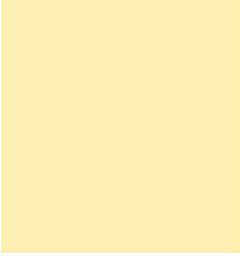


---



# **GemStone/S 64 Bit<sup>TM</sup>**

## **Release Notes**

**Version 3.5.1**

December 2019



## 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

This software product, its documentation, and its user interface © 1986-2019 GemTalk Systems LLC. All rights reserved by GemTalk Systems.

## PATENTS

GemStone software is or has been covered by U.S. Patent Number 6,256,637 "Transactional virtual machine architecture" (1998-2018), Patent Number 6,360,219 "Object queues with concurrent updating" (1998-2018), Patent Number 6,567,905 "Generational garbage collector with persistent object cache" (2001-2021), and Patent Number 6,681,226 "Selective pessimistic locking for a concurrently updateable database" (2001-2021).

## TRADEMARKS

**GemTalk**, **GemStone**, **GemBuilder**, **GemConnect**, and the GemTalk logo are trademarks of GemTalk Systems LLC, or of VMware, Inc., previously of GemStone Systems, Inc., in the United States and other countries.

**UNIX** is a registered trademark of The Open Group in the United States and other countries.

**Solaris**, **Java**, and **Oracle** are trademarks or registered trademarks of Oracle and/or its affiliates. **SPARC** is a registered trademark of SPARC International, Inc.

**Intel** and **Pentium** are registered trademarks of Intel Corporation in the United States and other countries.

**Microsoft**, **Windows**, and **Windows Server** are registered trademarks of Microsoft Corporation in the United States and other countries.

**Linux** is a registered trademark of Linus Torvalds and others.

**Red Hat** and all Red Hat-based trademarks and logos are trademarks or registered trademarks of Red Hat, Inc. in the United States and other countries.

**Ubuntu** is a registered trademark of Canonical Ltd., Inc., in the U.S. and other countries.

**SUSE** is a registered trademark of Novell, Inc. in the United States and other countries.

**AIX**, **POWER6**, **POWER7**, **POWER8** and **VisualAge** are trademarks or registered trademarks of International Business Machines Corporation.

**Apple**, **Mac**, **MacOS**, and **Macintosh** are trademarks of Apple Inc., in the United States and other countries.

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

**Raspberry Pi** is a trademark of the Raspberry Pi Foundation

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 LLC**  
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 3.5.1 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 3.5.1.

## 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](http://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. GemStone users that do not have support contracts may post issues on the customer forum. To join this list, visit [lists.gemtalksystems.com/mailman/listinfo/gemstone-smalltalk](https://lists.gemtalksystems.com/mailman/listinfo/gemstone-smalltalk).

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](https://techsupport.gemtalksystems.com)**

**Email: [techsupport@gemtalksystems.com](mailto: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.

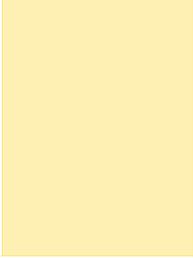
## 24x7 Emergency Technical Support

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. GemStone/S 64 Bit 3.5.1 Release Notes*

Overview . . . . .	9
Supported Platforms . . . . .	9
Platforms for Version 3.5.1 . . . . .	9
GemBuilder for Smalltalk (GBS) Versions . . . . .	10
VSD Version. . . . .	10
Upgrade . . . . .	10
Documentation . . . . .	11
Changes in this release . . . . .	11
Updated library versions . . . . .	11
Distribution change in packing libraries. . . . .	11
Distribution change for Raspberry Pi . . . . .	11
Recommendation to clear /opt/gemstone/locks/ on system restart . . . . .	11
gslist added status unknown(EPERM) . . . . .	11
Support for transparent huge pages on Linux . . . . .	12
Improved Shared Page Cache startup performance . . . . .	13
Improvements in disk space usage after restore from full backup . . . . .	13
The restore of a full backup now reduces the OOP high water mark. . . . .	13
Reduced space used by object table leaf pages after restore . . . . .	13
GsBitmap added methods. . . . .	13
Reporting extents and files in GB or MB. . . . .	14
SymbolDictionary printString change . . . . .	14
Improved reporting on transaction conflicts . . . . .	14
System >> remoteCachesReport now includes Stone's cache . . . . .	14
Added method . . . . .	14
objectAudit warnings on old 32-bit fanout in upgraded repositories . . . . .	15
gslist output in json format . . . . .	15
Return values for startnetldi, stopnetldi, startstone, stopstone . . . . .	16
stopnetldi by root user on any NetLDI. . . . .	16

Disallowed to startstone with an extent of a later version . . . . .	16
copydbf -i/-I of extent file now includes version . . . . .	17
Applications running without \$GEMSTONE . . . . .	17
topaz -S option. . . . .	17
topaz -S argument now does not suppress output. . . . .	17
topaz -S argument fails to work without a .topazini. . . . .	17
topaz now handles BOM input correctly . . . . .	17
Log file locations changes for remote login processes. . . . .	18
Verify your remote distribution log file locations . . . . .	18
startnetldi -X argument added . . . . .	18
Attaching to a running Gem for debugging . . . . .	18
Configuration parameter added to support debug gem . . . . .	19
Method added to support debug gem . . . . .	19
Topaz commands added to support debug gem. . . . .	19
GCI calls added to support debug gem . . . . .	20
Other topaz changes . . . . .	20
Other Changes in Configuration Parameters . . . . .	20
CONFIG_WARNINGS_FATAL now applied from Gem NRS. . . . .	20
Setting Gem environment variables via configuration parameters . . . . .	21
GEM_ABORT_MAX_CRG changed . . . . .	21
RTLD_DEEPBIND no longer used; User Action relink required . . . . .	21
Linking a GCI application on Linux. . . . .	21
Change in linker flags for Solaris . . . . .	21
Linking a user action library on Solaris . . . . .	21
Other Added GCI functions . . . . .	22
Changes in Thread-safe GCI. . . . .	22
GciTsLogin and GciTsForkLogin changed . . . . .	22
GciTsClassRemoveAllMethods changed . . . . .	22
GciTsCompileMethod changed . . . . .	22
GciTsFetchTraversal changed . . . . .	22
Added functions from standard GCI API. . . . .	23
Added functions to support X509 Logins. . . . .	23
Changes in Cache Statistics . . . . .	23
Bug Fixes. . . . .	24
GciStoreTravDoTravRefs issues with serious GBS impact . . . . .	24
GciStoreTravDoTravRefs_ may error, impacting linked GBS sessions . . . . .	24
Missing reports in traversals from GBS . . . . .	24
Terminated GsProcess may not have allowed ensure block to complete . . . . .	24
STN_COMMITS_ASYNC risk of lost record. . . . .	24
GciNewUtf8String does not put result object in export set. . . . .	24
startstone with precheckpoint and checkpoint records in different tranlogs . . . . .	24
libgcirpc shared library load performance . . . . .	24
Reclaim issues . . . . .	25
Following voting, reclaimAll may be inadvertently activated . . . . .	25
With many conflicts, reclaimAll could run hot and hold commit token . . . . .	25
NetLDI failed to close socket if Gem process fork fails . . . . .	25
SEGV or Access Violation in user action . . . . .	25

Incorrect compiler optimization . . . . .	25
RC objects not handled correctly in nested transactions . . . . .	25
continuousRestoreFromArchiveLogs could fail on initial setup . . . . .	25
Issues related to Configuration Parameters . . . . .	26
SHR_PAGE_CACHE_PERMISSIONS printed without leading 0. . . . .	26
Errors in STN_GEM_INITIAL_TRANSACTION_MODE caused log bloat . . . . .	26
STN_FREE_SPACE_THRESHOLD value greater than 4GB was truncated . . . . .	26
STN_ALLOW_NFS_EXTENTS not working . . . . .	26
Restore Issues . . . . .	26
During restore, possible CR backlog and C heap growth . . . . .	26
Invalid version number in extents, confusing later upgrade errors. . . . .	26
startCheckpointSync did not always checkpoint the current state . . . . .	26
backup start time was not reported in stone log . . . . .	27
Issues related to remote caches . . . . .	27
Failure to recover after loss of pageserver for a remote cache . . . . .	27
Remote process support logs may not be written to intended location. . . . .	27
Stuck hash table lock in remote cache . . . . .	27
Page server on Stone's node may not have exited after remote cache disconnect . . . . .	27
Extent names not updated on remote cache . . . . .	27
GCI client on Windows 7 hang on socket disconnect . . . . .	27
Find reference path search missed references via class instance variables. . . . .	28
GsBitmap issues . . . . .	28
GsBitmap did not correctly handle some cases with large objects . . . . .	28
Printing a very large GsBitmap . . . . .	28
Issues specific to X509-Secured GemStone . . . . .	28
GEM_HALT_ON_ERROR and other configuration parameters not usable in X509 logins. . . . .	28
SHR_PUSH_TO_MIDCACHES_THREADS did not push pages . . . . .	28
DecimalFloat >> asStringUsingFormat: did not 0-pad per argument . . . . .	28
System >> hostCpuUsage results. . . . .	28
GsFile issues. . . . .	29
GsFile stdin > atEnd incorrectly answered true. . . . .	29
position: for GsFile in append mode . . . . .	29
Unable to write to stdout on Windows GCI client . . . . .	29
copydbf -Z broken . . . . .	29
Missing collection protocol for CanonStringDict and CanonSymbolDict . . . . .	29
Incorrect report of fatal error in configuration files. . . . .	29
Diagnosing GsExternalSession login errors . . . . .	29
gemnetobject_slow failed to load slow; ran fast libraries . . . . .	29
become: crashed with extremely large NSC. . . . .	30
Extra step point at beginning of method. . . . .	30
Programmatic GemNativeCodeEnabled value for true . . . . .	30
GsCurrentSession >> inTransaction returned integer rather than boolean . . . . .	30
References to obsolete class JISCharacter in source code . . . . .	30
RcKeyValueDictionary hard coded references to RcCollisionBucket. . . . .	30



# GemStone/S 64 Bit

## 3.5.1 Release Notes

---

### Overview

GemStone/S 64 Bit™ 3.5.1 is a new version of the GemStone/S 64 Bit object server. This release includes a number of new features and bug fixes. We recommend everyone using or planning to use GemStone/S 64 Bit upgrade to this new version.

These release notes describe changes between the previous version of GemStone/S 64 Bit, version 3.5, and version 3.5.1. If you are upgrading from a version prior to 3.5, review the release notes for each intermediate release to see the full set of changes.

For details about installing GemStone/S 64 Bit 3.5.1 or upgrading from earlier versions of GemStone/S 64 Bit, see the *GemStone/S 64 Bit Installation Guide* for v3.5.1 for your platform.

### Supported Platforms

#### Platforms for Version 3.5.1

GemStone/S 64 Bit version 3.5.1 is supported on the following platforms:

- ▶ Red Hat Enterprise Linux Server 6.9 and 7.4, and Ubuntu 16.04 and 18.04, and SUSE Linux Enterprise 12, on x86. Testing has been done on some earlier Red Hat Enterprise Linux Server 6.x and 7.x versions with the latest security patches.
- ▶ Solaris 10 and 11.4 on x86
- ▶ AIX 6.1, 7.1, and 7.2
- ▶ OS X 10.14.3 (Mojave), with Darwin 18.2.0 kernel, on x86 (Mac is supported for development only)

For more information and detailed requirements for each supported platforms, please refer to the GemStone/S 64 Bit v3.5.1 Installation Guide for that platform.

## GemBuilder for Smalltalk (GBS) Versions

GemStone/S 64 Bit version 3.5.1 requires GBS version 8.4 or later for VisualWorks Smalltalk, or version 5.4.5 or later for VA Smalltalk.

The following versions of GBS are supported with GemStone/S 64 Bit version 3.5.1:

### GBS version 8.4

VisualWorks 8.3.2 32-bit and 64-bit	VisualWorks 7.10.1 32-bit	VisualWorks 7.10.1 64-bit
<ul style="list-style-type: none"> <li>▶ Windows 10 and Windows 7</li> <li>▶ RedHat ES 7.4 and 6.9; Ubuntu 18.04 and 16.04</li> </ul>	<ul style="list-style-type: none"> <li>▶ Windows 10 and Windows 7</li> <li>▶ RedHat ES 7.4 and 6.9; Ubuntu 16.04</li> </ul>	<ul style="list-style-type: none"> <li>▶ Windows 10</li> <li>▶ RedHat ES 7.4 and 6.9</li> </ul>

### GBS version 5.4.5

VA Smalltalk 9.1	VA Smalltalk 8.6.3
<ul style="list-style-type: none"> <li>▶ Windows 10</li> <li>▶ Windows 7, Professional or above</li> </ul>	<ul style="list-style-type: none"> <li>▶ Windows 10</li> <li>▶ Windows 8.1, Professional or above</li> <li>▶ Windows 7, Professional or above</li> </ul>

Note that Windows 7 will reach end of life in January of 2020. While GemTalk will continue to support GemStone clients on Windows 7, it will no longer be considered certified.

For more details on supported 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 v3.5.1 distribution includes VSD version 5.4.2. The previous version of GemStone/S, v3.5, included VSD v5.4.

VSD version 5.4.2 is a maintenance release, includes a few bug fixes and new cache statistics definitions for v3.5.1. For details on the changes, see the [Release Notes for VSD v5.4.2](#).

VSD versions are not tied to GemStone server versions: both older and newer versions of VSD can be used to read statmonitor files generated by both older and newer versions of GemStone/S and GemStone/S 64 Bit.

## Upgrade

There have been changes in the linking process for Linux and Solaris, and in the login API for the thread safe GCI (gcits.hf).

User actions will need to be relinked, and source code using the thread-safe API will need to be updated and recompiled.

## Documentation

In addition to the updated Installation Guides, an Installation guide is now provided for the Raspberry Pi Client. Note that the Raspberry Pi is not a supported configuration.

## Changes in this release

### Updated library versions

- ▶ The version of lz4 has been updated to v1.9.2
- ▶ The version of openssl has been updated to v1.1.1d

### Distribution change in packing libraries

In 3.5, slow and noop RPC shared libraries were added to the distribution, located in subdirectories of /lib. The linked libraries were also moved and renamed to remove the `_slow` or `_noop`; however, this caused load failures (see page 29), and these libraries have been reverted to the previous names and locations.

### Distribution change for Raspberry Pi

The experimental distribution of GemStone/S 64-Bit Raspberry Pi Client has updated directory names. For consistency with other GemStone products, the directories "bin" and "sys" are now "bin32" and "sys32". Only 32-bit Raspberry Pi Clients can be used with GemStone.

### Recommendation to clear /opt/gemstone/locks/ on system restart

If there is an unexpected shutdown, the lock files (*filename*.LCK) for GemStone processes such as Stone and cache remain in the /opt/gemstone/locks/ directory (or on older systems, /usr/gemstone/locks/). These lock files include the PID and other information on the running process, that is used to determine process status.

On restart, it is possible for kernel processes to reuse this PID. If the owner is root, GemStone cannot reliably determine the status of the process and thus cannot safely delete the lock file. These lock files must be manually deleted.

It is recommended that systems be setup so that on boot, the lock files in /opt/gemstone/locks/\*.LCK are deleted automatically on system restart.

Do not delete /opt/gemstone/locks/gemstone.hostid.

### gslist added status unknown(EPERM)

For the cases in which gslist cannot determine the status of a process due to EPERM failure, gslist will now give the status as **unknown(EPERM)**.

## Support for transparent huge pages on Linux

GemStone supports explicit use of "large" pages (that is, 2 MB and/or 1GB huge pages), in addition to the default 4KB page size. These are large pages that permanently allocated by the kernel, and if configured, the GemStone configuration parameter `SHR_PAGE_CACHE_LARGE_MEMORY_PAGE_POLICY` manages their use.

In addition to these explicitly configured huge pages, the Linux kernel (in versions later than 2.6.38) supports transparent huge pages, which are 2MB in size (`/sys/kernel/mm/transparent_hugepage/hpage_pmd_size` returns 2097152). Transparent huge pages are limited to anonymous memory regions. If they are enabled (they are by default), transparent huge pages are managed by the kernel without application level configuration.

GemStone now can take advantage of transparent large pages for the shared memory region. GemStone now always `mmaps` in exact multiples of the transparent huge page size, and invokes `madvise` on the shared memory region.

Linux systems should ensure:

- ▶ `/sys/kernel/mm/transparent_hugepage/enabled` should be set to "always" or "madvise".
- ▶ `/sys/kernel/mm/transparent_hugepage/defrag` should be set to "always" or "madvise".
- ▶ `/sys/kernel/mm/transparent_hugepage/shmem_enabled` should be set to "madvise", "within\_size", or "always".

To confirm these settings are correct in your linux environment, **cat** the file; the enabled parameter is within the square brackets. E.g.,

```
unix> cat /sys/kernel/mm/transparent_hugepage/enabled
always [madvise] never
```

To confirm that GemStone is using transparent huge pages, after GemStone has been started up, execute:

```
cat /proc/shrpcmonPid/smaps
```

and look for the `/SYSV` line, and the subsequent value for `ShmemPmdMapped`.

For example:

```
unix> cat /proc/shrpcmonPid/smaps
<snip>
7fa915a0000.... /SYSV197f0019 (deleted)
Size:                124928 kB
KernelPageSize:      4 kB
MMUPageSize:         4 kB
<snip>
AnonHugePages:       0 kB
ShmemPmdMapped:    120832 kB
<snip>
```

Note that this command is expensive and should not be run unnecessarily.

## Improved Shared Page Cache startup performance

The initialization of the Shared Page Cache (SPC) during Stone startup has been multithreaded. This provides a substantial performance improvement for Stone startup when the SPC is large.

## Improvements in disk space usage after restore from full backup

### The restore of a full backup now reduces the OOP high water mark

When a programmatic backup of a GemStone database is made, it now records the actual highest value of an OOP in the repository. This is used on restore of the backup, and may mean the size and number of pages required by the object table is reduced. (#48094).

### Reduced space used by object table leaf pages after restore

During restore of backup, leaf pages may be created in the object table that do not end up being needed. The allocation of these pages has changed so the pages are only allocated if needed. (#48093)

## GsBitmap added methods

The following methods have been added to GsBitmap, to allow more powerful repository-wide analysis of objects in the database.

`GsBitmap >> allReferencedObjects`

Similar to `referencedObjects`, returning a new instance of GsBitmap containing the objects directly referenced by the objects in the receiver; but in addition, the result includes all of the normally hidden interior nodes for large objects.

`GsBitmap class >> allValidOops`

Returns an instance of GsBitmap containing the oops of all valid objects in the repository. This bitmap only reflects currently committed objects that are not in the process of being garbage collected, i.e, not in the possible dead or `deadNotReclaimed` sets.

`GsBitmap class >> transitiveReferences: aCollection`

Returns an instance of GsBitmap that contains all the objects which are transitively referenced from the objects in *aCollection*.

`GsBitmap class >> allTransitiveReferences: aCollection`

Similar to `transitiveReferences:`, however the result includes all of the normally hidden interior nodes for large objects.

`GsBitmap class >> allObjectsExcept: aCollection`

Returns an instance of GsBitmap that contains all objects that exist which are not contained in the objects transitively referenced from *aCollection*. The objects in the result may or may not be live objects (reachable from the repository roots).

## Reporting extents and files in GB or MB

The methods `fileSizeReport` and `extentSizeReport`, which report on the sizes of extent files in the repository, now report in units of GB if the repository total size is greater than 2GB.

To return results in MB as in previous releases, the following methods have been added:

```
Repository >> extentsReportMB
Repository >> fileSizeReportMB
```

## SymbolDictionary printString change

SymbolDictionaries, both those defined by GemStone base (such as Globals), and user-created SymbolDictionaries, may have names, which are keys within the SymbolDictionary that reference itself.

Previously, printing a SymbolDictionary always printed the entire contents. Now, if a name is found, only the name is displayed. If no name is found, as previously, the entire contents is displayed.

For example, the new display for Globals is:

```
topaz 1> exec Globals printString %
aSymbolDictionary( name: #'Globals' )
```

## Improved reporting on transaction conflicts

When a transaction conflict occurs, the methods `System >> conflictReportString` and `conflictReportString:` can be called. These methods now include newlines, to make the output easier to read.

The following method has been added:

```
System class >> conflictReportString: conflicts maxOops: maxOops
The conflicts argument is the Array returned by System
Class>>transactionConflicts. The report contains oops and class names
for each object in conflict, limited to the first maxOops oops in each category of
conflict.
```

## System >> remoteCachesReport now includes Stone's cache

The report returned from `System class >> remoteCachesReport` previously did not include the stone's cache; this is now included in the report.

## Added method

```
Time >> addMilliseconds: anInteger
Returns a Time that describes a time of day anInteger milliseconds later than that of the
receiver.
```

## objectAudit warnings on old 32-bit fanout in upgraded repositories

Repositories that were converted from 32-bit Gemstone/S and did not run postconv may have the older fanout, which may be less performant. This is reported as a warning by objectAudit. This warning did not respect the topaz push command, and was only visible in the console, and thus was not always found. (#48277)

The oops were written to hidden set 26, which was not easily loaded as a GsBitmap. Now, the oops are instead put into the ListInstances hidden set, which is easily accessible as a GsBitmap. (#48311)

The warning has also been reworded for clarity.

## gslist output in json format

**gslist** has added the **-j** argument, which returns information about running GemStone processes comparable to the **gslist -x** output, in json format.

For example:

```
unix> gslist -j -n gs64stone
{"GemStoneServers": [
  {
    "Name": "gs64stone",
    "Host": "benton",
    "HostId": "69621bb0476b1937",
    "Ip": "10.94.141.15",
    "Status": "Exists",
    "Type": "Stone",
    "Version": "3.5.1",
    "Creator": "gsadmin",
    "Started": "2019-09-13T16:08:00.000-07:00",
    "Pid": 29010,
    "Port": 35043,
    "Options": {},
    "LogName": "/users/gsadmin/351/data/gs64stone.log",
    "Sysconf": "/users/gsadmin/351/data/system.conf",
    "Execonf": null,
    "GEMSTONE": "/users/gsadmin/351",
    "Exe": "/users/gsadmin/351/sys/stoned"
  }
]}
```

This can be easily processed as needed for system administration. For example:

```
(JsonParser new parse: (System performOnServer: 'gslist -j'))
```

creates a structure of Dictionaries and Arrays in GemStone Smalltalk containing all information about all GemStone services.

Note that all keys and values in the output are capitalized. This includes gslist status values (such as 'Exists'), and command line arguments.

## Return values for startnetldi, stopnetldi, startstone, stopstone

The utilities that start and stop the Stone and NetLDI have further changes in the process return values in this release, for cases that could be handled as either success or failure.

In versions prior to v3.5, cases in which an attempt was made to start a service that was already running, or to stop a service that was not running, returned 0 (success).

In version 3.5, the return value of these cases was modified to return 2 (warning). For v3.5.1, all return values have been reviewed and these cases have been updated again to return 1 (informational), to allow additional non-fatal error return values. These return values are documented in the **-h** help text for **startstone**, **stopstone**, **startnetldi**, and **stopnetldi**.

The following are the

**startnetldi** returns:

- ▶ 0 on successful start.
- ▶ 1 if the specified netldi is already running.
- ▶ 2 if already running but the executables have been deleted.
- ▶ 3 if an error occurred and the specified netldi was not started.

**stopnetldi** returns:

- ▶ 0 on successful stop.
- ▶ 1 if the specified netldi was not running.
- ▶ 3 if an error occurred and the specified netldi could not be stopped.

**startstone** returns:

- ▶ 0 on successful start.
- ▶ 1 if the specified stone is already running.
- ▶ 2 if already running but the executables have been deleted.
- ▶ 3 if an error occurred and the specified stone was not started.

**stopstone** returns:

- ▶ 0 on successful stop.
- ▶ 1 if the specified stone was not running.
- ▶ 3 or above, an error occurred and the specified stone could not be stopped. Values greater than 3 indicate specific kinds of errors.

In general, GemStone processes return 1 for Informational, 2 for warning, and 3 and above for error cases.

## stopnetldi by root user on any NetLDI

If **stopnetldi** is run as the root user, it will now stop any running NetLDI, including NetLDIs that are not running as root.

## Disallowed to startstone with an extent of a later version

Executing startstone requires that the extent files be of the same version as the startstone executable, or an earlier version in the case of upgrade.

It is now explicitly disallowed to startstone with extents that are of a later version than the startstone executable. (#48209)

## copydbf -i/-I of extent file now includes version

A line has been added to the `copydbf -i/-I` output for an extent file, providing the version number of the extent.

## Applications running without \$GEMSTONE

Further support has been added to allow GemStone clients to run without the `$GEMSTONE` environment variable.

To login from linked or RPC topaz clients, you must have the `bin` directory of the GemStone installation on the machine search path, and specify the full path to the topaz executable.

Running without `$GEMSTONE` may be particularly useful for running topaz solo scripting.

For example, without `GEMSTONE` set, the following will start a topaz solo session and execute the script `mySoloScript`.

```
unix> export PATH=$PATH:/gemstoneInstallDir/bin
unix> gemstoneInstallDir/bin/topaz -l -q -C 'GEM_SOLO_EXTENT=
gemstoneInstallDir/bin/extent0.dbf;' -S mySoloScript
```

Similarly, GCI application using GemBuilder for C can run without the `$GEMSTONE` environment variable, provided all the required GemStone libraries are located in the same directory as the executable containing the `main()` function of the GCI application.

## topaz -S option

The `-S topazcommandfile` option was added in v3.5, to support scripting with topaz.

### topaz -S argument now does not suppress output

For scripting to be useful, it is necessary to suppress the various messages that topaz outputs such as echos and login responses. However, for debugging scripts, these outputs may be useful.

Now, to suppress topaz output when using `-S`, you must also use the topaz `-q` (quiet) argument.

### topaz -S argument fails to work without a .topazini

The `-S` argument allows you to also have a `.topazini` file or use the `-I` command, which execute topaz commands prior to the `-S` commands. However, if there was no `.topazini` file found, the commands in the `-S` argument were not executed. (#48302)

## topaz now handles BOM input correctly

When a text file begins with a BOM (Byte Order Mark) character, this indicates the encoding of the file. topaz now uses the UTF-8 BOM character to interpret the file as UTF-8, regardless of the current `fileformat` setting. If there is no UTF-8 BOM character at the beginning of the file, topaz continues to use the `fileformat` setting to interpret the file.

topaz does not support UTF-16 or UTF-32, so attempting to read a file with the these BOM characters is an error. In v3.5, this error would crash topaz (#48198)

## Log file locations changes for remote login processes

GemStone/S 64 Bit v3.5 included some changes in the calculation of log file locations for process logs, both Gem logs and the logs for remote processes. These changes included

- ▶ an additional function for the **startnetldi -D** flag, to provide a default directory for log files, in the absence of **#dir:** or absolute-path **#log:** directives.
- ▶ using the **-D** argument to determine directories for files forked by each NetLDI, to ensure that in a distributed configuration with different file access, each NetLDI could be individually configured with log files directory locations that are valid on that node.

These changes introduced some incorrect behavior (see “Remote process support logs may not be written to intended location” on page 27). These issues have been fixed, and the code and log file locations have been reviewed and tested.

### Verify your remote distribution log file locations

Note that there are many ways to configure log file names and locations for distributed configurations, and existing applications may be expecting older, technically incorrect behaviors for some combinations. When upgrading applications to 3.5.1 that involve remote configurations, we recommend that you review the log file locations and names to ensure that the behavior is correct for your application.

### startnetldi -X argument added

The **-X** argument has been added to **startnetldi**, allowing you to pass in an NRS (containing one or more of **#log:**, **#dir:**, and **#netldi:** settings), which take precedence over an **GEMSTONE\_NRS\_ALL** setting in the environment of the **startnetldi**.

## Attaching to a running Gem for debugging

It is now possible to set up a Gem to execute code that will be debugged, and attach to this running gem from topaz or another process.

To do this, the Gem must be configured to start a separate thread in the VM that listens for the command to connect. This can be done in two ways.

- ▶ Before login, set the configuration parameter **GEM\_LISTEN\_FOR\_DEBUG = true;** With this defined, during login the Gem will print a line to the stdout (the linked console or gem log file):

```
Listening for debug: DEBUGGEM pid random32BitDebugToken
```

- ▶ By invoking the new method `System class>>listenForDebugConnection`. This method returns the `random32BitDebugToken.token` (a random 32-bit integer), that is an argument to the call to attach. To get the PID of the gem, which is also required for login, invoke `System class >> gemProcessId`.

Then, a topaz or GCI application (the debugging session) makes the calls to connect, specifying both the PID of the executing Gem session, and the token.

```
topaz 1> DEBUGGEM pid random32BitDebugToken
successful attach
topaz 2>
```

If the connection is successful, the debugging session will have debugging control over the executing session.

- ▶ If a Smalltalk call is in progress, the equivalent of a soft break occurs in the executing Gem, and the debugging gem can examine the stack. Operations that modify the stack, such as **step** are disallowed, and the debugging session cannot commit.
- ▶ If execution is not in progress, the debugging gem takes control and further operations in the executing session are blocked.
- ▶ If a fetch or store traversal is in progress, the debugging session will get control after this completed.

Either a **logout** or **continue** in the debugging session will close that session and allow the executing session to continue.

## Configuration parameter added to support debug gem

### GEM\_LISTEN\_FOR\_DEBUG

If true, the gem or topaz -l process will open a listening socket, listening on localhost only for a topaz -r process to connect using the topaz **DEBUGGEM** command. The listening socket will show up in gslisp with the name *gempiddebug*.  
Default: false

## Method added to support debug gem

System class >> listenForDebugConnection

Returns a debug token (a random 32-bit integer), and also prints the token to the Gem log or stdout of a linked topaz session.

If this session was configured with **GEM\_LISTEN\_FOR\_DEBUG=false**, starts the required thread to listen for a debugging connection.

If unable to setup the thread or other problem occurs, return an error.  
Requires CodeModification privilege.

## Topaz commands added to support debug gem

**DEBUGGEM** *pid token*

Attach to the process with the given process Id, which must be a Gem or linked topaz on localhost, that was either configured with **GEM\_LISTEN\_FOR\_DEBUG=true**, or executed System class >> listenForDebugConnection. A token is also required, which is a random integer printed by the gem process to the log file or stdout of linked topaz.

If successful, **DEBUGGEM** creates a topaz session in the executing topaz process (the debugging session) that has debugging control over the executing Gem (the executing session).

If a Smalltalk call is in progress, the equivalent of a soft break occurs in the executing Gem, and the debugging gem can examine the stack. Commit, and operations that modify the stack, such as **STEP** are disallowed.

If execution is not in progress, the debugging gem takes control and further operations in the executing session are blocked. If a fetch or store traversal is in progress, the debugging session will get control after this completed.

LOGOUT or CONTINUE closes the debugging session and allows the executing session to continue.

STACK SET *aGsProcess*

The argument an object specification that resolves to a GsProcess. Make that GsProcess the currently active stack for debugging.

## GCI calls added to support debug gem

GciDebugConnectToGem

Connect to a gem's listening debug port on localhost. This call must be followed by a call to GciDebugStartGemThread.

```
(BoolType) GciDebugConnectToGem(
    int gemPid
);
```

GciDebugStartDebugService

Following a call to GciDebugConnectToGem, GciDebugStartGemThread will validate the *debugToken* argument and interrupt any Smalltalk execution in progress in the target gem by the equivalent of GciSoftBreak. To continue an execution that was interrupted by this SoftBreak and release the target gem to continue running, use GciNbContinue followed by GciDebugStartGemThreadGciLogout\_(FALSE).

```
(BoolType) GciDebugStartDebugService(
    uint64 debugToken
);
```

## Other topaz changes

The following sub/commands have been added:

**fr\_1**

Similar to **frame**, but displays stack temps one per line in the frame.

**stack set** *aGsProcess*

Takes one argument an object specification that resolves to a GsProcess, and make that GsProcess the currently active stack for debugging.

## Other Changes in Configuration Parameters

### CONFIG\_WARNINGS\_FATAL now applied from Gem NRS

Normally, minor errors in configuration files are treated as warnings, and do not stop processes from starting up. Setting the configuration parameter CONFIG\_WARNINGS\_FATAL to true treats these warnings as fatal errors, and the process does not start. This reduces the risk of using incorrect or unexpected configuration values.

Now, CONFIG\_WARNINGS\_FATAL can be passed in as a -C argument within the gem NRS. For example, in topaz:

```
set gemnetid 'gemnetobject -e nosuch.file.conf -C
CONFIG_WARNINGS_FATAL=TRUE'
```

## Setting Gem environment variables via configuration parameters

GemStone has a number of environment variables that are designed to support debugging operations in a gem session. These can now be set via configuration files.

### GEM\_ENV

Specifies one or more String values to be added to the gem process environment during GciInit. Values must be of the form name=value and are passed to putenv() during GciInit. Multiple settings are comma-separated; values that include spaces must be double-quoted.

For example, to specify that gem logs should not be deleted even on a clean exit, you may modify the configuration file used by the Gem to include:

```
GEM_ENV=GEMSTONE_KEEP_LOG=1;
```

which is the equivalent of passing this directly to the gem during login:

```
set gemnetid 'gemnetobject -C GEM_ENV=GEMSTONE_KEEP_LOG=1'
```

## GEM\_ABORT\_MAX\_CRIS changed

The default has changed from 0 (unlimited) to 100000. Sessions remote from the stone will use a value that is 1/3 of the setting.

Also, system gems (such as GcGems), will configure themselves with a default in the range 10 to 3000.

## RTLD\_DEEPBIND no longer used; User Action relink required

The RTLD\_DEEPBIND option for dlopen() is no longer used by GemStone when loading shared libraries. The RTLD\_DEEPBIND option caused unintended side effects when used with the LD\_PRELOAD environment variable. See the Linux man pages for dlopen and for ld.so, for further information.

This change affects GemStone running on Linux only. On Linux, all user actions that were compiled/linked with v3.5 must be recompiled/relinked for 3.5.1.

When linking a GCI application, you should now explicitly specify the -Bdynamic flag; previously this could be omitted. The following is updated:

### Linking a GCI application on Linux

```
g++ userCode.o $GEMSTONE/lib/gcirtlobj.o -Wl,-traditional  
-Wl,-z,lazy -m64 -Wl,-Bdynamic,--no-as-needed -lpthread  
-Wl,--as-needed -lcrypt -ldl -lc -lm -lrt -o userAppl
```

## Change in linker flags for Solaris

When building user actions on Solaris, the -Bsymbolic linker flag should no longer be needed or used. The following is updated:

### Linking a user action library on Solaris

```
CC -m64 -xarch=generic -G -h libuserAct.so -i userCode.o  
$GEMSTONE/lib/gciualib.o -o libuserAct.so -Bdynamic -lc  
-lsendfile -lpthread -ldl -lrt -lsocket -lnsl -lm -lpam  
-lCrun -z nodefs
```

## Other Added GCI functions

The following other GCI functions have been added:

`GciLogout_`  
 Similar to `GciLogout`, but accepts a Boolean argument. A true argument is the same as `GciLogout`, waits for a reply packet from the gem. If the argument is `FALSE`, the library sends the logout command and closes the connection without waiting for a reply from the gem.

```
(void) GciLogout_(
    BoolType getResult
);
```

`GciShutdown_`  
 Similar to `GciShutdown`, but accepts a Boolean argument.

```
(void) GciShutdown_(
    BoolType doLogouts
);
```

## Changes in Thread-safe GCI

There have been a number of changes in thread-safe GCI calls that may affect existing applications. All thread-safe GCI applications will require attention, minimally to update the `GciTsLogin/GciTsForkLogin` calls.

### **GciTsLogin and GciTsForkLogin changed**

These functions now include an additional argument,

```
BoolType *executedSessionInit
```

Any GCI applications that use the thread-safe API (`gcits.hf`) will need to be edited and recompiled.

### **GciTsClassRemoveAllMethods changed**

This method was not working in earlier versions. The fix involves including an additional argument,

```
ushort environmentId;
```

### **GciTsCompileMethod changed**

Now returns `OOP_NIL` on success, and the oop of a `String` if there are warnings in `*err`.

### **GciTsFetchTraversal changed**

The following arguments were removed:

```
GciTravBufType *travBuff,
int flag,
```

And this now returns 0 if completed traversal and 1 if there is more traversal remaining. These changes make the call more consistent with `GciTsStoreTravDoTravRefs`.

## Added functions from standard GCI API

The following added functions are similar to the gci.hf counterparts:

```
(BoolType) GciTsDirtyObjsInit(  
    GciSession sess,  
    GciErrSType *err );
```

```
(OopType) GciTsI32ToOop(  
    int arg );
```

## Added functions to support X509 Logins

```
(BoolType) GciTsX509ForkedLogin(  
    GciX509LoginArg *args,  
    BoolType *executedSessionInit,  
    GciErrSType *err,  
    GciTsCallbackFType *callback );
```

```
(GciSession) GciTsX509Login(  
    GciX509LoginArg *args,  
    BoolType *executedSessionInit,  
    GciErrSType *err );
```

## Changes in Cache Statistics

The following statistics have been removed:

- LogHighQueueAdds**
- LogHighQueueSize**
- LogLowQueueAdds**
- LogLowQueueSize**

The following statistics have been added:

**LogAioQueueAdds** (Stone)

The total number of times a session was added to the tranlog aio write queue.

**LogAioQueueSize** (Stone)

The current number of sessions in the tranlog aio write queue.

**LogBufQueueAdds** (Stone)

The total number of times a session was added to the tranlog buffer queue.

**LogBufQueueSize** (Stone)

The current number of sessions in the tranlog buffer queue.

## Bug Fixes

### GciStoreTravDoTravRefs issues with serious GBS impact

#### GciStoreTravDoTravRefs\_ may error, impacting linked GBS sessions

GemBuilder for Smalltalk linked sessions were unreliable with GemStone/S 64 Bit v3.5. GBS linked session rely on `GciStoreTravDoTravRefs_`, a specialized GCI call that performs several functions in one call. This call failed with an object does not exist error if one of the arguments became too large. (#48148)

#### Missing reports in traversals from GBS

v3.5 included a change in the GCI function `GciStoreTravDoTravRefs`, which is used by GemBuilder for Smalltalk (GBS). For classes that are identity clamped and modified, the return from this function should not have a report, but in 3.4.x, two reports were returned. The fix in v3.5 was too general and introduced cases in which required reports, for non-behavior objects, were not returned at all. (#48358)

### Terminated GsProcess may not have allowed ensure block to complete

When sending `terminate` to a `GsProcess` that is executing an `ensure` block, there is a risk that the timeout may expire before the `ensure` block completes.

The timeout in `GsProcess >> terminate` has been raised from 10 to 30 seconds to reduce the risk of this problem. (#48271)

### STN\_COMMITS\_ASYNC risk of lost record

With `STN_COMMITS_ASYNC` set to true, if the tranlogs become full and the Stone had to wait for more space to become available, the tranlog record that was waiting to be written may not actually have been written, depending on the particular point in which the tranlog directories full conditions are encountered. This caused an error on restore. (#48098)

`STN_COMMITS_ASYNC` was disabled in v3.5 and 3.4.4.

### GciNewUtf8String does not put result object in export set

The GCI function `GciNewUtf8String()` does not put the newly created object in the export set, which may allow it to be garbage collected while still in use. (#48217)

### startstone with precheckpoint and checkpoint records in different tranlogs

It is possible for the precheckpoint and checkpoint records to be written to two different tranlogs. In this case, `startstone` may fail in spite of a clean shutdown. (#48169)

### libgcirpc shared library load performance

Due to changes in the way `openssl` libraries were located, there were cases, such as when loading libraries over NFS on heavily loaded system, in which the time required to load the libraries was much slower. (#48171)

## Reclaim issues

### Following voting, reclaimAll may be inadvertently activated

After the voting step, the Stone signals the ReclaimGem. The incorrect signal was sent, which caused the ReclaimGem to go into reclaimAll mode. In reclaimAll mode, the reclaim configuration values are set to optimize to reclaim as quickly as possible; these were not reverted when finishing the reclaimAll operations. (#48426)

### With many conflicts, reclaimAll could run hot and hold commit token

Operations such as clustering that affect many objects that also need to be moved during reclaim may cause the Reclaim Gem to see many conflicts. The Reclaim Gem does not handle these conflicts efficiently, and ran hot while holding the commit token. (#48421)

## NetLDI failed to close socket if Gem process fork fails

During an RPC login, the NetLDI forks the Gem Process, creating sockets to manage further communications. The NetLDI was failing to close these sockets if the fork failed for some reason, such as unable to create the gem log file. These open sockets could eventually cause NetLDI to fail due to too many open files in the NetLDI process. (#48370)

## SEGV or Access Violation in user action

If the OOP of the name of a user action is larger than 549755813633, and this OOP has bit 0x8000000000 set, errors in the code produce a negative hash value, which causes a SEGV on Unix and an Access violation error on Windows. (#48126)

## Incorrect compiler optimization

There was an error in the compiler's optimization of certain expressions, for example `(object ifNil:[ nil ]) == object`, such that this could return incorrect results. (#48224)

## RC objects not handled correctly in nested transactions

Reduced Conflict (RC) objects (such as RcKeyValueDictionary, etc.) maintain changes in a redo log that allows a commit conflict to be resolved by replay. However, when an RC object was modified within a nested transaction, and that nested transaction was aborted, the redo log was not updated. If later, the RC object has further changes, and the commit again has conflicts such that the redo log is replayed, the changes that had been made earlier and aborted may be applied. (#48389)

## continuousRestoreFromArchiveLogs could fail on initial setup

When setting up a new a hot standby system, the final steps are to run **startlogreceiver**, and then to execute `continuousRestoreFromArchiveLogs`: on the slave system. However, it takes a few seconds after executing **startlogreceiver** before the tranlogs are available on the slave system. If `continuousRestoreFromArchiveLogs`: is executed immediately after the **startlogreceiver** is started up, before any logs were available, it failed with a file not found error. (#48346)

## Issues related to Configuration Parameters

### **SHR\_PAGE\_CACHE\_PERMISSIONS printed without leading 0**

The value for SHR\_PAGE\_CACHE\_PERMISSIONS was printed in log files without the leading '0', which caused it to be interpreted as decimal. Now, the logs show the usable value that includes the leading 0. (#48157)

### **Errors in STN\_GEM\_INITIAL\_TRANSACTION\_MODE caused log bloat**

The error reported when there was a parsing error in the value for the configuration parameter STN\_GEM\_INITIAL\_TRANSACTION\_MODE was not handled correctly and caused startup to fail and excess printing in the stone log. (#48130)

### **STN\_FREE\_SPACE\_THRESHOLD value greater than 4GB was truncated**

If the setting for STN\_FREE\_SPACE\_THRESHOLD is specified to be greater than  $2^{32}-1$  (4294967295), the value that is actually set is the value truncated to the low-order 32 bits of the number, resulting in a value no greater than 65K (#48211).

### **STN\_ALLOW\_NFS\_EXTENTS not working**

Tranlogs on NFS-mounted drives were disallowed even with STN\_ALLOW\_NFS\_EXTENTS set to true. (#48433)

## Restore Issues

### **During restore, possible CR backlog and C heap growth**

When restoring transaction logs, such as in a warm standby, a commit record may not be released in a timely manner. This can result in a commit record backlog and the reader thread causing C heap growth in the stone process. (#48214)

The statmonitor stat BmCHeapPages, that measures C heap usage, wrapped and was unreliable (#48216)

### **Invalid version number in extents, confusing later upgrade errors**

The GemStone extents store internal version numbers, to verify compatibility between versions. While the numbers for the checks were correct, the version number stored for reporting errors was '820.0', which made the error messages unclear. This has been corrected, so upgrading from 3.5.1 or later 3.5.x versions to future versions will be able to report useful error messages. (#47952)

## **startCheckpointSync did not always checkpoint the current state**

Previously, if startCheckpointSync was started while a checkpoint was going on, it would return at the end of that checkpoint. This left a potential gap between the start of that checkpoint (which could be some time ago), and the moment at which startCheckpointSync was invoked. Now, if a checkpoint is underway when this method is invoked, a second checkpoint is started immediately after that one completes. This ensures that all changes, as of the time that startCheckpointSync is invoked, are written to the extents. (#48360)

## backup start time was not reported in stone log

The stone log normally contains the time a full programmatic backup was started and the time it completed. The start message was not being included. (#48100)

## Issues related to remote caches

### Failure to recover after loss of pageserver for a remote cache

When the pageserver on the Stone's node that services a remote cache dies, the remote cache should shutdown and a subsequent login on that remote node should trigger the restart of the remote cache. However, login attempts during the period could prevent the remote cache from shutting down to complete the recovery. (#48166)

### Remote process support logs may not be written to intended location

The location in which Gem process log files are written is controlled by a number of settings, including #dir and #log directives in the GEMSTONE\_NRS\_ALL for both NetLDI and Gem and the NRS for a Gem, and -D setting for startnetldi. There were several changes made in v3.5 in how the precedence of these settings calculated the directory that would be used.

Gem logs were located correctly, but the location calculation for the logs for the shared page cache monitor and cache page server for a remote Gem was not done correctly, and for some configurations, these logs could be written to default location (the home directory). (#48369)

### Stuck hash table lock in remote cache

It was possible for an attempt to remove an old lifetime of a page in a remote cache to result in a stuck hash table lock. (#48453)

### Page server on Stone's node may not have exited after remote cache disconnect

When a remote cache is no longer connected, the page server on the Stone's node that supports that remote host should shutdown. This was not occurring; this left excess page server processes on the Stone's node. (#48459)

### Extent names not updated on remote cache

When a new extent was created at runtime, the cached list of extent names was not updated in a remote cache unless a new gem logged into the remote cache. (#48183)

## GCI client on Windows 7 hang on socket disconnect

A GCI client (including RPC GBS and topaz) on Windows 7 may hang when a socket is killed and the Gem exits. This is related to that OS version's socket library returning EWOULDBLOCK, which otherwise implies data on the socket. The GCI client code now recognizes this condition and will terminate the connection. (#48448)

## Find reference path search missed references via class instance variables

If the only reference to an object was via a class instance variable, and the class itself was not in any user's SymbolDictionaries, the object could be incorrectly reported as unreachable/dead. (#48447)

## GsBitmap issues

### GsBitmap did not correctly handle some cases with large objects

Some GsBitmap methods to find references did not correctly handle cases of references through NSCs or LargeObjectNode, and errored. Specifically, finding references paths using GsSingleRefPathFinder could error (#48445)

### Printing a very large GsBitmap

GsBitmap printOn: did not limit the amount printed. For very large GsBitmaps, this failed. (#48443)

## Issues specific to X509-Secured GemStone

Further testing has been done on X509-Secured GemStone and a number of issues have been fixed.

### GEM\_HALT\_ON\_ERROR and other configuration parameters not usable in X509 logins

In a Gem created by an X509 login, the internal setting for GEM\_HALT\_ON\_ERROR did not use the configured value, and the Gem did not stop on error. This problem applied to other configuration parameters as well. (#48388)

### SHR\_PUSH\_TO\_MIDCACHES\_THREADS did not push pages

When this X509 GemStone-only configuration parameter was enabled, it pushed cache warmer pages, but did not push newly committed pages. (#48354)

## DecimalFloat >> asStringUsingFormat: did not 0-pad per argument

The arguments to asStringUsingFormat: include the number of digits to include following the decimal point. While DecimalFloat rounded if there were more digits, it did not pad with 0 as documented to the minimum number of digits. (#48305)

## System >> hostCpuUsage results

It was possible for System >> hostCpuUsage to return an array of zeros. (#48220).

Note that invalid results, indicating 100% idle, are still returned in some cases (such as on Linux for the first call and for two calls too close together).

The code supporting this method also did not round predictably, so the percentages of CPU did not always add up to 100. (#48232)

## GsFile issues

### GsFile stdin > atEnd incorrectly answered true

The stdin GsFile (accessed via `GsFile class >> stdin`), incorrectly answered true on the first message, and at other times during operation. (#48259)

### position: for GsFile in append mode

When a GsFile is opened in append mode (using `openAppend:*`, or the 'a' file mode), reading to a file is not allowed. In any append mode, the file pointer is always set to the end of the file before a write operation. In previous releases the `position:` method appeared to work, but effectively was a no-op. Now, attempting to invoke `position:`, or methods such as `skip:`, `rewind`, etc. which invoke `position:`, will error. (#48265)

### Unable to write to stdout on Windows GCI client

While it was possible to get a result from GsFile stdout (and stderr, etc.), from a Windows client application (such as GBS), it was not possible to perform write or flush operations to write data to the application's console. (#47998)

## copydbf -Z broken

The -Z option to the copydbf utilities reported an invalid compression argument error. (#48188)

## Missing collection protocol for CanonStringDict and CanonSymbolDict

These specialized collections did not implement the collection methods `do:`, `select:`, `reject:`, and `collect:`. (#48286)

## Incorrect report of fatal error in configuration files

There were code paths in Gem startup in which an unrelated error could incorrectly trigger the warning message "!!! FATAL ERRORS IN GemStone CONFIGURATION FILES !!!". (#48395)

## Diagnosing GsExternalSession login errors

When using GsExternalSession to login, and login failed, determining the cause by examining the specific login parameters was difficult. Now, a login error will also report details of the GsExternalSession parameters. (#48083)

## gemnetobject\_slow failed to load slow; ran fast libraries

GemStone provides "slow" libraries, which have less optimization and additional assertion checks, for debugging. Due to distribution changes in v3.5, using `gemnetobject_slow` could fail in loading the slow shared library, in which case it fell back to using the regular fast library. (#48219). See page 12 for the additional distribution changes in v3.5.1.

## become: crashed with extremely large NSC

Sending become: to a million-item UnorderedCollection (Non-sequential collection), such as IdentityBag, could crash the Gem with a SEGV. (#48294)

## Extra step point at beginning of method

The compiler produced an extra step point at the beginning of methods, although there was not actually a step point there. (#48118)

## Programmatic GemNativeCodeEnabled value for true

The configuration parameter GemNativeCodeEnabled can accept 0, 1, or 2. For backwards compatibility, it can also be set to **false** (0) or **true** (2). However, access by means of System configurationAt: #GemNativeCodeEnabled [ put: ] translated **true** to 1; in v3.5.1, **true** translates to 2. (#47661)

## GsCurrentSession >> inTransaction returned integer rather than boolean

This method was documented to return a boolean; however, it returned 1 or 0. (#48290)

## References to obsolete class JISCharacter in source code

The source code for an image method contained a reference to the class JISCharacter, which was moved to ObsoleteClasses in 3.5. This method could not be recompiled. The reference has been removed. (#48343)

## RcKeyValueDictionary hard coded references to RcCollisionBucket

It is allowed to create customized collision buckets for a subclass of RcKeyValueDictionary; however, there were hardcoded references directly to RcCollisionBucket. These have been changed to invoke \_bucketClass. (#48390)