GemStone®

GemStone/S Release Notes

Version 6.6.3.2

February 2013

vmware®

GemStone S™

INTELLECTUAL PROPERTY OWNERSHIP

COPYRIGHTS

PATENTS

GemStone software is 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”. GemStone software may also be covered by one or more pending United States patent applications.

TRADEMARKS

VMware, Inc.
15220 NW Greenbrier Parkway
Suite 150
Beaverton, OR 97006

Preface

About This Documentation

These release notes describe changes in the GemStone/S version 6.6.3.2 release. We recommend that everyone migrating to this version read these release notes before beginning installation, testing or development.

No separate Installation Guide is provided with this release. For instructions on installing GemStone/S version 6.6.3.2, or upgrading or converting from previous products or versions, see the Installation Guide for version 6.6.3.

These documents are also available on the GemStone customer website, as described below.

Technical Support

GemStone's Technical Support website provides a variety of resources to help you use GemStone products.

Documentation for released versions of all GemStone products is provided in PDF form on this website:

GemStone Web Site: http://support.gemstone.com

In addition to documentation, the GemStone support website provides:

- Downloads and Patches, including past and current versions of GemBuilder for Smalltalk.
- Bugnotes, identifying performance issues or error conditions that you may encounter when using a GemStone product.
- TechTips, providing information and instructions that are not in the documentation.
- Compatibility matrices, listing supported platforms for GemStone product versions.

This material is updated regularly; we recommend checking this site on a regular basis.

Help Requests

You may need to contact Technical Support directly, if your questions are not answered in the documentation or by other material on the Technical Support site.

Requests for technical assistance may be submitted online, by email, or by telephone. We recommend you use telephone contact only for more serious requests that require immediate evaluation, such as a production system down. The support website is the preferred way to contact Technical Support.

Website: http://techsupport.gemstone.com

Email: techsupport@gemstone.com

Telephone: (800) 243-4772 or (503) 533-3503

Your GemStone support agreement may identify specific designated contacts who are responsible for submitting all support requests to GemStone. If so, please submit your information through those individuals.

If you are reporting an emergency by telephone, select the option to transfer your call to the Technical Support administrator, who will take down your customer information and immediately contact an engineer. Non-emergency requests received by telephone will be placed in the normal support queue for evaluation and response.

When submitting a request, please include the following information:

- Your name, company name, and GemStone server license number.
- The versions of all related GemStone products, and of any other related products, such as client Smalltalk products.
- The operating system and version you are using.
- A description of the problem or request.
- Exact error message/s received, if any, including log files if appropriate.

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

24x7 Emergency Technical Support

GemStone 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 your GemStone account manager.

Contents

Chapter 1

GemStone/S 6.6.3.2 Release Notes

Overview

GemStone/S 6.6.3.2 is a new version of the GemStone 32-bit Smalltalk object server, addressing several issues in markForCollection as well as addressing other issues. We recommend everyone using GemStone/S upgrade to this new version.

These release notes provide changes between the previous general access version of GemStone/S, version 6.6.3, and version 6.6.3.2, and includes all changes described in the limited distribution 6.6.3.1 release. If you are upgrading from a version prior to 6.6.3, please also review the release notes for each intermediate release to see the full set of changes.

No separate Installation Guide is provided with this release. For installation instructions, use the Installation Guide for version 6.6.3.

Supported Platforms

GemStone/S version 6.6.3.2 is supported on the following platforms:

- Solaris 8, 9 and 10 on SPARC
- AIX 5.3, 6.1, and 7.1
- Red Hat Linux ES 5.0, 5.5, and 6.1
- Windows XP, Windows 2003, Windows 7, and Windows 2008 R2

For more information and detailed requirements for each supported platforms, please refer to the *GemStone/S Installation Guide* v6.6.3 for that platform.

GBS version summary

The following versions of GBS are supported with GemStone/S version 6.6.3.2, with the following client Smalltalk and platforms versions.

GBS version 7.5

VW 7.9.1	VW 7.9
▸ Windows XP, Windows 7, and Windows 2008 R2 ▸ Solaris 9 and 10 on SPARC ▸ RedHat Linux ES 5.0, 5.5, and 6.1	▸ Windows XP, Windows 7, and Windows 2008 R2 ▸ Solaris 9 and 10 on SPARC ▸ RedHat Linux ES 5.0, 5.5, and 6.1

GBS version 7.4.1

VisualWorks 7.8.1	VisualWorks 7.8	VisualWorks 7.7.1
▸ Windows XP, Windows 7, and Windows 2008 R2 ▸ Solaris 9 and 10 on SPARC ▸ RedHat Linux ES 5.0, 5.5, and 6.1	▸ Windows XP, Windows 7, and Windows 2008 R2 ▸ Solaris 9 and 10 on SPARC ▸ RedHat Linux ES 5.0, 5.5, and 6.1	▸ Windows XP and Windows 7 ▸ Solaris 9 and 10 on SPARC ▸ RedHat Linux ES 5.0 and 5.5

GBS version 5.2.7

VA Smalltalk 8.0.2	VA Smalltalk 7.5.2
▸ Windows XP, Windows 7, and Windows 2008 R2	▸ Windows XP

For more information and details, see the *GemBuilder for Smalltalk Installation Guide* for that version. This information, and instructions for upgrading client libraries for GBS, are in the *GemStone/S Installation Guide* v6.6.3, chapter 3.

Changes in this release

Object >> _objectWithOop: wrong return for object that does not exist

When the method Object >> _objectWithOop: is passed an OOP that is not in use, it returned a primitive error. Per method comments, this should return nil. (#42572)

Empty bitmap leaf pages clutter up caches

The caches could become filled with empty bitmap leaf pages, which reduced performance. This release also includes other performance improvements in bitmap operations. (#42664)

markForCollection handling out of memory when configured for large buffer

Previously, if there was not enough memory to allocate the buffers computed for the markForCollection (or findDisconnectedObjects) operation, based on the specificed #mfcGcPageBufSize, the MFC session would see a malloc() error and crash. (#42706)

Now, if the MFC session sees a failure when allocating memory, it will retry with smaller values until it is able to allocate the memory. The MFC output will specify the actual buffer size used, not the value specified by #mfcGcPageBufSize.

The improved memory handling also applies to buffers allocated for multi-threaded MFC and FDC; these do not use #mfcGcPageBufSize, but have variants that accept a buffer size argument.

GS_DEBUG_MFC_MEM

To get a complete printout of memory used by the MFC (or FDC), set the environment variable GS_DEBUG_MFC_MEM to any value while running linked topaz. When this is set, the output for markForCollection will include the sizes for buffer and associated other memory demands, such as hash map sizes, for the buffer size specified by #mfcGcPageBufSize and the actual buffer size used.

Extremely large #mfcGcPageBufSize cause very serious corruption

If a #mfcGcPageBufSize is specified that is sufficiently large, e.g. 8000000, it can result in all objects in the repository being collected. **The repository is no longer usable and must be restored from backup.** (#42708)

Now, there is a hard limit of 262143 pages; requests above this will automatically use a default 320 page buffer. The minimum size is 16 and smaller requests will also use 320.

Reduced memory required by markForCollection

The markForCollection allocates a buffer based on #mfcGcPageBufSize that includes, among other memory required, space for a hash map. This was sized considerably larger than required. This hash map size calculation has been adjusted so markForCollection should have a 30% smaller footprint.

Executable changes to allow larger SPC on AIX

The way the GemStone executable is built on AIX has been modified to allow a larger shared page cache. In this release, the hard upper limit on the SPC size is 3.25 GB (3355443200 bytes). (#42595)

One change that allows this is a modification to the address space model of the GemStone processes. Now, processes specify their address model using the equivalent of:

```
ldedit -b maxdata:0/dsa
```

With a maxdata of 0, AIX allows 13 segments to be used for shared memory. In previous releases, GemStone processes specified the very large address-space model using the equivalent of:

```
ldedit -b maxdata:0xD0000000/dsa
```

which allowed 12 segments.

The change permitting 13 segments to be used makes an additional 256M available for the shared cache, but fewer segments will be available for other memory demands, so other cache sizes will be more limited. The memory requirements for heap and stack together cannot exceed 256M for any process.

On AIX with this release, we recommend the STN_PRIVATE_PAGE_CACHE be limited to no more than 65536, regardless of SPC size. Since the stone allocates small amounts of memory as it runs, this size provides sufficient space so that unusual demands, such as a commit record backlog, do not result in the stone running out of memory and shutting down.

Gem processes will also need to limit the GEM_TEMPOBJ_CACHE_SIZE, GEM_PRIVATE_PAGE_CACHE_KB, and the size of the MFC buffers so that the total remains within the 256M limit.

In addition to this change, the Smalltalk interpreter stack is now allocated on the heap instead of using mmap(). This change allows another 256M of memory.

Added Configuration Parameters

STN_CHECKPOINT_DEFER_TIMEOUT

Time in seconds that stone will defer starting a checkpoint when the commit token is busy. Once this time has passed, stone will start the checkpoint even if the commit token is not free.

If the value of the parameter is 0, checkpoints are always started on-time and the status of the commit token is ignored.

Runtime equivalent: StnCheckpointDeferTimeout
Stone Cache Statistic: CheckpointDeferTimeout
Units: Seconds
Default: 30
Min: 0
Max: 536870911

STN_COMMIT_TOKEN_TIMEOUT

Maximum interval in seconds that a session may possess the commit token. If the session possess the token for longer than this period, the session will be logged off the system and an error message will be written to the stone log. GcGems (all types) are exempted from this time out.

Runtime equivalent: StnCommitTokenTimeout
Stone cache statistic: CommitTokenTimeout
Units: seconds
Default: 0 (stone waits forever)
Min: 0
Max: 86400

The log messages in the stone log will be of the form:

Session NNN, in checkForStuckCommitToken, was killed by timeout.

Added Cache Statistics

The following cache statistics have been added:

AllocatedOopsCount (Gem)
Number of object identifiers stone has allocated to the session.

AllocatedPagesCount (Gem)
Number of pages stone has allocated to the session.

CheckpointDeferState (Stone)
Current state of checkpoint deferral:
0 - checkpoints not deferred.
1 - start of checkpoint deferred.
2 - finish of checkpoint deferred.

CheckpointDeferTimeout (Stone)
Current setting of the STN_CHECKPOINT_DEFER_TIMEOUT stone configuration parameter.

ClientAllocatedOopsCount (Stone)
Number of object identifiers stone has allocated to the client gem of this page server.

ClientAllocatedPagesCount (Stone)
Number of pages stone has allocated to the client gem of this page server.

ClientNewOopRequests (Stone)
Number of times the client gem of this page server made a call to stone to request more object identifiers.

ClientNewPagesRequests (Stone)
Number of times the client gem of this page server made a call to stone to request more pages.

ClientReturnOopRequests (Stone)
Number of times the client gem of this page server made a call to stone to return object identifiers.

ClientReturnPagesRequests (Stone)
Number of times the client gem of this page server made a call to stone to return pages.

CommitTokenTimeout (Stone)
Current setting of the STN_COMMIT_TOKEN_TIMEOUT stone configuration parameter.

NewOopRequests (Gem)
Number of times the session made a call to stone to request more object identifiers.

NewPagesRequests (Gem)
Number of times the session made a call to stone to request more pages.

ReturnOopRequests (Gem)
Number of times the session made a call to stone to return object identifiers.

ReturnPagesRequests (Gem)
Number of times the session made a call to stone to return pages.