GemStone/S 64 Bit™
DRAFT Release Notes

Version 3.7.5

October 2025

This document is a DRAFT version of 9/15/25, and may change substantially before release.
You must review the final version at time of release for updates and corrections. Doc rev 258.2

INTELLECTUAL PROPERTY OWNERSHIP

This documentation is furnished for informational use only and is subject to change without notice. GemTalk Systems LLC assumes no responsibility or liability for any errors or inaccuracies that may appear in this documentation.

Warning: This computer program and its documentation are protected by copyright law and international treaties. Any unauthorized copying or distribution of this program, its documentation, or any portion of it, may result in severe civil and criminal penalties, and will be prosecuted under the maximum extent possible under the law.

The software installed in accordance with this documentation is copyrighted and licensed by GemTalk Systems under separate license agreement. This software may only be used pursuant to the terms and conditions of such license agreement. Any other use may be a violation of law.

Use, duplication, or disclosure by the Government is subject to restrictions set forth in the Commercial Software - Restricted Rights clause at 52.227-19 of the Federal Acquisitions Regulations (48 CFR 52.227-19) except that the government agency shall not have the right to disclose this software to support service contractors or their subcontractors without the prior written consent of GemTalk Systems.

This software is provided by GemTalk Systems LLC and contributors "as is" and any expressed or implied warranties, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose are disclaimed. In no event shall GemTalk Systems LLC or any contributors be liable for any direct, indirect, incidental, special, exemplary, or consequential damages (including, but not limited to, procurement of substitute goods or services; loss of use, data, or profits; or business interruption) however caused and on any theory of liability, whether in contract, strict liability, or tort (including negligence or otherwise) arising in any way out of the use of this software, even if advised of the possibility of such damage.

COPYRIGHTS

PATENTS

GemStone software has been covered by U.S. Patent Number 6,256,637 "Transactional virtual machine architecture", Patent Number 6,360,219 "Object queues with concurrent updating", Patent Number 6,567,905 "Generational garbage collector with persistent object cache", and Patent Number 6,681,226 "Selective pessimistic locking for a concurrently updateable database".

TRADEMARKS

GemTalk, **GemStone**, **GemBuilder**, **GemConnect**, and the GemStone and GemTalk logos are trademarks or registered trademarks of GemTalk Systems LLC, or of VMware, Inc., previously of GemStone Systems, Inc.

UNIX is a registered trademark of The Open Group.

Intel is a registered trademarks of Intel Corporation.

Microsoft, **Windows**, **Windows Server**, and **Azure** are registered trademarks of Microsoft Corporation.

Linux is a registered trademark of Linus Torvalds and others.
Red Hat, **Red Hat Enterprise Linux**, **RHEL**, and **CentOS** are trademarks or registered trademarks of Red Hat, Inc.
Rocky Linux is a trademark or registered trademark of Rocky Enterprise Software Foundation.
Ubuntu is a registered trademark of Canonical Ltd., Inc.

AIX, **Power**, **POWER**, **Power8**, **Power9**, and **VisualAge** are trademarks or registered trademarks of International Business Machines Corporation.

Apple, **Mac**, m**acOS**, and **Macintosh** are trademarks of Apple Inc.

Instantiations is a registered trademarks of Instantiations, Inc.

CINCOM, **Cincom Smalltalk**, and **VisualWorks** are trademarks or registered trademarks of Cincom Systems, Inc.

Raspberry Pi is a trademark of the Raspberry Pi Foundation.
RabbitMQ is a trademark of VMware, Inc.
Prometheus is a registered trademark of The Linux Foundation.
Grafana is a registered trademark of Raintank, Inc. dba Grafana Labs.

Other company or product names mentioned herein may be trademarks or registered trademarks of their respective owners. Trademark specifications are subject to change without notice. GemTalk Systems cannot attest to the accuracy of all trademark information. Use of a term in this documentation should not be regarded as affecting the validity of any trademark or service mark.

GemTalk Systems
15220 NW Greenbrier Parkway
Suite 240
Beaverton, OR 97006

Preface

About This Documentation

These release notes describe changes in the GemStone/S 64 Bit™ version 3.7.5 release. Read these release notes carefully before you begin installation, upgrade, or development with this release.

For information on installing or upgrading to this version of GemStone/S 64 Bit, please refer to the *GemStone/S 64 Bit Installation Guide* for version 3.7.5.

Terminology Conventions

The term "GemStone" is used to refer to the server products GemStone/S 64 Bit and GemStone/S, and the GemStone family of products; the GemStone Smalltalk programming language; and may also be used to refer to the company, now GemTalk Systems LLC, previously GemStone Systems, Inc. and a division of VMware, Inc.

Technical Support

Support Website

gemtalksystems.com

GemTalk's website provides a variety of resources to help you use GemTalk products:

- **Documentation** for the current and for previous released versions of all GemTalk products, in PDF form.
- **Product download** for the current and selected recent versions of GemTalk software.
- **Bugnotes**, identifying performance issues or error conditions that you may encounter when using a GemTalk product.

- **Supplemental Documentation** and **TechTips**, providing information and instructions that are not in the regular documentation.
- **Compatibility matrices**, listing supported platforms for GemTalk product versions.

We recommend checking this site on a regular basis for the latest updates.

Help Requests

GemTalk Technical Support is limited to customers with current support contracts. Requests for technical assistance may be submitted online (including by email), or by telephone. We recommend you use telephone contact only for urgent requests that require immediate evaluation, such as a production system down. The support website is the preferred way to contact Technical Support.

Website: techsupport.gemtalksystems.com

Email: techsupport@gemtalksystems.com

Telephone: (800) 243-4772 or (503) 766-4702

Please include the following, in addition to a description of the issue:

- The versions of GemStone/S 64 Bit and of all related GemTalk products, and of any other related products, such as client Smalltalk products, and the operating system and version you are using.
- Exact error message received, if any, including log files and statmonitor data if appropriate.

Technical Support is available from 8am to 5pm Pacific Time, Monday through Friday, excluding GemTalk holidays.

24x7 Emergency Technical Support

GemTalk offers, at an additional charge, 24x7 emergency technical support. This support entitles customers to contact us 24 hours a day, 7 days a week, 365 days a year, for issues impacting a production system. For more details, contact GemTalk Support Renewals.

Training and Consulting

GemTalk Professional Services provide consulting to help you succeed with GemStone products. Training for GemStone/S is available at your location, and training courses are offered periodically at our offices in Beaverton, Oregon. Contact GemTalk Professional Services for more details or to obtain consulting services.

Table of Contents

Chapter

1 DRAFT Release Notes for 3.7.5

Overview

GemStone/S 64 Bit™ 3.7.5 is a new release of the GemStone/S 64 Bit object server, fixing several critical bugs and with other changes, minor new features, and bug fixes.

These Release Notes include changes between the previous version of GemStone/S 64 Bit, v3.7.4.3, and 3.7.5. If you are upgrading from a version prior to 3.7.4.3, review the release notes for each intermediate release to see the full set of changes.

For details about installing GemStone/S 64 Bit 3.7.5 or upgrading from earlier versions of GemStone/S 64 Bit, see the *GemStone/S 64 Bit Installation Guide* for v3.7.5 for your platform.

NOTE: These Release Notes are a draft version and are not complete with respect to changes in this release, and may contain inaccuracies. Updated Installation Guides are not yet available, nor VSD 5.6.4 distribution or Release Notes.

Supported Platforms

Platforms for Version 3.7.5

GemStone/S 64 Bit version 3.7.5 is supported on the following platforms:

- Red Hat-compatible Linux 7.9, 8.10, 9.5, and 10.0, and Ubuntu 22.04 and 24.04, on x86_64.
  GemStone is tested on a mixture of Red Hat, CentOS, and Rocky; these are all considered fully certified platforms. Any reference to Red Hat applies to any Red Hat-compatible distribution.
- Ubuntu 24.04 on ARM (Raspberry Pi 5). Linux on ARM is for development only, not for production.
- macOS 15.3.2 (Sequoia) and 11.7.10 (Big Sur), on x86 and Apple silicon (ARM). macOS distributions are for development only, not for production.

For more information and detailed requirements for each supported platforms, please refer to the *GemStone/S 64 Bit v3.7.5 Installation Guide* for that platform.

GemBuilder for Smalltalk (GBS) Versions

The following versions of GBS are supported with GemStone/S 64 Bit version 3.7.5:

GBS/VW version 8.8.1

VisualWorks 9.4.1 64-bit	VisualWorks 9.3.1 64-bit	VisualWorks 8.3.2 64-bit	VisualWorks 8.3.2 32-bit
▸ Windows 10, Windows 11 ▸ RedHat ES 7.9, 8.10, 9.5, and 10.0; Ubuntu 22.04 and 24.04	▸ Windows 10, Windows 11 ▸ RedHat ES 7.9, 8.10, 9.5, and 10.0; Ubuntu 22.04 and 24.04	▸ Windows 10, Windows 11 ▸ RedHat ES 7.9, 8.10, 9.5, and 10.0; Ubuntu 22.04 and 24.04	▸ Windows 11

GBS/VA version 5.4.7

VAST Platform 13.0.0	VAST Platform 11.0.1	VA Smalltalk 8.6.3
▸ Windows Server 2016 and Windows 11	▸ Windows Server 2016, Windows 11, and Windows 10	▸ Windows Server 2016, Windows 101 and Windows 10

For more details on GBS and client Smalltalk platforms and requirements, see the *GemBuilder for Smalltalk Installation Guide* for that version of GBS.

VSD Version

The GemStone/S 64 Bit v3.7.5 distribution includes VSD version 5.6.4. The previous release, v3.7.4.3, contained VSD v5.6.1. The difference includes bug fixes, a minor new feature, and updated statistics. For details on the changes, see the Release Notes for VSD v5.6.4.

VSD 5.6.4 is included with the GemStone distribution, and can also be downloaded as a separate product from https://gemtalksystems.com/vsd/

Changes in this Release

Updated shared library versions

The version of zlib has been updated from 1.3 to 1.3.1

Support for log rotation for application log files

In v3.7.4, support was added for system log file rotation, including gem and server process logs, using the SIGHUP signal. The general rotation process is: to first move (or delete) the existing disk log file, then send SIGHUP to the process, which will start writing to a new log file with the previous name and mode.

This process can now be applied to files opened by the application using GsFile or GsLog (for FileSystem files).

Added method GsFile >> reopen

The following method has been added:

```
GsFile >> reopen
```

If the underlying disk file no longer present, GsFile creates a new disk file with the same name and path and returns true. Reopen is done using freopen, per man 3 fopen, using the existing path and mode. If the disk file still exists, nothing is done and this method returns false. If an error occurs, this method returns nil. Receiver must be a server file, with mode containing 'a' or 'w', and cannot be stdin, stdout, or stderr.

Added class GsLog

GsLog is a new class in v3.7.5, which provides a mechanism to do SIGHUP-based file rotation with a FileSystem-based log file. GsLog implements #reopen

An instance of GsLog encapsulates a FileReference and a ZnCharacterWriteStream, allowing the underlying file to be reopened by a LogRotateNotification handler. GsLog is thread-safe. Writes to the log file are protected by a mutex. Reopen tries to lock the mutex and signals a Warning it cannot lock. If a file exists during initial open or during a reopen, the existing file is appended to. If the specified file does not exist, a new file is created.

Note that there is a risk of deadlock when file writes can occur during Ephemeron mourning. See the class comments for GsLog for more information.

Added class LogRotateNotification

The class LogRotateNotification has been added. An instance of this class is signalled asynchronously when the Gem process or topaz -l receives a SIGHUP. By default, it is not enabled. If enabled, it does not need to be reenabled after a signal is received. When enabled, handler for this notification should be installed that calls #reopen on each GsFile or GsLog that is configured for log rotation.

Log rotation process

The application sets up a handler for the LogRotateNotification signal (using `addDefaultHandler:`). This handler sends `reopen` to the file being rotated. If more than one file is rotated, even if the rotation is on different schedules, the handler should reopen all of these files. A reopen is a no-op if the file was not moved or deleted.

At the OS level, log rotation of application GsFiles should move the disk file to an archive location, and send SIGHUP to the Gem.

Note that using **mv** within the local file system ensures no writes can be lost. If a file write occurs within the small gap between log archiving and sending SIGHUP, there is a risk a write may be lost if the archive process does a log copy and deletes the old log, including **mv** to a non-local file system destination.

Example handler

In this example, `myGsFile` is a GsFile and `myGsLog` is an GsLog, both of which are open for write or append. The following handler reopens both.

```
LogRotateNotification enableSignalling.
LogRotateNotification addDefaultHandler: [:ex |
   myGsFile reopen ifNil: [
      self error: 'reopen failed ', myGsFile pathName].
   myGsLog reopen.
   ex resume.
   ].
```

Changes in Tranlog Restore

Tranlog restore does not continue if session terminated

In previous releases, when the session that initiated restore from tranlogs was terminated, the restore continued until all specified tranlogs were restored. Now, terminating the session will stop tranlog restore at that point. Restore can be continued by logging in and re-executing the tranlog restore operation.

GcReclaim Gem threads for tranlog restore

The method `Repository >> restoreFromBackup:*` shuts down the reclaimGem, and when it completes, restarts the reclaimGem with an increased number of threads, to ensure reclaim does not hold up tranlog restore. Previously, the system computed the number of threads, which could be considerably larger and did not respect STN_MAX_GC_RECLAIM_SESSIONS. Now, the maximum number of threads is the configured STN_MAX_GC_RECLAIM_SESSIONS or 4, whichever is larger.

You should review your configuration of STN_MAX_GC_RECLAIM_SESSIONS, to ensure the setting is appropriate for your application. This is most important for repositories performing a large volume of tranlog restore.

Excess ReclaimGem threads incorrectly left running after restore

After the number of reclaim threads was increased following a programmatic `restoreFromBackup*`, the extra threads were left running until the next Stone restart. (#51269)

One-step restore and commitRestore

After a programmatic `restoreFromBackup*` (of a repository in full tranlogging mode), you must login again to execute `commitRestore`. In cases where you do not need to restore tranlogs, the restore process can be simplified with the added method `disableRestoreFromLogs`. After executing this method (in both encrypted as well as non encrypted extent systems), the `restoreFromBackup` skips restoring tranlogs and automatically performs a `commitRestore`.

Note that any commits that are not in the programmatic backup are lost, since transaction logs are not replayed, and transaction logs cannot be replayed after `commitRestore`.

This method should be executed after login and before the restore operation. The status is set in SessionTemps, and only applies for that restore operation (since restore terminates the session).

`Repository >> disableRestoreFromLogs`
Causes the next restoreFromBackup operation (normal or secure) by the session to immediately commit the restore and return the repository to normal mode. Executing the method #commitRestore is unnecessary.

Add fullBackup method variants

The following methods have been added:

`Repository >> fullBackupGzCompressedTo:` *fileNames* `MBytes:` *mByteLimit* `threads:` *numThreads*

`Repository >> fullBackupLz4CompressedTo:` *fileNames* `MBytes:` *mByteLimit* `threads:` *numThreads*

These are variants of the existing `fullbackup` methods that provide specification of both extent sizes and the number of threads.

Added method to find all instances with a single reference

A new method has been added to find all objects in the repository that have only a single unique reference.

`Repository >> listObjectsWithOneReferenceWithMaxThreads:` *maxThreads* `waitForLock:` *lockWaitTime* `percentCpuActiveLimit:` *percentCpu*
Scans the entire repository for objects referenced once and only once by any other object, and returns these as a GsBitmap.

Multiple references from a parent object to a child object are considered a single references and the the child object will be included in the result. Objects which reference themselves will be included in the result if that reference is the only one.

This method begins a transaction and runs in this transaction for its duration. It should therefore not be used in production systems due to commit record backlogs which may cause excessive repository growth.

Uncommitted objects and dead not reclaimed objects are excluded from the result. Note that objects in the result set may be disconnected from the repository (unreachable) and therefore could disappear after the next garbage collection cycle.

Raises an error if the session has modified persistent objects. Raises an error if a garbage collection operation is in progress with the repository vote state is not zero

(see `System class >> voteState`). The *lockWaitTime* argument is used to specify how many seconds method should wait while attempting to acquire the gcLock. No other garbage collection operations may be started or in progress while this method is running. There also must be no outstanding possible dead objects in the system for the GC lock to be granted.

Starts *maxThreads* on the host system and allows the host to run up to *percentCpu* percent CPU usage. A page buffer of 16 pages (256 KB) is allocated per thread.

Added method to search GsBitmap by class

The following method has been added:

`GsBitmap >> allCommittedInstancesOf:` *aClass*
Searches the receiver for all committed objects which are instances of *aClass*, and returns a new GsBitmap containing those instances. Uncommitted objects present in the receiver are ignored, and not included in the result.

GsTsExternalSession improved handling of String/Unicode string results

When code executed by a GsTsExternalSession returns a kind of string, the external session automatically convert the result into a string in the calling session.

Previously, only remote objects of type String or Unicode7 were correctly converted, since these hold Characters requiring no more than 8 bits. The behavior for these objects is unchanged; the local object created for the remote object is of the same class as the remote object.

Remote objects of DoubleByteString, QuadByteString, Unicode16 and Unicode32 were previously returned as ByteArrays. Now, these values are automatically encoded into instance of Utf8 prior to returning. This is the same process as the recommended workaround in previous releases.

Instance of Utf8 were, and are, automatically converted to a local object of the appropriate class:

- When Repository is not in Unicode Comparison Mode, a traditional string: String, DoubleByteString, or QuadByteString.
- When Repository is in Unicode Comparison Mode, a Unicode string class: Unicode7, Unicode16, and Unicode32).

Note that when the remote execution's return values are traditional or unicode strings that do not match the default classes for the local repository's Unicode Comparison Mode (most commonly if the remote repository has a different Unicode Comparison Mode than the local repository), the class of the result in the local repository may vary if some require encoding and others do not.

If you previously were not using the workaround, but doing the multi-step process required to decode the ByteArray into a kind of string, this will no longer work; you will need to remove the manual decode.

Note that legacy GsExternalSession has no change in behavior.

Compile problems when GsTsExternalSession executes source in Utf8

When the input to a GsTsExternalSession execute is a Utf8, the compiler may attempt to parse beyond the end of the decoded input. This results in a compiler error (#51543)

Error during GciError >> signalCompileError when compile error occurs in GsTsExternalSession

When a compile error occurs in the external session code, during `resolveResult:` it can encounter an error while processing the error. (#51563)

ClassOrganizer support for class categories

Methods have been added to provide information about class categories. These methods accept and return kinds of String, since they are designed to support GUI tools. These category methods treat hyphen-delimited class category strings as representing a hierarchal structure and the added methods are provided to query both for specific full category strings and association with logically-inherited supercategories.

`ClassOrganizer >> classCategoryNames`

`ClassOrganizer >> allClassCategoryNames`

`ClassOrganizer >> classCategoryNamesInDictionaryName:` *dictionaryName*

`ClassOrganizer >> allClassCategoryNamesInDictionaryName:` *dictionaryName*

`ClassOrganizer >> classNamesInClassCategoryNamed:` *categoryString* `dictionaryName:` *dictionaryName*

`ClassOrganizer >> classNamesUnderClassCategoryNamed:` *categoryString* `dictionaryName:` *dictionaryName*

See the image method comments for more details.

Changes in STON support

STON is an object interchange format used by Rowan. Support has been added for FileReference, Path, Association, Class, Metaclass3, Fraction, and ScaledDecimal; instance of these classes previously were encoded incorrectly.

The STONFileReference class as been added to the image.

Removed methods

The following methods have been removed:

`GsSingleRefPathFinder >> printTimestampToLog`

`ProcessorScheduler >> dbgfatallog:`

`Repository >> _getShrinkRepository`

`Repository >> _primRestoreSecureBackups:scavPercentFree:bufSize:privateDecryptionKey:passphrase:numThreads:shrinkRepos:newSystemUserPassword:`

`Repository >> _restoreBackups:scavPercentFree:bufSize:numThreads:shrinkRepos:newSystemUserPassword:`

Topaz Changes

Added topaz -P command line argument

Topaz now supports the **-P** argument. This is similar to **-S**, but while **-S** inputs the script file argument and exits if no error occurs, **-P** inputs the script file argument but does not exit.

When using **-P**, **quit** and **exit** are considered errors if they occur in an input file, other than the **-P** argument script itself.

Topaz -I did not correctly handle echo when error occurred

Using the -I option to Topaz is designed to allow input of an initialization script. With this option, echo is suppressed unless an error occurs; when an error occurs, the entire output is printed, but passwords are redacted.

- When the script contained an explicit EXIT and an error occurred, the echo output was not printed, so specifics of the error were not available for error diagnosis. (#51544)
- With nested scripts when an error occurred, the output was printed, but the password was not redacted in all cases. (#51545, #51546)

ExitClientError handling

An instance of ExitClientError can be signalled when the application wants topaz or other GCI client to exit with a specific error status.

Now, if ExitClientError is signalled to the GCI of an interactive linked topaz process, and the linked topaz was configured with GEM_LISTEN_FOR_DEBUG=true in a configuration file or on the command line, or if System listenForDebugConnection was executed, then topaz will stop at the command line, to allow the ExitClientError to be debugged, and not exit.

Automatic resultcheck

If the environment variable GS_TOPAZ_AUTO_RESULTCHECK is defined in topaz's environment, then result checking is automatically enabled, equivalent to DISPLAY RESULTCHECK. See the help text for DISPLAY RESULTCHECK for more details.

New shared library to support statprom utility

The **statprom** utility provides an interface to use Prometheus to monitor GemStone by recording statistics from the GemStone shared page cache. In this version, **statprom** code is packaged in a new shared library, `libstatprom-3.7.5-64.so`. This simplifies installation in some configurations.

There are no changes in behavior or configuration.

Changes in GCI errors

The error corresponding to LogRotationNotification has been added:

```
RT_ERR_SIGHUP/6026
```

The following errors, related to internal errors in X509-secured GemStone, have been added:

```
ERR_REBUILD_SCAVENGABLE/4022
ERR_DEPMAP_FAILURE/4023
ERR_OT_AUGMENT_FAILURE/4024
ERR_COMPOSE_CR_FAILURE/4025
```

The long-obsolete error `GS_ERR_SHRPC_INVALID_CONFIG/4011` has been removed.

Cache Statistics Changes

New stat for total Gem memory use

GemMemoryFootPrintKb (Gem)
Approximate total memory footprint of allocated temp obj memory.

New stats for huge memory pages

Previously, Linux system statistics for HugePages did not differentiate between different memory page sizes. These statistics have been replaced by new ones that include the page size in the name.

The following statitics have been removed:

HugePagesTotalKB
HugePagesFreeKB
HugePagesRsvdKB
HugePagesSurpKB
HugePageSizeKB

The following statistics have been added. Note that the unit of these statistics is pages, not KB/MB.

HugePages2MbTotal
The total number of 2 Mb huge memory pages configured.

HugePages2MbFree
The number of free 2 Mb huge memory pages.

HugePages2MbRsvd
The number of reserved 2 Mb huge memory pages.

HugePages2MbSurp
The number of surplus 2 Mb huge memory pages.

HugePages1GbTotal
The total number of 1 Gb huge memory pages configured.

HugePages1GbFree
The number of free 1 Gb huge memory pages.

HugePages1GbRsvd
The number of reserved 1 Gb huge memory pages.

HugePages1GbSurp
The number of surplus 1 Gb huge memory pages.

Linux ARM supports, in addition to 2MB and 1 GB huge pages, the additional huge memory page sizes 64 KB and 32 MB. The following are reported on Linux/ARM only:

HugePages64KbTotal
The total number of 64 Kb huge memory pages configured.

HugePages64KbFree
The number of free 64 Kb huge memory pages.

HugePages64KbRsvd
The number of reserved 64 Kb huge memory pages.

HugePages64KbSurp
The number of surplus 64 Kb huge memory pages.

HugePages32MbTotal
The total number of 32 Mb huge memory pages configured.

HugePages32MbFree
The number of free 32 Mb huge memory pages.

HugePages32MbRsvd
The number of reserved 32 Mb huge memory pages.

HugePages32MbSurp
The number of surplus 32 Mb huge memory pages.

Bugs Fixed

Printing the Smalltalk stacks did not handle fatal errors properly

When printing the smalltalk stack using pstack or kill -USR1, if the Gem had a fatal error, such as termination due to a lostOT, this was not handled correctly; gdb may fail to detach, hanging the gem. (#51478).

SIGUSR1 is also now ignored if sent less than 2 seconds since the previous such signal.

Upgrade cleared Deprecated action

The Deprecated class supports automatically writing to a log or erroring if a deprecated method is executed, as well as doing nothing (the default). Upgrade cleared this status to remove a non default deprecated action. Now, upgrading a repository with a non-default deprecated action preserves the deprecated action (#51520)

GsObjectSecurityPolicy dynamicInstanceVariables failed on an upgraded repository

Sending `#dynamicInstanceVariables` to an instance of GsObjectSecurityPolicy in a repository upgraded from v3.5.6 or earlier reported a corrupt object error. (#51411).

When stopstone was prompting for input, it did not respond to Control-C

When stopstone is executed without arguments, it prompts for the Stone name, user name, and password. While in the prompt, the process did not respond to control-C to terminate. (#51428)

Risk of lostOT during reclaimAll

`Repository >> reclaimAll` waited for an increasing time when no progress in reclaim was occurred. This created a risk of a lostOT when the ReclaimGem was committing at a high rate. (#51506)

Issues with detached execution Gems

Detached execution Gems are started using `GsTsExternanlSession forkAndDetach*` methods.

Detach execution Gem did not handle DEBUGGEM

When a Gem is running with detached execution, and topaz DEBUGGEM is used to attach to that Gem, the resume command cannot be used to resume; the detached execution Gem must be terminated. (#51457)

Detached execution handled compiler errors were logged

When a Gem is running with detached execution, unhandled errors cause stacks to be written to the log of the detached execution Gem. Previously, CompileErrors with handlers were excepted, and were logged in spite of the handler. Now, a CompileError that is caught and handled will not have a stack printed in the Gem log.

Issues related to Hot Standby

Risk of Gem protocol error if session is signaled after logsenderSessionId

After executing `logsenderSessionId`, interrupt handling may sometimes result in a protocol error, 'Unexpected packet received from Stone'. (#41448)

stoplogreceiver may have silently failed to stop the logreceiver

It is possible for **stoplogreceiver** to return 0 and does not print an error, but to leave the logreceiver process running. (#51472)

Restarting tranlog restore after a stopContinuousRestore has risk of error requiring Stone restart

If `stopContinuousRestore` is executed while the Stone thread running recovery is processing a large work queue, there is a risk that a subsequent `continuousRestoreFromArchiveLogs:`, or `restoreFromArchiveLogs:`, may fail with a detected Fork in Time error. To continue, the Stone must be restarted. (#51548)

RSR ignored TransactionBacklog

The RemoteServiceReplication open-source project that is included with GemStone for use by other open-source projects. This framework only handled the GsFinalizeNotification asynchronous signal; the TransactionBacklog signal was ignored. Now, it will respond to a TransactionBackup by aborting. (#51433)

To control TransactionBacklog handling in an RSR environment, the application must run its own asynchronous event mechanism, which must handle GsFinalizeNotification as well as TransactionBackup. If GsFinalizeNotification is not handled it may result in deadlocks or hangs.

For more on asynchronous event handling, see `GsSignalingSocket class>>newForAsyncExceptions:`.

Behavior>>removeSelector:ifAbsent: did not complete removal in a Rowan repository

`Behavior>>removeSelector:ifAbsent:` removed the method, but did not remove the loaded method in a Rowan repository. (#49861)

Deleting a UserProfileGroup may fail on upgraded repository with few security policies

A repository upgraded from v2.1 or earlier with fewer than 20 object security policies has nils in the SystemRepository (the collection that holds object security policies) for slots under 20. This causes `UserProfileGroup >> deleteGroup:` to fail. (#51518)

GsFile issues

GsFile opening with mode 'ab+' did not allow read positioning.

When a GsFile is opened for read and append, the mode may be specified either 'a+', 'ab+', or 'a+b'. The specification 'ab+' was not handled correctly; the resulting file was open for append but could not be positioned for read. (#51525)

GsFile open with nil path or mode failed to set error message

GsFile operations that error return nil, rather than signalling an error, and put the error message in the server or client error buffer. This was not done for nil argument cases.

Incorrect values in DateTime instance creation out-of-range error message

When the second or hour argument to `DateAndTimeANSI class>>year:day:hour:minute:second:offset:` was out of range, the error message provided an incorrect upper limit (off by one) (#51523)

systemLocksDetailedReport did not handle in-logout sessions

`System class >> systemLocksDetailedReport` could have reported "session does not exist", if a session logs out during execution of the method. (#51474)

X509-Secured GemStone Changes and Bug Fixes

X509 mid-level cache startup now fails without required configuration

Now, if the NETLDI_PORT_RANGE and other required configuration parameters are not supplied, the mid-level cache in an X508 configuration will not start up. (#51552)

Mid-level caches failed to improve performance in X509 configuration

In X509-secured GemStone, if the page lookup on a leaf host missed, the leaf host did not attempt to read from the mid cache. (#51534).

Mid-level cache warming failed to completely warm the cache

The pusher threads in the HostAgent that warm the mid-level cache did not scan the Stone's entire cache, resulting in an incompletely warmed cache. (#51558)

X509-secured GemStone did not calculate default large page size

If SHR_PAGE_CACHE_LARGE_MEMORY_PAGE_SIZE_MB is not set, and large pages are enabled using SHR_PAGE_CACHE_LARGE_MEMORY_PAGE_POLICY, the default large memory page size on the given host should be used. This was not being done in an x509-Secured GemStone system. (#51423)

HostAgent >> startLeafCache did not respect timeout

It was possible for the HostAgent to create a commit record backlog when the NetLDI on a leaf host failed to respond. The underlying code was not respecting the timeout provided. (#51549)

SEGV in HostAgent after session logout during heavy use of InterSessionSignal

There is a risk of a SEGV in the HostAgent, when the HostAgent closes the SSL connection to the Gem during periods when the Gem can be receiving InterSessionSignals, or when the HostAgent is forwarding a sigAbort to an x509 Gem. (#51564)

Fatal error in HostAgent can leave stuck commit token

When a thread in the HostAgent had a fatal error, it did not correctly handle the exit to ensure the Stone knew the session had disconnected and release the commit token and other resources. (#51566)

Error in mid-cache HostAgent if connection to a leaf host was lost

The mid-cache HostAgent did not correctly handle the case where a leaf cache connection was lost. (#51538)