a DIRECT environment to be configured without any DBSs and Shadowbase will fail to start if this is done. This may impact DCR-only environments where the only desired replication thread is the DCR thread, and no DBSs are needed. To work around this issue, add a DBS to the \*dir configuration script. The source and target table specifications can be anything as this DBS is not effectively in use.
- 49) When SBSQLCI evaluates a TACL define contained within a SQL DDL statement, it does not remove the node name from the statement. This will cause a problem if replicating DCR events to systems that have a different node name than the source system. A work around is to use table/catalog/physvol names directly instead of TACL defines.
- 50) SBSQLCI's config file settings CTRL\_FILENAME and CONFIG\_FILENAME override the =SBCONTROLFILE and =SBSQLCICONFIG TACL defines. A work-around to this is to delete the CTRL\_FILENAME and CONFIG\_FILENAME parameters from the config file prior to defining the =SBCONTROLFILE and =SBSQLCICONFIG TACL defines.
- 51) The MPSEVER consumer does not currently have the ability to process certain SQL DDL commands such as ALTER or DROP when replicating MP DCR commands on un-audited target tables. The user will see a SQL error 1353 when attempting to UPDATE or DROP a SQL table that is not audited and perform the actions dictated by the MPERRORACTION parameter. To work around this issue:
- MPERRORACTION is set to SKIP, either
    - Alter the given table to turn audit ON in a separate SQLCI session, then re-execute the DDL commands via SBSQLCI to have them re-replicate, or
    - Manually apply the DDL commands to the target table in a separate SQLCI session
  - MPERRORACTION is set to PAUSE: ALTER the table to turn AUDIT ON in a separate SQLCI session, then do RESUMEDDL RETRY in AUDCOM to retry the command
  - MPERRORACTION is set to STOP: either manually enter the DDL commands in a separate SQLCI session, or ALTER the table to turn audit ON, then restart Shadowbase

- 52) The use of the NEXTDOCTIME and NEXTDOCTRIGGER collector parameters causes several issues in the DOC rolling logic:
- The DOC rolls repeat once per second, even if NEXTDOCTRIGGER is set to TIMEOFDAY
  - The NEXTDOCTIME parameter is ignored, and the DOC rolls begin as soon as the Shadowbase environment is started
  - The EMS message pertaining to the initiated DOC roll shows a timestamp that is off by several hours; between 2 and 6 hours seen in internal testing.
- 53) Due to existing volume recovery logic, the Queue Manager holds onto transaction abort records until it sees another event without the volume recovery flag set (typically an IUD event.) In some cases, when no IUD events are sent down the Queue Manager's thread for long enough (12 hours), the Queue Manager will clear these transactions and output an EMS message saying that transactions are being "age cleared." However, the Queue Manager still acknowledges to the collector that the abort event is stored, so the collector's restart point is NOT held back for the 12 hour duration. In this scenario, either a subsequent replicated event without the volume recovery flag, or the eventual age clear successfully cleans up the transaction from memory without causing any issues.
- 54) The use of DBS parameters FETCHSOURCE and INSERTNOTFOUND for DBSs containing key-sequenced Enscribe files with a key-offset of 1 causes issues with auditcompressed updates. Under such circumstances, Shadowbase would remove the offset in the target file, replacing the first non-key byte with the leftmost byte of the key, modifying the primary key value. This causes INSERTNOTFOUND to trigger, causing Shadowbase to effectively double the target file. This behavior has only been observed with Enscribe files.
- The work-around for this is to set the FETCHSOURCE DBS parameter to (ON, SOURCE).
- 55) SBDDLUTL traps when supplied a non-existent input file.
- 56) Shadowbase does not currently support replication of SQL/MX VARCHAR2, BINARY, VARBINARY, and DATE (type2) data types.
- 57) The REPLICATION collector parameter cannot be altered from ON to SUSPEND in a bidirectional replication environment, specifically for AUDCOLLN, while the environment is running. The work-around for this is to shut down Shadowbase, alter the parameter, and re-start Shadowbase.
- 58) In replication environments between tables where primary keys differ and no user-exit is supplied, Shadowbase creates a faulty WHERE clause for non-auditcompressed UPDATE events. The WHERE clause includes only the primary key columns that exist in both the source and target tables, which can cause unintended effects if the target table has columns in the primary key the source is

missing (e.g. the update might affect multiple rows).

DBS mapping can be used as a work-around by using the FLD++ syntax to move the addition source table primary keys not present on the target into the WHERE clause. For example, in a source table containing keys C1, C2, and C3, where only key C1 is shared between the source and target table, the following DBS mapping will force the update statement to contain all three source table key columns in the WHERE clause:

```
[DBS+]U:DBS-01  
[FLD++]C2=%C2%  
[FLD++]C3=%C3%
```

59) As of this release, Shadowbase does not officially support Enscribe DDL dictionaries/schemas generated by the DDL2 program. This limitation impacts several products including the Consumer (AUDCONSN), the Shadowbase Mapping Utility (SBMAP), the Shadowbase DDL Utility (SBDDLUTL), and possibly others. At this time it is advised to generate Enscribe DDL schemas using the original DDL compiler.

60) SOLV does not support the use of the PARTONLYFNAME parameter in conjunction with partition key breaks whose data type does not match the type of the primary key column(s). For instance, if a table is defined with a primary key column of TYPE DATETIME YEAR TO FRACTION(6) and is partitioned with FIRST KEY values of type DATETIME YEAR TO DAY, then the SOLV load would fail with SQL error - 7011:

The two date-time (DATE, TIME, TIMESTAMP, or DATETIME) operands in this arithmetic or relational expression do not have the same range of DATETIME fields.

The work-around for this issue is to remove the use of PARTONLYFNAME and instead prefer to use the SQLWHERE parameter instead to manually specify the key ranges for the SOLV load.

61) There is an existing issue in Shadowbase with dynamically adding replication threads to a running NSK -> Other Servers configuration. Shadowbase will allow you to run the necessary SET, ADD, START, and RUN commands within AUDCOM to make the changes to the environment, and the newly added objects can be seen within the STATUS AUD output and INFO DBS output, however, the collector will not recognize the new replication thread and will not select events for the newly added DBS. The work-around for this is to stop and re-start the collector. Note that this is not an issue for NSK -> NSK environments.

62) In a replication environment where multiple DBSes have the same TARGETFILE parameter, but different TARGETREC parameters, the first TARGETREC parameter will be used for all DBSes with that TARGETFILE. The work-around for this is to use a different TARGETFILE for each DBS.

63) In a replication environment with QMGRs configured, when the environment becomes idle and there are no events for it to replicate, the LAG value in the STATS QMGR and STATS CONS output may show large values or start to grow over time. This is expected because these LAG values are calculated when events are replicated. When there are no events to replicate, these large LAG values can be misleading and make it seem that the environment is not performing properly. As long as the LAG in the STATS COLL output is low, and the number of events queued in the QMGR is not increasing, there should be no cause for alarm in this case.

**[\*\*\*\*\* End of Document \*\*\*\*\*]**