GemStone Bug Summary for Search

December 2021

GemTalk™ Systems

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.

COPYRIGHTS

TRADEMARKS

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

Bugnotes as of December 15 2021

For ease in searching bugnotes, this document provides all details on the bugnotes for selected versions of all GemTalk products. This information is provided for GemTalk customers to ensure that they are aware of error conditions, potentially unexpected behavior, and available workarounds.

This document is current as of the give date. For the most up-to-date information, see https://gemtalksystems.com/techsupport/bugnotes/

The following product bugnotes are included in this master document:

- GBS/VW
- GBS/VA
- GemStone/S 64 Bit
- GemStone/S (32 Bit)
- GemBuilder for Java
- GemConnect
- VSD

GBS-VW

Bug 49537 - Errors on inspecting an object with dynamic instance variables containing unreplicateable objects

Product: GemBuilder for Smalltalk/VW

Versions: 8.5, 8.4

GBS inspectors use evaluate instead of execute when displaying server objects that included dynamnic instance variables. If the value of the dynamic instance variable has unreplicateable objects, then this resulted in an error "Block replication failed"

Workaround:

Replace the methods GbxBasicInspector>>dynamicInstVarFields and GbsSession >> unsynchronizedExecuteBlock:withArguments:context: with the following:

```
GbxBasicInspector >> dynamicInstVarFields
    GbxServerVersion current atLeast64v3 ifFalse: [^Array new].
    [| dynamicInstVars |
    dynamicInstVars := OrderedCollection new.
    self object delegate session
        sessionCritical:
            [(self object delegate
                unsynchronizedPerformOnGsServer: #dynamicInstanceVariables) do:
                        [:varName |
                        | attr instVarDelegate |
                        attr := GbxDynamicInstVarAttribute
                                    label: #varName << #gbs >> (‘dynamic: ’ , varName
asString)
                                    value: (instVarDelegate := self object delegate
session
                                                    unsynchronizedExecuteBlock:
‘[:obj :var | obj dynamicInstVarAt: var]'
                                                    withArguments: (Array with: self
object delegate with: varName)
                                                    context: nil)
                                    printString: (self remotePrintStringFor:
instVarDelegate)
                                    object: self object.
                        dynamicInstVars add: attr]]
        unlessBlocked: [^OrderedCollection new]
        retryFor: 500.
    ^dynamicInstVars asArray]
            on: Core.MessageNotUnderstood
            do: [:ex | ^Core.Array with: ‘error getting dynamic inst vars’]


GbsSession >> unsynchronizedExecuteBlock: sourceString withArguments: args context:
contextDelegate
        "Private to GBS.  Does not synchronize server and client object spaces, so safe
        only in specific internal GBS operations.
        Immune from debugger stepping and asynchronous events."

        | evaluation |
        self gbsMessenger
```

```
                statusTag: #unsynchronizedEvaluateBlock
                message: '(s%1) GbsSession>>unsynchronizedExecuteBlock: %2 withArguments:
%3 context: %4'
                with: self sessionId
                with: sourceString
                with: args
                with: contextDelegate.
        evaluation := self newServerEvaluation.
        ^evaluation
                arguments: args;
                sourceCode: sourceString;
                contextDelegate: (self delegateForUnsynchronizedContext: contextDelegate);
                isBlock: true;
                shouldSynchronize: false;
                shouldReplicateResult: false;
                replicationScheme: GbxNullReplicationScheme new;
                stopForBreakpoints: false;
                execute
```

Bug 48148 - GciStoreTravDoTravRefs_ may error, impacting linked GBS sessions

Product: GemStone/S 64 Bit

Versions: 3.5

Fixed In: 3.5.1

Impact: Critical

GemBuilder for Smalltalk interacts with the server using specialized GCI calls, such as GciStoreTravDoTravRefs_, that perform several functions in one call. Due to code changes in GS64 v3.5, if the numNotReplicated buffer gets too large, GciStoreTravDoTravRefs_ fails with Error 2101, oop 0 does not exist.

As a result, linked GBS sessions are unreliable with GemStone/S 64 Bit v3.5 servers, particularly if many objects are replicated and there are many object changes.

Workaround:

Avoid using linked logins with GBS 8.4 and GS64 v3.5.

Bug 47632 - "Compile in GS" may have incorrectly reported structure differences between GemStone and VisualWorks

Product: GemBuilder for Smalltalk/VW

Versions: 8.3

Fixed In: 8.4

The private method GbsSession>>privdoesClientClass:matchServerClass: incorrectly did not include inherited instance variables when performing its calculation, which resulted in it sometimes detecting differences between the client and server classes where none actually existed.

Bug 47511 - GS operations in Workspace not working correctly with 7.10.1

Product: GemBuilder for Smalltalk/VW

Versions: 8.3, 8.2

Fixed In: 8.4

Impact: Critical

GBS adds operations GS-Do it, etc., to the pop up menus, menu bar menus, and the icon bar. Due to changes in the way VW handles these menus, these operations fail with VW 7.10.1; they work correctly with VW 8.x.

In GBS version 8.2, the pop-up menu items works correctly, but the menu bar and icon bar items fail.

Workaround:

Download and install the following patches.

Gbs7.10.1-Patch47511-Popup.st fixes the problem with the Popup menu in 8.3.

Gbs7.10.1-Patch47511-MenuBar.st fixes the problem with the Menu and icon bar, which is in 8.3 and in earlier versions.

Bug 47392 - GS-Senders of optimized selectors gets walkback

Product: GemBuilder for Smalltalk/VW

Versions: 8.3, 8.2, 8.1

Fixed In: 8.4

While optimized selectors appear in the source code for methods, the compiled bytes are optimized, so these are not message sends. This means that the optimized selector is not in the symbols associated with the server method structure, and thus not reported when searching for senders. This is not handled correctly in GBS 8.x, and a subscript out of bounds error occurs.

Bug 47389 - Browse class from inspector on class failed

Product: GemBuilder for Smalltalk/VW

Versions: 8.3

Fixed In: 8.4

When in a inspector on a GemStone server class, the menu item Browse did not correctly open a GemStone class hierarchy browser

Bug 47385 - Committed method changes may be lost if class option instancesNonPersistent added

Product: GemBuilder for Smalltalk/VW

Versions: 8.3

Fixed In: 8.4

Impact: Critical

Adding the class option #instancesNonPersistent does not create a new version of the class; this is behavior on the GemStone server.

When the #instancesNonPersistent is added to a class's definition, the methods for that class are reverted to the methods that existing the previous time that class had been versioned, regardless of method changes that had been committed meanwhile. This results in the loss of committed method changes.

This is an example of the sequence:

1. create a new version the class, and commit.

2. add and modify some methods on the class, and commit.
3. add #instancesNonPersistent to the class definition (making no other changes) and commit.

All changes made in step 2 are lost.

Workaround:

Forcing a new class version, such as by adding an instance variable, avoids problems. You may wish to add temporary instance variable, commit, then remove that instance variable in the same transaction as adding the class option.

In versions subject to this bug, file out method source before making changes to class options, to be safe.

Bug 47294 - UserTools update authentication scheme to GemStone does not work

Product: GemBuilder for Smalltalk/VW

Versions: 8.3

Platform with 3.4 server only

Fixed In: 8.4

Attempting to modify a GemStone user's authenticationScheme to GemStone, using the User Tools, encounters an error, #enableGemStoneAuthentication: is not understood. In v3.4, the method has been renamed to #enableGemStoneAuthenticationWithPassword:.

Workaround:

Update the authentication scheme programmatically.

Bug 47208 - Class > Add > Instance Var... does not copy methods to the new class version

Product: GemBuilder for Smalltalk/VW

Versions: 8.5, 8.4, 8.3, 8.2, 8.1, 7.6.1

Browser menu item Class > Add > Instance Var... allows you to add an instance variable to a class, similarly to editing the class definition. However, adding an instance variable does not copy the methods from the class into the new class version; the browser will show a class with no methods.

Bug 46781 - Primitive failure when unstubbing for mapped classes of different sizes

Product: GemBuilder for Smalltalk/VW

Versions: 8.2, 8.1, 8.0

Fixed In: 8.3

If an application uses automatic name-matching class mapping for a subclass of OrderedCollection, and theses OrderedCollection subclasses have unmatched named instance variables, unstubbing of instance with fewer unnamed instance variables can fail.

The problem is when the class on the client more named instance variables than the class with the same name on the server, and a stub is created for an instance with fewer elements than number of instance variables, before the classes are mapped. The attempt to change the class of the stub fails since the stub is too small to accommodate the named instance variables.

Workaround:

If you hit this scenario, you should log out of GemStone.

Bug 46617 - Browser override/overridden arrows do not distinguish between class and instance methods

Product: GemBuilder for Smalltalk/VW

Versions: 8.2, 8.1, 7.6.1

Fixed In: 8.3

Browsers use up and down arrows to distinguish methods that override one inherited from a superclass, or that are implemented by a subclass.

The computation of inheritance did not distinguish between class and instance methods, so the arrows incorrectly were displayed when the superclass or subclass method was not actually overriden.

Bug 46610 - While in the debugger, executing GS-Debug it gets walkback, leaves debugger in stuck state

Product: GemBuilder for Smalltalk/VW

Versions: 8.2

Platform with VW 8.x

Fixed In: 8.3

In a debugger, attempting to execute code using GS-Debug it results in a walkback, Message not understood: #gbxIsServerContext. This error leaves the debugger state such that the debugger does not respond to close.

Bug 46580 - Dragging a method to a new protocol makes hierarchy browser unusable

Product: GemBuilder for Smalltalk/VW

Versions: 8.2, 8.1, 7.6.1, earlier versions

Fixed In: 8.3

If the hierarchy browser is opened on a class, and a subclass of that class is selected and a method dragging between protocols (method categories), after the drag the browser resets itself to select the original class (the superclass of the one with the moved methods).

If this superclass has fewer protocols than the class in which the method was moved, while the method move is successful, the reset of focus results in a subscript out of bounds error. The browser is no longer usable and must be closed.

Bug 46483 - Privileges browser broken under VW 8.x

Product: GemBuilder for Smalltalk/VW

Versions: 8.2

Platform with VW 8.x

Fixed In: 8.3

Attempting to open the privileges browser with GBS 8.2 in VisualWorks 8.1.1 encounters an MNU.

Workaround:

Privileges can be viewed and edited using server methods.

Bug 46471 - Saving server method in debugger errors, resulting in lost session

Product: GemBuilder for Smalltalk/VW

Versions: 8.3, 8.2, 8.1

Platform with 3.3 or 3.3.1

Fixed In: with 3.3.2 or later server

Impact: Critical

Under some (uncommon) conditions, saving a method in the debugger can encounter an error similar to "nativeStackNotConvertableToInterpreted, a InternalError occurred (error 2261), The object with object ID 82447617 is corrupt. Reason: 'CorruptObj, fetch past end".

This is not recoverable, and the session will terminate.

This is related to server changes introduced in v3.3, and does not occur with 3.2.x.

Bug 46434 - Browser auto-commit icon does not indicate state

Product: GemBuilder for Smalltalk/VW

Versions: 8.2

Platform with VW 8.x

Fixed In: 8.3

Under VW 8.x, the Icon bar's auto-commit toggle button (the first one on the left) does not change its display when clicked on. There is no way to detect if autocommit is in effect or not.

Workaround:

The Browser's session menu has a menu item Auto-Commit that has a check mark when auto commit is on.

Bug 46433 - Debugger doesn't highlight inside of some blocks

Product: GemBuilder for Smalltalk/VW

Versions: 8.2

Platform with GS64 3.3.x

Fixed In: 8.3

When running with GS/64 v3.3.2 and later 3.3.x versions, when using the GBS Debugger and stepping into certain blocks, highlighting of the current code being executed does not happen. Stepping is still occurring, you just can't tell where you are. When you step out of the block highlighting will resume. If you step-into a method, the new method is displayed and highlighting resumes until you return back to the original method.

This only happens for certain blocks (for example, a block associated with #do:). Other blocks (for example, a block associated with #timesRepeat:) will work properly.

Workaround:

Patch file available by request from GemTalk Technical Support.

Bug 46361 - Browser Control-F dialog cannot easily be closed

Product: GemBuilder for Smalltalk/VW

Versions: 8.2

Platform with VW 8.x

Fixed In: 8.3

When you use the control-F keys to open a find class dialog, the resulting dialog does not readily close. The Cancel button, OK button, and dialog close all result in the dialog reopening.

This is not an issue when using the Find menu item Find Class option.

Workaround:

Clicking on cancel 10 times closes the dialog.

Bug 46338 - Loading the incorrect client parcel succeeds but reports failure

Product: GemBuilder for Smalltalk/VW

Versions: 8.4, 8.3, 8.2

Impact: Informational

Loading GBS v8.2, 8.3, or 8.4 into a VisualWorks image involves loading the GbsTools parcel, which automatically loads the VisualWorks version-specific additional parcel that is appropriate for the current version of VisualWorks, either 7.x or 8.x. If you accidentally request to load the incorrect additional parcel, GBS will detect this and load the correct parcels.

However, the attempt to load the incorrect additional parcel in 8.2 or 8.3 results in VW error on load failure, which reports that the load was aborted. This refers only to the load of the incorrect parcel; GBS otherwise was loaded correctly; check the Transcript for errors.

In 8.4, this reports an error and the parcel is not loaded.

Bug 46334 - Client map in 64-bit clients become corrupted by delete

Product: GemBuilder for Smalltalk/VW

Versions: 8.1, 7.6.1, 7.6, 7.5, 7.4.1, 7.3.3, earlier versions

Platform 64-bit VisualWorks

Fixed In: 8.2

Impact: Critical

Specifically in the client map code with 64-bit VisualWorks, under some rare cases the delete may not be done correctly, corrupting the client map.

The problem occurs when an entry being deleted is near the beginning of the table and falls between an entry whose hash is near the end of the table and the place where that entry has been placed due to collisions (which means that the entry falls in the collision chain of the entry that wraps around the end of the table to the beginning), the entries will fail to be swapped.

Searches for the key that should have been swapped will fail, and the invalid state could lead to further corruption as well.

This does not affect clients on 32-bit VisualWorks, which implement the clientMap somewhat differently.

Workaround:

We strongly recommend not using GBS versions earlier than 8.2 with 64-bit VisualWorks clients. Due to bug 46333, this kind of corruption cannot be detected by audit.

If you are using 64-bit VisualWorks clients, please contact GemTalk Technical Support for patch code and instructions.

Bug 46333 - Audit of client map in 64-bit clients does not detect problems

Product: GemBuilder for Smalltalk/VW

Versions: 8.1, 7.6.1, 7.6, 7.5, 7.4.1, 7.3.3, earlier versions

Platform 64-bit VisualWorks

Fixed In: 8.2

The client map has the ability to audit, using #auditHashing. This did not detect certain kinds of corruption, such as when a key's hash would place it near the end of the table, but collisions have wrapped around and put the key near the beginning of the table, and there is an illegal intervening nil.

Bug 46308 - Tools become unresponsive when debugger open on some errors

Product: GemBuilder for Smalltalk/VW

Versions: 8.5, 8.4, 8.3, 8.2

Platform VW 8.x

In the GUI framework introduced in the VisualWorks 8.x environment, editor panes may become unresponsive when a debugger is open on certain kinds of errors that arise from evaluating code in that editor, using the pop-up menu.

For example, a do it or GS-do it that gets a message non understood error will require that the debugger be closed on that error, before you can edit text in that editor.

Workaround:

Using the toolbar or edit menu options to execute code avoids this issue, and closing the debugger allows normal operation to continue.

Bug 45982 - Debugger evaluation containing an explicit return results in error or corrupts variable values

Product: GemBuilder for Smalltalk/VW

Versions: 8.5, 8.4, 8.3, 8.2, 8.1, 7.6.1, 7.6, 7.5, 7.4.1, 7.4, 7.3.3, 7.3.2, 7.3.1, 7.3, 7.2.2, 7.2.1, 7.2, 7.1.2

If a GemStone smalltalk expression is executed while in a GemStone context in the debugger, using GS-do it, GS-Print it, or GS-Inspect it, and the code includes an explicit return (^), execution results are not correct.

In most cases, this results in a subscript out of bounds error. In the case of evaluations that affect method variables in the debugger, however, this may corrupt the values of the temporary variables in the debugger.

Workaround:

Avoid executing code containing explicit returns within the debugger.

Bug 45966 - GS-File in of code that was filed out from topaz in 8-bit in ambiguous range is corrupted

Product: GemBuilder for Smalltalk/VW

Versions: 8.5, 8.4, 8.3, 8.2, 8.1, 8.0., 7.6.1, 7.6, older versions

Platform GS64 3.2.x and later

As of GemStone/S 64 Bit 3.2.x and 3.3.x, server code fileout from topaz or via server methods may include the fileformat command, which may specify utf8 (for UTF-8 encoded fileouts) or 8bit (for traditional GemStone format), depending on the mode of the GemStone image and the specifics of the fileout.

Filein using GS-Filein does not error on a fileformat command in 7.6.1 and later, but the argument (utf8 or 8bit) is not used. GBS relies on VW's file management, so the file is interpreted as UTF-8.

UTF-8 and 8-bit encoding is the same for the ASCII range, and Characters with codepoints over 255 cannot be filed out in 8-bit. But Characters with code points in the range 128...255 are ambiguous; the bytes in the file will produce different results depending on if they are interpreted as UTF-8 or 8bit.

If the topaz fileout has fileformat 8bit, and the contents includes Characters in the range 127..255, then filing in using GBS GS-filein will cause these characters to be incorrectly read as UTF-8, and the resulting methods will be corrupt.

Workaround:

Avoid using GBS to file in code that was filed out from topaz, unless your image is configured to use Unicode Comparison Mode or you have otherwise ensured that fileouts are always UTF-8.

It is reliable to use GS-File in for code that was filed out using GBS, and topaz file of code filed out using topaz.

Bug 45696 - Using instancesAreForwarders without a class connector can cause problems

Product: GemBuilder for Smalltalk/VW

Versions: 8.5, 8.4, 8.3, 8.2, 8.1, 7.6.1, 7.6, 7.5, 7.4.1, 7.3.3, 7.3.2, 7.1.2, 6.2

Platform All Platforms

Impact: Informational

Client classes that implement the class side method #instancesAreForwarders to return true should also establish a class connector for this class. This is to insure that a replication clamp is setup on the server during login that recognizes the client will be treating instances of this class as a forwarder, and not attempt to ship back for replication unnecessary data.

Usually this unnecessary data is just ignored by the client, but in certain cases can cause spurious errors. For example, if the GBS client has generateClientClasses disabled, and the unnecessary data sent back happens to contain an instance of a class not known by the client, this will trigger a "No client class... class generation is disabled" error.

Workaround:

Establish class connectors for all classes implementing #instancesAreForwarders.

Bug 45694 - Superclass recompile by filein causes Hierarchy Browser to report incorrect superclass version

Product: GemBuilder for Smalltalk/VW

Versions: 8.2, 8.1, 7.6.1, 7.6, 7.5, 7.4.1, 7.4, 7.3.3, 7.3.2, 7.3.1, 7.3, 7.2.2, 7.2.1, 7.2, 7.1.2

Fixed In: 8.3

When you file in code that defines a class that already exists, such that a new version of the class is created, it does not automatically recompile the subclasses of this class. Any subclasses continue to be subclasses of the previous version of the class.

The Hierarchy Browser, however, reports that the superclass of the class is the new version, and displays the new version of the superclass. This can be confusing if the protocol in the superclass is different, since methods in the new superclass version will not be understood.

Bug 45614 - Changing user's default security policy in the User Editor throws error

Product: GemBuilder for Smalltalk/VW

Versions: 8.2, 8.1, 7.6.1, 7.6

Fixed In: 8.3

The User Editor allows you to edit attributes of GemStone users. However, attempting to edit the default security policy resulted in an error indicating you do not have privileges to change this.

Workaround:

Being a member of DataCuratorGroup allows editing the default security policy.

Alternately, use GemStone Smalltalk code directly to perform this operation. For example,

```
(AllUsers userWithId: 'john_smith') defaultObjectSecurityPolicy: nil
```

Bug 45563 - In an Inspector on a very large object, the basic tab gets an error

Product: GemBuilder for Smalltalk/VW

Versions: 8.5, 8.4, 8.3, 8.2, 8.1, 7.6.1, 7.6, 7.5, 7.4.1, 7.4, 7.3.3, 7.3.2, 7.3.1, 7.3, 7.2.2, 7.2.1, 7.2, 7.1.2

An Inspector on a very large object, such as a String or Array of size 4837800 or larger, will throw a Bad Offset exception if the basic tab is selected.

Bug 45392 - Creating a user in User Editor changes the owner of the selected default Security Policy

Product: GemBuilder for Smalltalk/VW

Versions: 8.2, 8.1, 7.6.1, 7.6

Fixed In: 8.3

When creating a user in the User Editor, you must select an Object Security Policy, which may be an existing security policy. The selected Object Security Policy has its owner changed to this user, which is incorrect and may create problems for any other users using that Object Security Policy.

Workaround:

Use GemStone code to set a default Object Security Policy for a user.

For example,

```
    (AllUsers userWithId: 'john_smith')
        defaultObjectSecurityPolicy: PublishedObjectSecurityPolicy
```

Bug 45384 - After GS64 3.x server upgrade, GBS browsers get errors

Product: GemBuilder for Smalltalk/VW

Versions: 7.6.1, 7.6

After a GemStone/S 64 Bit server upgrade to version 3.x, all methods must be recompiled from instances of GsMethod to instances of GsNMethod before they can be executed.

If there are server methods that are not recompiled, attempting to browse these methods in GBS results in an MNU on _allBreakpoints.

Workaround:

In GemStone/S 64 Bit server versions 3.0 through 3.2.6, log in as SystemUser, file in the following method, and commit. This method is present, and the bug is fixed, in server versions after 3.2.6.

```
category: 'Debugging Support'
method: GsMethod
_allBreakpoints

 "Returns nil if no method breakpoints set in the receiver."

^ nil
%
```

Bug 45321 - Debugging: Step on an interval #do will skip down to the end of the code

Product: GemBuilder for Smalltalk/VW

Versions: 8.5, 8.4, 8.3, 8.2, 8.1, 7.6.1, 7.6, 7.5, 7.4.1, 7.4, 7.3.3, 7.3.2, 7.3.1, 7.3, 7.2.2, 7.2.1, 7.2, 7.1.2, 7.1.1, 7.1, 7.0.2, 7.0.1, 7.0, 6.2, 6.1, 6.0

Platform All Platforms

Fixed In: with GS64 3.3

If you are debugging code in GBS that contains an interval #do: loop like the following:

```
...
(1 to: 3) do: [ ... loop code ... ].
... subsequent code ...
```

And step through the loop code, at the end of the loop code, rather than stepping into the subsequent code you will instead complete execution of the subsequent code and end up on the "GbsSession>>gsDoIt: codeString inContext: receiver" frame in the debugger.

This is related to server bug 45320, which is fixed in GemStone/S 64 Bit v3.3 and later

Workaround:

Replace interval #do: loops like:

(1 to: 3) do: [...]

with a #to:do: loop like:

1 to: 3 do: [...]

Bug 45207 - Creating new user with nil security policy does not work

Product: GemBuilder for Smalltalk/VW

Versions: 8.2, 8.1, 7.6.1, 7.6

Fixed In: 8.3

GemStone/S 64 Bit versions allow a Security Policy (Segment) to be set to nil, to bypass object level security. Attempting to create a user with a nil default security policy using the User Editor created a new Security Policy with the name 'nil'.

Workaround:

If your users have security policies of nil, use GemStone server code to create new users. For example,

```
 AllUsers addNewUserWithId: 'john_smith' password: 'swordfish'
```

or

```
 AllUsers
   addNewUserWithId: 'john_smith'
   password: 'swordfish'
   defaultObjectSecurityPolicy: nil
   privileges: (Array new)
   inGroups: (Array with: 'DataCuratorGroup')
```

Bug 45118 - On logout, Inspectors on server objects remain open but object is "Not cached"

Product: GemBuilder for Smalltalk/VW

Versions: 8.2

Fixed In: 8.3

Inspectors on server objects that are not replicated to the client should be closed when the session ends. However, the inspectors remain open, but since the server object is unavailable, the content is the ByteString "Not cached".

Bug 45100 - Debugger close opens a second debugger; subsequent run may hang connection

Product: GemBuilder for Smalltalk/VW

Versions: 8.3, 8.2, 8.1, 8.0, 7.6.1, 7.6, 7.5, 7.4.1, 7.4

Platform With GS64 v3.1.x and 3.2.x

Impact: Critical

If the debugger is opened as a result of encountering a breakpoint, and the debugger window is closed, in some cases a second debugger will appear on the same code. If you press run on this second debugger, the GemStone connection may become stuck.

This is due to a server bug in which the terminate message used by GBS did not work correctly in GemStone/S 64 Bit 3.1.x and 3.2.x versions; it does not occur with GemStone/S 64 Bit 2.4.x or 3.3.x.

Workaround:

If you see this second debugger, use the windowing system close, do not attempt to restart execution.

Bug 44931 - Class method file-out from implementors/senders of window produces incorrect output

Product: GemBuilder for Smalltalk/VW

Versions: 8.4, 8.3, 8.2, 8.1, 7.6.1, 7.6, 7.5, 7.4.1, 7.4, 7.3.3, 7.3.2, 7.3.1, 7.3, 7.2.2, 7.2.1, 7.2, 7.1.2, 7.1.1, 7.1, 7.0.2, 7.0.1, 7.0, 6.2, 6.1, 6.0

Platform All Platforms

Fixed In: 8.5

Filing out a class-side method from an "Implementors of" or "Senders of" window produces an output file where the name of the class is printed with the additional text "class" appended to the actual class name. For example, rather than getting:

```
   classmethod:  MyClass
```

the output is:

```
   classmethod: MyClass class
```

Note that this does not cause any problems while filing in, as topaz will accept the command and simply print out the warning message:

```
   The rest of the command line (' class') was ignored.
```

Workaround:

No workaround is necessary as topaz will ignore the additional text.

Bug 44702 - Copied session parameter disappears if moved

Product: GemBuilder for Smalltalk/VW

Versions: 8.2, 8.1, 8.0., 7.6.1, 7.6

Fixed In: 8.3

In the GemStone Launcher, you may copy a session parameters, creating a second parameters with the same name and login information. If you move this parameter to a new position in the list, it disappears.

Workaround:

Edit the copied parameters, so the two parameters are not identical, before moving it.

Bug 44700 - Failed to find or generate a server class for client class or metaclass Metaclass

Product: GemBuilder for Smalltalk/VW

Versions: 7.6.1, 7.6, 7.5, 7.4.1, 7.4

Platform All

Fixed In: 8.1, 8.0

GBS logins can sometimes fail with a GbsClassGenerationError, reported as "Failed to find or generate a server class for client class or metaclass Metaclass".

The defect is exposed when growing the 'server map' during a server interaction. If VisualWorks runs a garbage collection after the server map is resized but before the server interaction finishes, the delegates, and their client object mappings, that were added to the extended portion of the server map will also be garbage collected.

The server map associates client objects with the corresponding delegate to the server object. It is initialized with a capacity of 2^30 (one billion) objects.

When GBS adds a delegate for an object whose oop is greater than its current capacity, GBS has to grow the server map. The new server map is created as holding weak references, and the added content of this map, holding the new delegate, is also set up with weak references.

During a server interaction, the server map is set to hold strong references, in order to hold the graph of returned objects. Because the new portion of the server map is weak, a garbage collection at the wrong time can remove the newly mapped delegates, triggering this error while trying to re-fetch the results of the interaction.

Workaround:

Download and install the following patch:

Gbs7.6.1-Patch44700.st

Bug 44475 - Debugger stack arguments incorrect

Product: GemBuilder for Smalltalk/VW

Versions: 7.6.1, 7.6

Platform With GS64 3.x servers

When stepping through code in the debugger, the upper right panes should show the stack arguments, the immediate argument and the top of stack.

When running against GemStone/S 64 Bit 3.x servers, these arguments are not correct. The variable names are not provided, instead displaying .t1, .t2, etc. With most versions, there are additional entries (referring to other method variables). In versions 3.2.2 through 3.2.6, the method variables are not present, and the immediate argument is also not included.

Bug 44403 - #replicationSpec not utilized on first replication if class not connected

Product: GemBuilder for Smalltalk/VW

Versions: 8.5, 8.4, 8.3, 8.2, 8.1, 7.6.1, 7.6, 7.5, 7.4.1, 7.4, 7.3.3, 7.3.2, 7.3.1, 7.3, 7.2.2, 7.2.1, 7.2, 7.1.2, 7.1.1, 7.1, 7.0.2, 7.0.1, 7.0

Platform All Platforms

Fixed In: n/a

Impact: Informational

If you have a class MyClass defined on both client and server, with a #replicationSpec defined for that class on the client, and *do not* have a class connector established for this class, then the very first time an instance of this class is replicated from the server to the client, the replicated instance on the client will not take the #replicationSpec into account. For example, if an instvar is configured as #forwarder, the first replicated instances will replicate the contents instead.

During the replication of this first instance, an auto-generated class connector will be made and all subsequent replicated instances will be correct.

Workaround:

To use a replicationSpec when replicating instance to the client, you must setup the class connector before performing the replication.

Bug 44381 - Recompile in debugger does not handle compiler errors properly

Product: GemBuilder for Smalltalk/VW

Versions: 7.6.1, 7.6

Fixed In: 8.1

If you recompile a method in the debugger, and the code changes you accept include a compiler error (such as a missing paren, etc.), the compiler error is written to the transcript, and the debugger appears to ignore the accept.

Workaround:

Check the transcript for details, and correct the syntax error.

Bug 44374 - MNU on commit if browsers reference removed methods

Product: GemBuilder for Smalltalk/VW

Versions: 8.5, 8.4, 8.3, 8.2, 8.1, 7.6.1, 7.6

Platform All

When methods are removed, while being displayed in other browsers such as a method list browser, the other browser will display them in red italic and with text indicating that they have been removed.

On commit, when this browser referencing the removed method is updated, it throws an MNU exception.

Workaround:

The commit has not failed, so the error can be ignored. Closing the browser on the removed methods avoids further errors.

Bug 44353 - Removing most recent version of a server class causes problems

Product: GemBuilder for Smalltalk/VW

Versions: 8.5, 8.4, 8.3, 8.2, 8.1, 7.6.1, 7.6, 7.5, 7.4.1, 7.4, 7.3.3, 7.3.2, 7.3.1, 7.3, 7.x

GemStone server classes can have multiple versions, in which the latest version is the one on which GBS tools operate. The class version browser allows you to remove versions of a server class. However, if you remove the latest version, the change is not handled correctly; the SymbolDictionary remains referring to the removed class version, which is displayed by most GBS browsers, while the class's classHistory does not include that version.

Bug 44245 - Debugger stepping does not highlight in in methods with DoubleByteString source

Product: GemBuilder for Smalltalk/VW

Versions: 7.6.1, 7.6, 7.5

Fixed In: with GS64 3.2.2

When a server method has DoubleByteString source, stepping though the method in the debugger works, but the currently step points are not highlighted in the debugger's method pane.

Bug 44209 - Inspector cannot inspect Sets or Bags larger than 2000

Product: GemBuilder for Smalltalk/VW

Versions: 8.2, 8.1, 7.6.1, 7.6

Fixed In: 8.3

The inspector for instances of Set and Bag does not handle Sets or Bags with more than 2000 elements. The inspector will report that an error occurred with MessageNotunderstood on bitAnd:.

Workaround:

Download and install the following workaround:

`GbxCollectionCache-setExecutionString.st`

Bug 44180 - Dynamic instance variables not displayed

Product: GemBuilder for Smalltalk/VW

Versions: 8.3, 8.2, 8.1, 7.6.1, 7.6, 7.5

Platform with GS64 v3.0 and later

Fixed In: 8.4

GemStone/S 64 Bit v3.0 and later support dynamic instance variables, which are symbol/value pairs associated with specific instances of any class. GBS inspectors do not display any dynamic instance variables associated with a server object that is being inspected.

Note that dynamic instance variables do not replicate to GBS.

Bug 44106 - Find Class... errors for SymbolDictionary entries that are not classes

Product: GemBuilder for Smalltalk/VW

Versions: 8.2, 8.1, 7.6.1, 7.6

Fixed In: 8.3

When a SymbolDictionary contains an key-value pair in which the value is not a Class, using the Find Class... menu item to search for the value results in a server compile error, undefined symbol 'name'.

Workaround:

You can determine which SymbolDictionaries in your SymbolList contain a particular key by executing server code of the form:

```
 | sd |
sd := System myUserProfile symbolList select: [:ea | ea includesKey: mySymbol].
sd collect: [:ea | ea keys detect: [:key | (ea at: key) = ea ] ifNone: [nil].]
```

Bug 44103 - Hierarchy implementors misses implementors in lower subclasses

Product: GemBuilder for Smalltalk/VW

Versions: 8.1, 7.6.1, 7.6

Fixed In: 8.1

Hierarchy implementors should return implementors in all superclasses and all subclasses of the selected Class. However, it only descended to the first level of subclasses, and therefore missed implementors in any subclasses below that.

Bug 43547 - Connector to missing server class raises exception on login

Product: GemBuilder for Smalltalk/VW

Versions: 7.6

Platform All Platforms

Fixed In: 7.6.1

If your session connectors include a connection to a class that does not exist on the server, attempting to login will raise the exception: "Failed to find or generate a server class for client class or metaclass".

Workaround:

Remove the offending connection.

Bug 43149 - GbsSessionParameters removeClassConnectorsFor: may fail

Product: GemBuilder for Smalltalk/VW

Versions: 7.6.1, 7.6, 7.5, 7.4.1, 7.4, 7.3.3, 7.3.2, 7.3.1, 7.3, 7.2.2, 7.2.1, 7.2, 7.1.2, 7.1.1, 7.1, 7.0.2, 7.0.1, 7.0, 6.2, 6.1, 6.0

Fixed In: 8.1

Class connectors may be created with the class name as a String or as a Symbol, depending on code paths. If the class connector's #gsName is a String, removeClassConnectorsFor: will fail with an error.

Workaround:

Change the method GbsSessionParameters >> removeClassConnectorsFor: to send #asString to #name and #gsName.

Bug 43069 - File out of multiple selected method errors

Product: GemBuilder for Smalltalk/VW

Versions: 7.6

Fixed In: 7.6.1

If two methods are selected, an attempt to file out the methods using Method
menu item File out As... will result in an error.

Workaround:

File out methods individually, or by category.

Bug 43056 - Connector Browser incorrect status for Session connectors

Product: GemBuilder for Smalltalk/VW

Versions: 7.6

Platform All

Fixed In: 7.6.1

In the Connector Browser, you can create and manage global and session connectors. While session connectors can be created correctly, the disconnected/connect status will incorrectly state that the connectors are not connected, even when they are connected. Attempting to inspect the server object will result in an error message that no appropriate logged in session is available.

Workaround:

The connectors are connected correctly and there is no underlying problem, so programmatically managing connectors is unaffected. Global connectors correctly show their status.

Bug 43046 - Versioning class in Class/Hierarchy Browsers does not update view of class version

Product: GemBuilder for Smalltalk/VW

Versions: 8.2, 8.1, 7.6.1, 7.6, 7.5, 7.4.1, 7.4, 7.3.3, 7.3.2, 7.3.1, 7.3, 7.2.2, 7.2.1, 7.2, 7.1.2

Platform All

Fixed In: 8.3

If you modify a class's class definition in the Class Hierachy (and in some versions, the Class) Browsers, thus creating a new version of the class, the browser keeps the reference to the previous version of the class. As a result, the changes you made in the class are not visible. In addition, in some versions, the class definition reports for inDictionary: that "(class not in your dictionaries)".

Workaround:

Modify class definitions using the System Browser.

Bug 43044 - Adding instance variables using instance variable tab menu does not add methods to new class version

Product: GemBuilder for Smalltalk/VW

Versions: 8.5, 8.4, 8.3, 8.2, 8.1, 7.6.1, 7.6

Platform All

Version 7.6 introduces new menu functionality on the instance variable tab, next to the class and instance tabs that control visibility of class and instance methods. The Category menu Add... menu item allows you to add an instance variable to the selected class.

However, modifying the class definition in this way creates a new class version, but does not also bring the methods forward to the new class version. This means that the modified class that is visible in your browser has no methods.

Workaround:

Do not use this functionality. To add an instance variable, modify the class defintion on the class pane, or use the Class menu item Add > Instance Var. In later versions, this operation has been disabled.

Bug 43042 - Removing instance variables using instance variable tab menu is broken

Product: GemBuilder for Smalltalk/VW

Versions: 8.5, 8.4, 8.3, 8.2, 8.1, 7.6.1, 7.6

Platform All

Version 7.6 introduces new functionality on the instance variable tab, next to the class and instance method tabs that control visibility of class and instance methods. The remove menu item is designed to allow you to remove an instance variable. This incorrectly modified the class definition so other instance variable names are merged.

Workaround:

Do not use this function. Editing the class definition in the class pane will allow you to remove an instance variable. In later versions, these operations have been disabled.

Bug 43015 - Auto-commit not usable in manual transaction mode

Product: GemBuilder for Smalltalk/VW

Versions: 7.6

Platform All

Fixed In: 7.6.1

Impact: Informational

Auto-commit is a new feature in v7.6 that allows code changes to be automatically committed to the repository as part of accept. This feature is only intended to be used with a session that is in automatic transaction mode, which is the default. If the session is changed to manual transaction mode, accepting code changes, which triggers the automatic commit, will fail with an error.

Workaround:

Do not enable auto-commit if running in manual transaction mode.

Bug 42965 - On Inspector methods tab, browse local senders/implementors gets walkback.

Product: GemBuilder for Smalltalk/VW

Versions: 7.5, 7.4.1, 7.4, 7.3.2, 7.3.1, 7.3, 7.2.1, 7.2

Fixed In: 7.6

The Inspector on a GemStone object includes a tab labeled "GS Methods". This tab allows you to view GemStone server methods for the inspected object. The method tab pop-up menu on this tab includes options "Local Senders" and "Local Implementors", which fail with MNU #includesBehavior:

These menu options have been renamed to "Senders" and "Implementors" in v7.6.

Bug 42940 - Debugger cannot step into blocks with 3.x server

Product: GemBuilder for Smalltalk/VW

Versions: 7.6.1, 7.6

Platform With GS64 3.x server

Fixed In: 7.6.1 (with 3.2 server)

When logged into a 3.x server earlier than v3.2, the debugging steps over blocks when it should step into the block.

The fix for this requires changes in both GBS, which are present in 7.6.1 and later, and changes in the GemStone/S 64 Bit server, which are present in v3.2 and later. This bug will manifest when using v7.6 with any 3.x version, or when using 7.6.1 with server versions 3.0.x or 3.1.x.

Bug 42784 - Code formatting errors with GemStone-specific syntax

Product: GemBuilder for Smalltalk/VW

Versions: 8.1, 7.6.1, 7.6

Platform All

Fixed In: 8.2

The option to format GemStone server source code relies on the underlying VisualWorks code formatter. This code formatter does not understand the GemStone-specific syntax using curly braces {}. This includes both indexed query syntax and the 3.x Array constructor syntax. Attempting to format code that include indexed queries or 3.x Array constructors will result in "Unknown character".

Bug 42669 - In debugger, GS functions on menu bar Edit menu broken

Product: GemBuilder for Smalltalk/VW

Versions: 7.5, 7.4.1, 7.4, 7.3.3, 7.3.2, 7.3.1, 7.3, 7.2.2, 7.2.1, 7.2, 7.1.2, older versions

Platform All

Fixed In: 7.6

The debugger provides both a pop up menu in the method text pane, and an
Edit item on the menu bar, with the functions GS-Do it, GSiPrint it, and
so on appearing on both.

While these functions on the pop-up menu work correctly, from the Edit
menu they result in a walkback with a error of the form Message not understood:
#gbxGSDoIt.

Workaround:

Use the pop-up menu function

Bug 42577 - Login hang/out of memory with server Seaside keyfile

Product: GemBuilder for Smalltalk/VW

Versions: 7.5, 7.4.1, 7.4, 7.3.3, 7.3.2, 7.3.1, 7.3, 7.2.2, 7.2.1, 7.2, 7.1.2, 7.x

Platform All

Fixed In: 7.6

GBS applications cannot log into servers that have been started with a keyfile that does not have traversal permission, such as the web edition keyfile that comes with GemStone Seaside (GLASS). You cannot login from GBS to applications that use this limited-functionality free Seaside keyfile.

Attempting to login from GBS into a server that does not support traversal does not error as it should; the login attempt hangs with infinite recusion as the login attempts to use traversal to get details on the error. The image eventually runs out of memory and exits.

Workaround:

Restart your server with a keyfile that allows traversal. If you do not have an appropriate keyfile, contact keyfiles@gemtalksystems.com. Note that unlimited GemStone keyfiles require product purchase or other specific arrangements.

Bug 42492 - Login failures when connectors involve large Integers

Product: GemBuilder for Smalltalk/VW

Versions: 7.5, 7.4.1, 7.4, 7.3.3

Platform All

Fixed In: 7.6

If there is a connector that connects objects involving Large Integers
(that is, Integers that are outside the SmallInteger range), then on login,
the connector initialization and therefore login fails, with the error
nil does not understand newFromGSObjectReport:class:.

This may occur if a named connector connects two large integers, or if
the OOP of an class is a large integer, or similar scenarios.

This bug is much more likely to show up when logged into 32-bit GemStone/S,
which has a much smaller SmallInteger range than GemStone/S 64 Bit.

Workaround:

A patch fix for this is to edit GbsSession>>postLogin, moving the line:

```
    integerConverter := (GbxServerVersion current integerConverterClass: self)
new.
```

from its current position late in the method, and instead make it the second line in the method, right after the line:

```
    self isLoggedIn ifFalse: [^self].
```

Bug 42447 - Create or compile in ST yields walkback, unprotected ServerMap access

Product: GemBuilder for Smalltalk/VW

Versions: 7.5

Fixed In: 7.6

In a GBS browser, selecting a class and doing a "create in ST" or "compile in ST" yields a walkback, 'GBS Error - ServerMap access without sessionProtect'.

Bug 42446 - Create in ST silently does nothing

Product: GemBuilder for Smalltalk/VW

Versions: 7.5, 7.4.1, 7.4, 7.3.3, 7.3.2

Fixed In: 7.6

The GBS browser menu item "create in ST" does not do anything, and does not report an error.

Since the menu item "compile in ST" first performs a create in ST, this operation may fail with a variety of errors, such as reporting that the client class does not exist.

Note that in version 7.5, while technically this bug exists, it is not observed due to bug #42447.

Bug 42445 - VW's feature to write stacks to log file causes MNU on server exceptions

Product: GemBuilder for Smalltalk/VW

Versions: 7.5, 7.4.1, 7.4, 7.3.3, 7.3.2, 7.3.1, 7.3

Platform VW

Fixed In: 7.6

VW permits you to edit the method DebuggerService>>logErrorFor:label: to have error stacks automatically written to a file. If this change is made, then on GemStone server error, a inspector on MessageNotUnderstood pops up, as well as the error notification.

Workaround:

A workaround (not the official patch) can be made:

In the method DebuggerService>>logErrorFor:label:, change the line:

```
  cp := ContextDumpPolicy new initialize. to read:
  cp := GbxContextDumpPolicy new initialize.
```

With this change, server side stacks are written to the log, no inspector pops up, and client side stacks will still write to the file. With this workaround, the client side stacks will follow the GBS stack display format.

Bug 41695 - DoubleByteStrings cause server compiler errors

Product: GemBuilder for Smalltalk/VW

Versions: 7.4

Platform Windows and Linux clients, against GS64 3.0 on any platform

Fixed In: 7.4.1

The way DoubleByteStrings are handled by the GCI changed in 3.0; as a result, TwoByteStrings from GBS little-endian clients (Windows and Linux) are incorrectly swizzled, resulting in server compiler errors when attempting to compile source code that is a DoubleByteString.

Since any use of a Character outside the single byte range modifies the underlying VW tools to use a TwoByteString, you will also see compiler errors performing any evaluate from a workspace whose text previously included a large Character (for example, the result of a print-it). This error occurs even though the code being evaluated has no large characters.

Bug 41331 - Do not disconnect clientForwarder connectors

Product: GemBuilder for Smalltalk/VW

Versions: 8.5, 8.4, 8.3, 8.2, 8.1, 7.6.1, 7.6, 7.5, 7.4.1, 7.4, 7.3.3, 7.3.2, 7.3.1, 7.3, 7.2.2, 7.2.1, 7.2, 7.1.2, 7.1.1, 7.1, 7.0.2, 7.0.1, 7.0, 6.2, 6.1

Impact: Informational

Disconnecting a connector with the post connect action of #clientForwarder may result in errors, since the clientForwarder remains on the server.

Workaround:

Sessions using connectors with a post connect action of #clientForwarder should not explicitly disconnect these connectors. The connectors will be cleared on session logout.

Bug 41310 - Instances of subclasses of Array may turn into Arrays on client

Product: GemBuilder for Smalltalk/VW

Versions: 7.3.3, 7.3.2, 7.3.1, 7.3, 7.2.2, 7.2.1, 7.2

Platform VW

Fixed In: 7.4

If you have an instance of a class that is a subclass of Array, and the class on the client changes size, the class of the instance on the client may change to Array. Since GemStone Arrays may change size, but VW Arrays may not, when the replicate changes size a new instance must be created.

This new instance was incorrectly created as an instance of Array.

Workaround:

Modify the method Array >> indexableSize:, by changing "Array" to "self class".

Bug 41120 - Groups nil segment causes walkback in Segment tool

Product: GemBuilder for Smalltalk/VW

Versions: 7.3.2, 7.3.1, 7.3

Platform GS64 only

Fixed In: 7.3.3

If AllGroups segment is set to nil, creating a group in the Segment tool
results in a walkback.

Bug 41115 - Incorrect replication of object in certain interconnected graphs

Product: GemBuilder for Smalltalk/VW

Versions: 7.3.2, 7.3.1, 7.3

Platform 32-bit GS/S

Fixed In: 7.3.3

Impact: Critical

Under certain conditions of interconnected object graphs, after replication
of the object graph, an OrderedCollection or Set that contains objects
may be empty. If this is committed, this corruption may become persistent.

The condition requires that the object graph references an OrderedCollection
or a Set, or one of their subclasses, with a level one less than that required
to get a full report for the OrderedCollection or Set. The report for
this empties any already-replicated client replicate of that OrderedCollection
or Set.

Bug 41045 - Post-Lost OT processing may unstub stubs or synchronize incorrectly

Product: GemBuilder for Smalltalk/VW

Versions: 7.3.2, 7.3.1, 7.3

Platform VW

Fixed In: 7.3.3

After a lost OT Root from the server, all client replicates must be updated,
since synchronization with the server has been lost.

The code that performed this collected replicates in a Dictionary, which
required sending #hash to the entry, as well as risking incorrect synchronization
if replicates compared equal.

Bug 40891 - Unreadable text for errors on server MultibyteString

Product: GemBuilder for Smalltalk/VW

Versions: 8.2, 8.1, 7.6.1, 7.6, 7.5, 7.4.1, 7.4, 7.3.3, 7.3.2, 7.3.1, 7.3, 7.2.2, 7.2.1, 7.2, 7.1.2

Platform All

Fixed In: 8.3

If a server error occurs on a DoubleByteString or QuadByteString - for example, a DoubleByteString is sent a message that is not implemented for DoubleByteString - the error that gets reported to GBS has a bad error text. The error report does not have a mechansim to determine if the resulting error text is not a single byte String.

Bug 40874 - Missing data in replicated objects after server error

Product: GemBuilder for Smalltalk/VW

Versions: 7.3.3, 7.3.2, 7.3.1, 7.3, 7.2.2, 7.2.1, 7.2

Platform With GS64 only

Fixed In: 7.4

Impact: Critical

When server objects that were replicated to the client are modified on the server, updates to the replicated objects are sent to the client for each server operation. If an error occurs during a server operation, when there were modifications on the server to replicated objects, there may be incomplete updates under some circumstances. The problematic cases involve hashed collections, circular reference paths, and instances of classes in which the class is not on the client. In these cases, valid object references may be nil on the client. If the replicates are modified, it may result in data loss.

This bug is in the Single-trip protocol, which is used with the GemStone/S 64 Bit server. This bug does not affect systems using 32-bit GemStone/S.

Workaround:

File the following into the VW image:

```
<?xml version="1.0"?>

<st-source>

<methods>
<class-id>GemStone.Gbs.GbxServerInteraction</class-id> <category>private -
executing</category>

<body package="GbsServerInterface" selector="fetchReplicationClamp">fetchReplicationClamp
        "If the replication clamp has not been explicitly set,
        find or create and answer the replication clamp for this interaction.
        If the replication clamp must be created, this will involve other server
interactions.
        If the replication clamp must be created, creates another ServerInteraction and
executes it
        to create the clamp. Therefore, the sender should *NOT* hold my sessionProtect.
The session
        semaphore is not needed to create a class connector, so the replication clamp
could be flushed
```

```
        at any moment even if we did hold sessionProtect, so there would be no advantage "
        "Even if we're not replicating results, we need a clamp to handle traversal of
altered immediate-faulting objects."

        replicationClampDelegate ~~ nil ifTrue: [^replicationClampDelegate].
        ^replicationSpecSelector ~~ nil
                ifTrue: [session replicationClampForSpecSet: replicationSpecSelector]
                ifFalse: [session oopNil]</body>

<body package="GbsServerInterface"
selector="replicateAlteredObjectsAfterError:">replicateAlteredObjectsAfterError: error
        "Used after an error interrupted execution. Note that an 'error' includes things
like
        client forwarder sends and breakpoints.

        Normally, answer the given error. However, if we run into an error replicating the
altered objects,
        that error is probably worse than the original error, so answer the new error."

        | serverTraversal result |
        serverTraversal := serverInterface newServerTraversal.
        "stopForBreakpoints false to avoid asynch events at this point."
        "Only traversing the altered objects, no actual result, so rootDelegate nil."
        "Explicitly use my replication clamp. If the session's replication clamp has been
flushed
        and we tried to recompute it at this point we'd get the altered objects with a nil
        replication clamp and choke on any objects that should have been traversed by
callback."
        "Similarly, cannot cache GsInstVarNames at this point, so shouldSynchronize must
be false."
        result := serverTraversal
                                shouldReplicateResult: false;
                                shouldSynchronize: false;
                                replicationClampDelegate: replicationClampDelegate;
                                stopForBreakpoints: false;
                                rootDelegate: session oopNil;
                                execute.
        "Shouldn't normally error here, but need to check in case of things like
        session termination, or server out of memory."
        ^result gbxIsGSError ifTrue: [result] ifFalse: [error]</body>
</methods>

</st-source>
```

Bug 40509 - Incorrect connectors for String classes

Product: GemBuilder for Smalltalk/VW

Versions: 7.3.2, 7.3.1, 7.3

Platform VW

Fixed In: 7.3.3

In GBS releases earlier than 7.3, GBS used the Locale's stream encoder to determine what client class to map to the server ISOLatin class. This encountered problems in some OS environments due to changes in VW 7.6; see bug #38785.

The fix for this bug was to always map the server ISOLatin class to a new GBS class, GemStone.Gbs.ISOLatin class. However, this results in no mapping for MSCP1252String or ISO8859L1String. If instances of these client specific Strings are created, such as by reading a file, replicating them to the server resulted in errors.

Workaround:

Remove the connector from GemStone.Gbs.ISOLatin to ISOLatin. Create a connector from MSCP1252String to ISOLatin (on Windows) or ISO8859L1String to ISOLatin (on Unix).

Bug 39319 - asGSObjectCopy will not copy collection instance variables

Product: GemBuilder for Smalltalk/VW

Versions: 7.6, 7.5, 7.4.1, 7.4, 7.3.3, 7.3.2, 7.3.1, 7.3, 7.2.2, 7.2.1, 7.2, 7.1.2, 7.1.1, 7.1, 7.0.2, 7.0.1, 7.0

Fixed In: 7.6.1

Object>>asGSObjectCopy is deprecated public protocol which creates a copy of a client object, then creates the server object. This prevents the original client object from becoming a replicate and synchronization is done to the copy. However, if a dictionary or set subclass is created which has instance variables, #asGSObjectCopy will not copy those instance variables.

Workaround:

Do not use #asGSObjectCopy as it will be removed in a future release.

Bug 39230 - GBS gives constraint violation when class history removed

Product: GemBuilder for Smalltalk/VW

Versions: 8.3, 8.2, 8.1, 7.6.1, 7.6, 7.5, 7.4.1, 7.4, 7.1.2, 7.1.1, 7.1, 7.0.2, 7.0.1, 7.0, 6.2

Instance variable constraints are specified in a server class definition by specifying a version of the class to which the inst var is constrained. If the version of the constrained class has been purged from the class version history, GBS may raise an error. This is because the constraint is pointing to a version of the class that no longer exists.

This only applies to GemStone/S servers.

Workaround:

Make sure all constraints specify the latest version of the class.

Bug 39188 - Out of Range error evaluating code in debugger

Product: GemBuilder for Smalltalk/VW

Versions: 8.4, 8.3, 8.2, 8.1, 7.6.1, 7.6, 7.5, 7.4.1, 7.4, 7.3.3, 7.3.2, 7.3.1, 7.3, 7.x

Fixed In: with GS64 3.1

Evaluating an expression within a debugger pane may result in an out of range error.

This is an issue with the server code that supports the debugger, and is fixed in GemStone/S 64 Bit v3.1 and later.

Workaround:

Copy code to a workspace to evaluate, or open an inspector on an object and execute code in that context.

Bug 39182 - Restarting an anonymous server context does not reset execution state

Product: GemBuilder for Smalltalk/VW

Versions: 8.4, 8.3, 8.2, 8.1, 7.6.1, 7.6, 7.5, 7.4.1, 7.4, 7.3.3, 7.3.2, 7.3.1, 7.3, 7.2.2, 7.2.1, 7.2, 7.1.2, 7.1.1, 7.1, 7.0.2, 7.0.1, 7.0, 6.2

Platform With GemStone/S or GS64 versions prior to 2.3

Fixed In: n/a

If you restart an anonymous GS method against GemStone/S servers or GemStone/64
servers prior to version 2.3, the stack will be trimmed correctly, but
the server state will not be reset accordingly.

Bug 39060 - Accept of server code in debugger can give message not understood.

Product: GemBuilder for Smalltalk/VW

Versions: 7.3.2, 7.3.1, 7.3, 7.2.2, 7.2.1

Fixed In: 7.3.3

An error, message not understood #hasActiveDebugOperation, may be raised occasionally when trying to accept server in the text pane of the debugger. This has to do with subtle timing changes introduced in VisualWorks 7.6.

Workaround:

Make the code change in a GemStone browser.

Bug 38477 - Server contexts missing in debugger during user action or GsFile server errors

Product: GemBuilder for Smalltalk/VW

Versions: 7.6.1, 7.6, 7.5, 7.4.1, 7.4, 7.3.3, 7.3.2, 7.3.1, 7.3, 7.2.2, 7.2.1, 7.2, 7.1.2, 7.1.1, 7.1, 7.0.2, 7.0.1, 7.0, 6.2

Platform With servers earlier than GS64 v3.2

If an error occurs in a user action, or by extension a GsFile method which
uses user actions, an error will be raised. However, the stack in the debugger
will be missing server contexts.

The actual problem is in server code, and is fixed in v3.2

Bug 36774 - Debugger shows incorrect variables in ExecutableBlock methods

Product: GemBuilder for Smalltalk/VW

Versions: 8.3, 8.2, 8.1, 7.6.1, 7.6, 7.5, 7.4.1, 7.4, 7.3.3, 7.3.2, 7.3.1, 7.3, 7.2.2, 7.2.1, 7.2, 7.1.2, 7.1.1, 7.1, 7.0.2, 7.0.1, 7.0, 6.2, 6.1, 6.0

Platform When running with 32-bit servers

When running against a 32-bit server, the context inspector in the debugger
may show the incorrect contextual variables if the selected stack frame
is a method on ExecutableBlock. Instead of showing the values of the variables
within the block, it will show the variables from the previous stack frame,
or outer context. The receiver inspector will show self as the ComplexBlock

instead of the outer context of the block.

Bug 36203 - Image portabilty problem due to load-time String mapping

Product: GemBuilder for Smalltalk/VW

Versions: 8.5, 8.4, 8.3, 8.2, 8.1, 7.6.1, 7.6, 7.5, 7.4.1, 7.4, 7.3.3, 7.3.2, 7.3.1, 7.3, 7.2.2, 7.2.1, 7.2, 7.1.2, 7.1.1, 7.1, 7.0.2, 7.0.1, 7.0

Platform VW

GBS creates global class mappings for Strings based on the appropriate string class for the platform upon which the client image is currently running. So loading GBS into a VisualWorks image will map the appropriate string class to the GemStone class ISOLatin, a subclass of String. However, if this image is saved, and started up on a different platform with a different string class (such as Windows to Unix) strings will not map correctly.

Bug 36153 - Invalid class connectors can make session unusable

Product: GemBuilder for Smalltalk/VW

Versions: 7.6.1, 7.6, 7.5, 7.4.1, 7.4, 7.3.3, 7.3.2, 7.3.1, 7.3, 7.2.2, 7.2.1, 7.2, 7.1.2, 7.1.1, 7.1, 7.0.2, 7.0.1, 7.0, 6.2, 6.1, 6.0

Platform All

Fixed In: 8.0

Invalid class connectors can cause login failures. Upon login, if GemBuilder finds a class connector for which it cannot successfully connect, the user is presented with a dialog asking the user if he or she would like to remove the invalid connector. If the user chooses to say 'No', subsequent openings of the GemStone classes browser will attempt to use that invalid connector and cause errors. It is possible that even if the user chooses 'Yes', errors could occur if the the invalid connector was encountered inside of a flush to the server.

Workaround:

If you encounter a failed connector, select 'Yes' to remove the connector, then logout and log back in. If you wish to keep that connector, resolve the failure before opening the class browser.

Bug 35858 - Library name setting automatically changed from default

Product: GemBuilder for Smalltalk/VW

Versions: 7.4.1, 7.4, 7.3.3, 7.3.2, 7.3.1, 7.3, 7.2.2, 7.2.1, 7.2, 7.1.2, 7.1.1, 7.1

Platform All

If you set the GbsConfiguration setting #libraryName to its default of the empty string, GBS attempts to load a library named "libgcilnk.<ext>", where <ext> is the shared library name extension for the local platform (such as 'dll' or 'so'). If that fails, it attempts to load "libgcirpc.<ext>". This is the correct behavior.

However, if GBS succeeds in loading one of the default libraries, its name is set as the setting of #libraryName. If the image is subsequently saved, this can cause two problems when launching that image in another environment:

1. GBS will only look for a library of the exact name of the library

it found previously. If only the "other" default library is now available, library loading will erroneously fail.

2. If the image has been moved to another platform, loading may fail due to the library name extension being wrong.

Workaround:

Either:

1. Explicitly set the name of the library expected in each environment, or

2. Set #libraryName to the empty string before saving an image which must be portable.

Bug 35756 - Can't Fileout GemStone Method with DoubleByteString

Product: GemBuilder for Smalltalk/VW

Versions: 7.3.2, 7.3.1, 7.3, 7.2.2, 7.2.1, 7.2, 7.1.2, 7.1.1, 7.1, 7.0.2, 7.0.1, 7.0

Fixed In: 7.3.3

If a method contains a DoubleByteString, an error will result if a fileout is attempted on the method.

Bug 35729 - In stack dumps, temporary names and values do not always match

Product: GemBuilder for Smalltalk/VW

Versions: 8.5, 8.4, 8.3, 8.2, 8.1, 7.6.1, 7.6, 7.5, 7.4.1, 7.4, 7.3.3, 7.3.2, 7.3.1, 7.3, 7.2.2, 7.2.1, 7.2, 7.1.2, 7.1.1, 7.1, 7.0.2, 7.0.1

Platform All

In stack dumps produced by the GbsStackDumper, part of the information dumped for each Context is a list of the names and values of the temporary variables in that Context. In dumps of some Contexts, some temporary variable names may be associated with the value of a different temporary variable.

Workaround:

There is no workaround, although it is usually possible by studying the dump to figure out the proper correspondence between temporary variable names and values.

Bug 35386 - Possible to lose modifications through concurrent operations

Product: GemBuilder for Smalltalk/VW

Versions: 7.6.1, 7.6, 7.5, 7.4.1, 7.4, 7.3.3, 7.3.2, 7.3.1, 7.3, 7.2.2, 7.2.1, 7.2, 7.1.2, 7.1.1, 7.1, 7.0.2, 7.0.1, 7.0, 6.2, 6.1, 6.0, 5.2.3, 5.2.2, 5.2.1

Platform All

Fixed In: 8.0

Impact: Critical

If multiple Smalltalk Processes concurrently manipulate the same replicate, it is possible for a change to that replicate to be lost. For this to happen, specific circumstances must be occur in a quite narrow time window.

Specific circumstances:

In most versions of GBS, if one Process modifies a replicate while another process is making the replicate into a stub, the modification can be lost. Replicates become stubs due to either an explicit request or as the result of server execution if the object has changed on the server and its fault policy is #lazy.

In a few versions of GBS, if one Process modifies a replicate while another Process is initiating server execution, the modification can be lost.

When using the GBS automatic dirty-marking feature in VA Smalltalk and VisualWorks 7.x, it is possible for GBS to make these failures impossible. In VisualWorks 5i, making automatic dirty-marking immune to these failures would slow execution down, so we do not anticipate ever fixing this problem in VisualWorks 5i.

All applications which manually mark objects dirty are vulnerable to this problem.

We recommend that all customers use VisualWorks 7.x or VA Smalltalk, and that that their applications use automatic dirty-marking. If you must use VisualWorks 5i or manual dirty-marking, the application must be very careful when accessing sessions and their replicates from more than one Smalltalk Process.

Workaround:

No workaround

Bug 34677 - Debugger may highlight incorrect code when single stepping

Product: GemBuilder for Smalltalk/VW

Versions: 7.6.1, 7.6, 7.5, 7.4.1, 7.4, 7.3.3, 7.3.2, 7.3.1, 7.3, 7.2.x, 7.1.2, 7.1.x, 7.0.x

Platform with GS64 2.x

When single stepping through server code, the debugger highlights the step that is about to be executed. In some cases, an incorrect offset is returned by the server, and the code for a following step point is highlighted. Execution and temporaries are otherwise handled correctly.

The underlying server bug is fixed in GS64 3.x, so this bug does not occur with these server versions.

Bug 32899 - VisualWorks Delays shorter than 10 or 20 milliseconds can take too long

Product: GemBuilder for Smalltalk/VW

Versions: 7.4.1, 7.4, 7.3.3, 7.3.2, 7.3.1, 7.3, 7.2.2, 7.2.1, 7.2, 7.1.2, 7.1.1, 7.1, 7.0.2, 7.0.1, 7.0, 6.2, 6.1, 6.0

Platform VW 7.x

Impact: Informational

Instances of Delay less than 10 or 20 milliseconds can delay for significantly longer than the specified time on some machines. A Delay of 1 millisecond, for example, might actually take 15 milliseconds to return, which can become a significant performance problem when it occurs frequently.

This occurs in VisualWorks but can affect applications using GBS. It has been known to reproduce on Windows and Solaris. It may be present on other platforms as well.

Cincom is tracking this as AR 49402. Improvements have been made in VW

7.7, but the problem is not completely resolveable.

Workaround:

In application code, avoid using Delay to perform short waits to wait for an external condition to be true.

In GBS 6.2 and later, this problem is avoided by changes in the default way polling is done, controlled by the new configuration parameters pollForRpcResponse and pollForAsynchronousEvents. Avoid setting these to true, which causes GBS to utilize many short delays to wait for the server. When set to false, the default, GBS uses socket activity detection rather than delays in server communication.

Bug 32757 - GemStone Smalltalk execution in debugger breaks on explicit return

Product: GemBuilder for Smalltalk/VW

Versions: 8.5, 8.4, 8.3, 8.2, 8.1, 7.6.1, 7.6, 7.5, 7.4.1, 7.4, 7.3.3, 7.3.2, 7.3.1, 7.3, 7.2.2, 7.2.1, 7.2, 7.1.2, 7.1.1, 7.1, 7.0.2, 7.0.1, 7.0, 6.2

Platform VW 7.x

If the currently-selected context in a debugger is a server context, and a GS-Do It, GS-Print It, GS-Inspect It, Or GS-Debug It is performed in the code pane, and the code selected for execution executes an explicit return (^), an error will result.

Workaround:

When debugging in server contexts, avoid selecting code that contains explicit returns, unless you know that the explicit return is on a code path that will not be executed.

Bug 31966 - Cannot suspend or debug server processes through VisualWorks process monitor

Product: GemBuilder for Smalltalk/VW

Versions: 8.5, 8.4, 8.3, 8.2, 8.1, 7.6.1, 7.6, 7.5, 7.4.1, 7.4, 7.3.3, 7.3.2, 7.3.1, 7.3, 7.2.2, 7.2.1, 7.2, 7.1.2, 7.1.1, 7.1, 7.0.2, 7.0.1, 7.0, 6.2, 6.1

Platform VW 7.x

In Visualworks, if you press Ctrl-\(backslash), all processes are suspended and the process monitor is brought up. But, for a process that is doing GemStone execution, only the client process is suspended, not the gem process.

Also when debugging processes that are in GemStone execution, you can debug them, but only the client.

Bug 31001 - GBS debugger can't step into ClientForwarder send

Product: GemBuilder for Smalltalk/VW

Versions: 8.5, 8.4, 8.3, 8.2, 8.1, 7.6.1, 7.6, 7.5, 7.4.1, 7.4, 7.3.3, 7.3.2, 7.3.1, 7.3, 7.2.2, 7.2.1, 7.2, 7.1.2, 7.1.1, 7.1, 7.0.2, 7.0.1, 7.0, 6.2, 6.1, 6.0

Platform VW 7.x

When stepping through GemStone execution in the GBS debugger,
you should be able to step into a client forwarder message.
The debugger should show the client context as the new top
of stack on top of the server context that sent the message.
Instead, the execution point appears to be stuck in the same
place in the server context. Additionally, if you keep trying
to press the step into button, on the fifth time you'll get
an unhandled exception - rtErrStep. If you instead step over
the client forwarder send, it executes correctly.

Workaround:

Either step over client forwarder sends, or if you want to
debug that send, set a client breakpoint in the method
invoked by the client forwarder.

Bug 30994 - VisualWorks platform string classes all map to ISOLatin

Product: GemBuilder for Smalltalk/VW

Versions: 7.3.2, 7.3.1, 7.3, 7.2.2, 7.2.1, 7.2, 7.1.2, 7.1.1, 7.1, 7.0.2, 7.0.1, 7.0, 6.2, 6.1, 6.0

Platform VW 7.x

Fixed In: 7.3.3

If you read a file using VisualWorks on Windows, you get an instance of
MSCP1252String. If you do this on unix, you get an instance of ISO8859L1String.
If you store these into GemStone, they both get mapped to the class ISOLatin.
That's fine if you are only using windows or only using unix clients,
but if you store an MSCP1252String from your windows client and then read
it from GemStone in your unix client, it will be mapped to the wrong class,
and some characters with encodings ranging from 129 to 160 will be interpreted
differently.

Workaround:

Before storing a platform string into GemStone, first convert it to a ByteString
or TwoByteString.

Bug 30986 - GBS PDP debugger args/temps inspector "copy" is broken

Product: GemBuilder for Smalltalk/VW

Versions: 7.6, 7.5, 7.4.1, 7.4, 7.3.3, 7.3.2, 7.3.1, 7.3, 7.2.2, 7.2.1, 7.2, 7.1.2, 7.1.1, 7.1, 7.0.2, 7.0.1, 7.0, 6.2, 6.1, 6.0

Platform VW 7.x

"Copy" always copies nil to the clipboard in the GBS PDP debugger's
arg/temp inspector pane when a GemStone context is selected. "Paste"
does work - you can "copy" an object from the stack inspector pane
and "paste" it into an arg/temp inspector.

Workaround:

Execute code in the inspector's text pane to save a reference to
"self" in a global variable.

Bug 30983 - GBS object inspector and debugger receiver inspector are missing clipboard copy/paste

Product: GemBuilder for Smalltalk/VW

Versions: 7.6, 7.5, 7.4.1, 7.4, 7.3.3, 7.3.2, 7.3.1, 7.3, 7.2.2, 7.2.1, 7.2, 7.1.2, 7.1.1, 7.1, 7.0.2, 7.0.1, 7.0, 6.2, 6.1, 6.0

Platform VW 7.x

VisualWorks inspectors have a clipboard. You can copy objects to it and paste from it. GBS object inspectors and the GBS debugger receiver inspector are missing this functionality.

Workaround:

Execute code in the inspector's text pane to save a reference to self in a global variable.

Bug 30981 - Debugger step across client forwarder send stops prematurely

Product: GemBuilder for Smalltalk/VW

Versions: 8.5, 8.4, 8.3, 8.2, 8.1, 7.6.1, 7.6, 7.5, 7.4.1, 7.4, 7.3.3, 7.3.2, 7.3.1, 7.3, 7.2.2, 7.2.1, 7.2, 7.1.2, 7.1.1, 7.1, 7.0.2, 7.0.1, 7.0, 6.2, 6.1, 6.0

Platform VW 7.x

With a server context selected, a 'step' should
advance server execution until the selected context is the top
context. If it was already the top context, it should execute as much
code as necessary to advance one step point and return from any
message sends done as part of the step.

However, if a step invokes a client forwarder send execution will stop
early, at the point where the client forwarder execution is about to
return a result from the client to the server.

Workaround:

After the 'short' step, reselect the desired context and
step again. Repeat if necessary until you get where you should be.

Bug 30860 - Global connectors added by applications are lost after gbsImageStartup

Product: GemBuilder for Smalltalk/VW

Versions: 7.6, 7.5, 7.4.1, 7.4, 7.3.3, 7.3.2, 7.3.1, 7.3, 7.2.2, 7.2.1, 7.2, 7.1.2, 7.1.1, 7.1, 7.0.2, 7.0.1, 7.0, 6.2, 6.1, 6.0

Platform VW 7.x

If you add a new global connector, save your image, quit and restart, you will see your new connector in the global connector list until the first time this image logs in to GemStone. At this point, GBSM initializeGlobalConnectors is invoked, which loses all global connectors except the kernel classes / objects.

Workaround:

Avoid creating global connectors - Use session connectors instead.

Bug 30744 - Debugger "Run until Return" not implemented:

Product: GemBuilder for Smalltalk/VW

Versions: 8.5, 8.4, 8.3, 8.2, 8.1, 7.6.1, 7.6, 7.5, 7.4.1, 7.4, 7.3.3, 7.3.2, 7.3.1, 7.3, 7.2.2, 7.2.1, 7.2, 7.1.2, 7.1.1, 7.1, 7.0.2, 7.0.1, 7.0, 6.2, 6.1, 6.0

Platform VW 7.x

With a GS context selected, the "Run until Return" is not implemented.

Workaround:

Explicitly set a breakpoint at the end of the method, and resume execution.

Bug 30725 - PDP Debugger server context spelling correction not implemented

Product: GemBuilder for Smalltalk/VW

Versions: 8.5, 8.4, 8.3, 8.2, 8.1, 7.6.1, 7.6, 7.5, 7.4.1, 7.4, 7.3.3, 7.3.2, 7.3.1, 7.3, 7.2.2, 7.2.1, 7.2, 7.1.2, 7.1.1, 7.1, 7.0.2, 7.0.1, 7.0, 6.2, 6.1, 6.0

Platform VW 7.x

Spelling correction, using the "Correct selector" or "Correct it..." menu options and buttons, is not implemented for Debugger server contexts; the menu option is disabled, and the buttons are not presented.

Workaround:

Use a browser to make corrections in server code.

Bug 30723 - Debugger "return" from server Contexts is not yet implemented

Product: GemBuilder for Smalltalk/VW

Versions: 8.5, 8.4, 8.3, 8.2, 8.1, 7.6.1, 7.6, 7.5, 7.4.1, 7.4, 7.3.3, 7.3.2, 7.3.1, 7.3, 7.2.2, 7.2.1, 7.2, 7.1.2, 7.1.1, 7.1, 7.0.2, 7.0.1, 7.0, 6.2, 6.1, 6.0

Platform VW 7.x

Attempting to use the "return" feature of the VisualWorks 7.x debugger when a server Context is selected results in a walkback.

Bug 30722 - PDP debugger "Jump to carat" not yet implemented.

Product: GemBuilder for Smalltalk/VW

Versions: 8.5, 8.4, 8.3, 8.2, 8.1, 7.6.1, 7.6, 7.5, 7.4.1, 7.4, 7.3.3, 7.3.2, 7.3.1, 7.3, 7.2.2, 7.2.1, 7.2, 7.1.2, 7.1.1, 7.1, 7.0.2, 7.0.1, 7.0, 6.2, 6.1, 6.0

Platform VW 7.x

The PDP debugger function "Jump to carat" has not yet been implemented.

Bug 30701 - GBS PDP debugger: Altering temp on first entry into method gets GbsRtErrArgOutOfRange

Product: GemBuilder for Smalltalk/VW

Versions: 7.6, 7.5, 7.4.1, 7.4, 7.3.3, 7.3.2, 7.3.1, 7.3, 7.2.2, 7.2.1, 7.2, 7.1.2, 7.1.1, 7.1, 7.0.2, 7.0.1, 7.0, 6.2, 6.1, 6.0

Platform VW 7.x

Fixed In: with GS64 3.x

On first entry into a server method such as:

```
 someMethod
```

```
  | a b |
 a := 5.
 b := 6.
```

the step point highlighted initially is the first line, the method name itself. At this point, the temporary inspector pane contains the field names 'a' and 'b'. If you try to select one of these fields, enter a value into the inspector text pane, and accept that value, an error is reported:

```
 GbsRtErrArgOutOfRange - The following argument is too
 large or out of range: 1
```

After the first step in this method, however, assignment of temporaries from the inspector will succeed.

Workaround:

Step at least once in the method before assigning temporaries in the temp inspector pane.

Bug 29774 - If server session timezone is changed, DateTime replication uses old timezone

Product: GemBuilder for Smalltalk/VW

Versions: 8.5, 8.4, 8.3, 8.2, 8.1, 7.6.1, 7.6, 7.5, 7.4.1, 7.4, 7.3.3, 7.3.2, 7.3.1, 7.3, 7.2.2, 7.2.1, 7.2, 7.1.2, 7.1.1, 7.1, 7.0.2, 7.0.1, 7.0, 6.2, 6.1, 6.0, 5.2.3, 5.2.2

Platform All

When a GBS session flushes a Timestamp (in VW) or DateTime (in VA), this has the side affect of initializing the cached value of the server session's TimeZone. This TimeZone is stored in the server instance of DateTime. If that GBS session then changes the session's current TimeZone on the server, then flushes another Timestamp/DateTime, it will have the original TimeZone when it should have the second TimeZone.

Workaround:

Log out after changing the session's timezone.

Bug 28135 - stack corruption returning from ClientForwarder send

Product: GemBuilder for Smalltalk/VW

Versions: 8.3, 8.2, 8.1, 7.6.1, 7.6, 7.5, 7.4.1, 7.4, 7.3.3, 7.3.2, 7.3.1, 7.3, 7.2.2, 7.2.1, 7.2, 7.1.2, 7.1.1, 7.1, 7.0.2, 7.0.1, 7.0, 6.2, 6.1, 6.0, 5.2.3, 5.2.2, 5.2.1

Platform With 32-bit GemStone/S only

Fixed In: n/a

Occasionally, under unknown circumstances, the value returned back into GemStone from a client forwarder send will be a two element array containing the expected result, rather than the result itself. When this happens, the first element of the array is the selector of the client forwarder send, and the second element is the expected result. This usually subsequently results in a "Does not understand" error in GemStone.

Workaround:

In GemStone code, check if the result of a client forwarder send is a two element array, and if so, use the second element as the result.

Bug 26694 - ClassInstVar changes aren't flushed or faulted

Product: GemBuilder for Smalltalk/VW

Versions: 8.5, 8.4, 8.3, 8.2, 8.1, 7.6.1, 7.6, 7.5, 7.4.1, 7.4, 7.3.3, 7.3.2, 7.3.1, 7.3, 7.2.2, 7.2.1, 7.2, 7.1.2, 7.1.1, 7.1, 7.0.2, 7.0.1, 7.0, 6.2, 6.1, 6.0, 5.2.3, 5.2.2, 5.2.1

Platform All

ClassInstVarConnectors do connect properly on the first login, but changes aren't propagated because of issues with marking a class as dirty.

Workaround:

A workaround would be to use a value model like an Association instance in the classInstVar to hold the value that would normally be put in the classInstVar. Code would then access with #value and #value: in both the client and server code to ensure changes are propagated as the association is dirtied.

Bug 15830 - Session Browser may not show correct transaction mode

Product: GemBuilder for Smalltalk/VW

Versions: 8.5, 8.4, 8.3, 8.2, 8.1, 7.6.1, 7.6, 7.5, 7.4.1, 7.4, 7.3.3, 7.3.2, 7.3.1, 7.3, 7.2.2, 7.2.1, 7.2, 7.1.2, 7.1.1, 7.1, 7.0.2, 7.0.1, 7.0, 6.2, 6.1, 6.0, 5.2.3, 5.2.2, 5.2.1, 5.1.x

Platform All

The GemBuilder Session Browser has two radio buttons, labeled "Automatic" and "Manual", for setting and displaying a session's transaction mode. The Session Browser does not automatically update these buttons to reflect changes in a session's transaction mode made programmatically (that is, without using the browser). The display will be refreshed to show the current mode for a session if you deselect and reselect the session's entry in the browser.

Beginning in GemStone/S version 5.1.1, some applications make use of a newly-introduced third transaction mode, known as "transactionless" mode. The Session Browser is not designed to display this state. When the Session Browser is refreshed as described above, it indicates "Manual" for a session in transactionless mode.

Workaround:

You can use a workspace to check on the transaction mode by executing the following using "client print-it":

```
  GBSM currentSession transactionMode
```

Possible responses are #autoBegin, #manualBegin, and #transactionless.

Bug 15436 - Embedded ClientForwarder can't print itself

Product: GemBuilder for Smalltalk/VW

Versions: 8.5, 8.4, 8.3, 8.2, 8.1, 7.6.1, 7.6, 7.5, 7.4.1, 7.4, 7.3.3, 7.3.2, 7.3.1, 7.3, 7.2.2, 7.2.1, 7.2, 7.1.2, 7.1.1, 7.1, 7.0.2, 7.0.1, 7.0, 6.2, 6.1, 6.0, 5.2.3, 5.2.2, 5.2.1, 5.1.x

Platform All

An embedded ClientForwarder can't print itself. Instead, it tries to
replicate OPrintStream, resulting in the exception:

```
'Unhandled exception: Class creation error
- attempt to create a Smalltalk class for GemStone OPrintStream'
```

Workaround:

As a workaround for this bug, file the following method in as SystemUser
and commit it.

```
category: 'printing'
method: ClientForwarder
printOn: aStream

    "Forward #printString instead of #printOn:, to avoid replicating the
     argument, a PrintStream."

    aStream nextPutAll: self printString
%
```

Bug 14551 - GBS debugger step from GS into client forwarder fails

Product: GemBuilder for Smalltalk/VW

Versions: 8.5, 8.4, 8.3, 8.2, 8.1, 7.6.1, 7.6, 7.5, 7.4.1, 7.4, 7.3.3, 7.3.2, 7.3.1, 7.3, 7.2.2, 7.2.1, 7.2, 7.1.2, 7.1.1, 7.1, 7.0.2, 7.0.1, 7.0, 6.2, 6.1, 6.0, 5.2.3, 5.2.2, 5.2.1, 5.1.x

Platform All

When stepping through GemStone Smalltalk execution, the GemBuilder
debugger will be unable to "step into" a client forwarder send.

Workaround:

"Step over" the client forwarder send, or place a "halt"
in client code that will enable you to debug the client forwarder
send.

Bug 10755 - Compiling temporary server class on client and aborting leaves proxy that blocks further compiles

Product: GemBuilder for Smalltalk/VW

Versions: 8.5, 8.4, 8.3, 8.2, 8.1, 7.6.1, 7.6, 7.5, 7.4.1, 7.4, 7.3.3, 7.3.2, 7.3.1, 7.3, 7.2.2, 7.2.1, 7.2, 7.1.2, 7.1.1, 7.1, 7.0.2, 7.0.1, 7.0, 6.2, 6.1, 6.0, 5.2.3, 5.2.2, 5.2.1, 5.1.x

Platform VW

If you replicate a server class that is not committed to the Smalltalk
client using
"Create in GS" or "Compile in GS", this will cache a proxy for the GemStone
class in GBS. If you abort the session, the proxy remains in the GemBuilder
cache, even though the class in GemStone no longer exists.

The existence of the proxy can result in the class never getting recreated
in GemStone in the new transaction; "Create in GS" reports no error, but

the GemStone browser and launcher report no such class exists.

Bug 10754 - Session Browser Fails to Update Session Status

Product: GemBuilder for Smalltalk/VW

Versions: 8.5, 8.4, 8.3, 8.2, 8.1, 7.6.1, 7.6, 7.5, 7.4.1, 7.4, 7.3.3, 7.3.2, 7.3.1, 7.3, 7.2.2, 7.2.1, 7.2, 7.1.2, 7.1.1, 7.1, 7.0.2, 7.0.1, 7.0, 6.2, 6.x, 5.2.3, 5.2.2, 5.2.1, 5.1.x

The Session Browser's Auto and Manual radio buttons are not properly updated when the logged-in session's transaction mode is changed by other means.

Workaround:

Deselect and reselect the session. Its status will be displayed properly until the next change.

Bug 10741 - GemStone Forwarders Do Not Support VisualWorks ValueHolders

Product: GemBuilder for Smalltalk/VW

Versions: 7.5, 7.4.1, 7.4, 7.3.3, 7.3.2, 7.3.1, 7.3, 7.2.2, 7.2.1, 7.2, 7.1.2, 7.1.1, 7.1, 7.0.2, 7.0.1, 7.0, 6.2, 6.1, 6.0, 5.2.3, 5.2.2, 5.2.1, 5.1.x

Platform VW

When a ValueHolder holds a forwarder to a GemStone object, operations performed on the ValueHolder from the VisualWorks screen can successfully create, save, and modify the object in GemStone. However, the GemStone server object does not send a message back the ValueHolder telling it that the object to which it points has changed. As a result, the VisualWorks screen does not redraw itself to properly reflect the GemStone object change.

Bug 10497 - No query to migrate instances if new version defined from class definition template in GemStone Browser

Product: GemBuilder for Smalltalk/VW

Versions: 8.2, 8.1, 7.6.1, 7.6, 7.5, 7.4.1, 7.4, 7.3.3, 7.3.2, 7.3.1, 7.3, 7.2.2, 7.2.1, 7.2, 7.1.2, 7.1.1, 7.1, 7.0.2, 7.0.1, 7.0, 6.2, 6.1, 6.0, 5.2.3, 5.2.2, 5.2.1, 5.1.x

Fixed In: 8.3

If you generate a new version of an existing class by filling in the name of the existing class in the NameOfClass field of the GemStone Browser's class creation template (rather than by selecting a class in the class pane), you do not receive the expected dialog box asking if you wish to migrate instances.

Workaround:

Bring up a Class Version Browser and migrate instances explicitly.

Bug 9551 - GemBuilder fileout/filein of GemStone Smalltalk doesn't initialize classes

Product: GemBuilder for Smalltalk/VW

Versions: 8.5, 8.4, 8.3, 8.2, 8.1, 7.6.1, 7.6, 7.5, 7.4.1, 7.4, 7.3.3, 7.3.2, 7.3.1, 7.3, 7.2.2, 7.2.1, 7.2, 7.1.2, 7.1.1, 7.1, 7.0.2, 7.0.1, 7.0, 6.2, 6.x, 5.2.3, 5.2.2, 5.2.1, 5.1.x

From GemBuilder, when you file out GemStone Smalltalk classes that have

class initialize methods, and then file that fileout back in, the classes
will not get initialized automatically after the filein.

GemBuilder fileout relies on GemStone server code to produce the fileout
text, and GemStone server classes do not by default perform class initialization.

Workaround:

You can implement the class method fileOutPostMethodsOn: aStream on your
GemStone server class that requires initialization, to write the appropriate
initialization to the fileout.

Bug 8474 - GemBuilder filein class creation does not version subclasses

Product: GemBuilder for Smalltalk/VW

Versions: 7.4.1, 7.4, 7.3.3, 7.3.2, 7.3.1, 7.3, 7.2.2, 7.2.1, 7.2, 7.1.2, 7.1.1, 7.1, 7.0.2, 7.0.1, 7.0, 6.2, 6.1, 6.0, 5.2.3, 5.2.2, 5.2.1, 5.1.x

Platform VW

Impact: Informational

If a GemStone Smalltalk filein from GemBuilder does a doIt command that creates a new version of an existing GemStone class, subclasses of that class are not versioned along with it. For example, if I have class A with subclass B, and a filein redefines A to add an instance variable, then class B will still inherit from the old version of A, without the new instance variable.

Workaround:

Use a GemBuilder browser to make changes to a GemStone class definition. If you are filing in, file in all class definitions, in superclass order, to ensure they are recompiled.

- file out that class, and all classes that inherit from it
- make the modification to the filed out file
- file in the the classes in superclass order - that is,
  start with the class that was modified, and then file in
  its subclasses, then their subclasses, and so on

GBS-VA

Bug 48680 - Problem loading shared library via full path on Windows 10

Product: GemBuilder for Smalltalk/VA

Versions: 5.4.5, 5.4.4, 5.4.3

Platform Windows 10

Fixed In: 5.4.6

Loading the server shared library, libgcirpc-N.N.N-32.dll, using libraryName: accepts a full path to the libgcirpc server library, and should be then able to locate the other required server shared libraries in the same directory. On Windows 10, this is not sufficient to allow other libraries to be found. (#48680)

Workaround:

Ensure that the server shared libraries are on the %PATH%

Bug 48144 - Array replication can break hash tables

Product: GemBuilder for Smalltalk/VA

Versions: 5.4.4, 5.4.3, 5.4.2, 5.4.1, 5.4, 5.3.2, 5.3.1, 5.3, 5.2.7, 5.2.6, 5.2.5, 5.2.4, 5.2.3, 5.2.2, 5.2.1, 5.1.x

Fixed In: 5.4.5

When an already replicated Array is resized on the server, the changed size Array is replicated back to the client. In order to increase the size of the client object, replication creates a new instance and switches it for the old instance.

The way this was done failed to preserve the identity hash of the original, which broke lookin within any tables.

Bug 48012 - Dynamic instance variables not displayed

Product: GemBuilder for Smalltalk/VA

Versions: 5.4.6, 5.4.5, 5.4.4, 5.4.3, 5.4.2, 5.4.1, 5.4

Platform with GS64 v3.0 and later

GemStone/S 64 Bit v3.0 and later support dynamic instance variables, which are symbol/value pairs associated with specific instances of any class. GBS inspectors do not display any dynamic instance variables associated with a server object that is being inspected.

Note that dynamic instance variables do not replicate to GBS.

Bug 47870 - Replicating a header-only report for a LargeInteger corrupts the cached copy in the client

Product: GemBuilder for Smalltalk/VA

Versions: 5.4.4, 5.4.3, 5.4.2, 5.4.1, 5.4, 5.3.x, 5.2.7, 5.2.6

Fixed In: 5.4.5, 5.2.7.1

Impact: Critical

When a a header-only report is returned to the client containing a LargeInteger that is already replicated on the client, the LargeInteger may be corrupted such that all bits are 0.

Bug 45983 - Recursive dictionary replication problem under VA Smalltalk 8.6.2

Product: GemBuilder for Smalltalk/VA

Versions: 5.4.6, 5.4.5, 5.4.4, 5.4.3

Platform VA 8.6.2

Replication of an example like:

```
|dict nestedDict |
dict := Dictionary new.
nestedDict := Dictionary new.
nestedDict at: dict put: (Array with: 'dict').
dict at: nestedDict put: (Array with: 'nestedDict').
dict
```

does not work properly. VA Smalltalk 8.6.2 introduced a new hash calculation for Dictionary which is dependent on the size of the dictionary.

During replication, the nested dictionary is empty when it is added as a key inside the outer dictionary. Subsequent addition of the outer dictionary inside the nested dictionary changes its size, resulting in its previously hashed position becoming incorrect.

Workaround:

Customers encountering this issue should adjust their application to send #rehash to such dictionaries following replication.

Bug 45222 - GbsObjErrDoesNotExist if multiple sessions send #asGSObjectInSession: to the same object

Product: GemBuilder for Smalltalk/VA

Versions: 5.4.2, 5.4.1, 5.4, 5.3.2, 5.3.1, 5.3, 5.2.7, 5.2.6, 5.2.5, 5.2.4, 5.2.3, 5.2.2, 5.2.1, 5.1.x

Platform All

Fixed In: 5.4.3

When creating a server object from a client object, if another process is trying to create a server object from the same client object, there is a small chance that two delegates will be created for the client object. The first delegate is immediately thrown away by the second delegate. This can result in GbsObjErrDoesNotExist errors.

Bug 41760 - When proceeding from a debugger, forwarder messages are sent to the client

Product: GemBuilder for Smalltalk/VA

Versions: 5.2.7, 5.2.6, 5.2.5, 5.2.4, 5.2.3, 5.2.2, 5.2.1, 5.2.x

Platform VA

Fixed In: 5.3

When proceeding from within a debugger, messages sent to forwarders sometimes
perform their lookup in the client rather than on the server.

For example, the following code illustrates the problem:

```
|f|
f := GBSM fwevaluate: 'Object new'. GBSM halt. f segment
```

When a debugger comes up, hit the proceed button and you get a
'Message not understood: #segment' error. 'segment' exists on the server
but not on the client.

Workaround:

There are a few alternatives - choose the one that suits your circumstances:

Set forwarderDebugging to true (GBSM forwarderDebugging: true).

Remove or relocate the 'halt' message that invoked the debugger.

Step through the code instead of proceeding through it.

Bug 41759 - GemBuilder filein class creation won't version subclasses or copy methods

Product: GemBuilder for Smalltalk/VA

Versions: 5.4.6, 5.4.5, 5.4.4, 5.4.3, 5.4.2, 5.4.1, 5.4, 5.3.2, 5.3.1, 5.3, 5.2.7, 5.2.6, 5.2.5, 5.2.4, 5.2.3, 5.2.2, 5.2.1, 5.2, 5.1.x

Platform VA

If a GemStone Smalltalk filein from GemBuilder does a doIt command that creates a new version of an existing GemStone class, subclasses of that class are not versioned along with it. For example, if I have class A with subclass B, and a filein redefines A to add an instance variable, then class B will still inherit from the old version of A, without the new instance variable.

Also, a filein that simply redefines the class (to do something like add an instance variable) will not try to copy the methods from the old class into the new class.

These are features that you get if you use a browser to do similar operations. These features might not be as important in a filein, since fileins are usually (but not always) complete descriptions of a class or collection of classes.

Workaround:

Use a GemBuilder browser to make changes to a GemStone class definition. If you must use another interface, you should:

- file out that class, and all classes that inherit from it
- make the modification to the filed out file
- file in the the classes in superclass order - that is,
  start with the class that was modified, and then file in
  its subclasses, then their subclasses, and so on

Bug 41729 - "No client class SomeClass, class generation is disabled" even though SomeClass exists

Product: GemBuilder for Smalltalk/VA

Versions: 5.4.6, 5.4.5, 5.4.4, 5.4.3, 5.4.2, 5.4.1, 5.4, 5.3.2, 5.3.1, 5.3, 5.2.7, 5.2.6, 5.2.5, 5.2.4, 5.2.3, 5.2.2

Platform All

Given a class SomeClass, if the GBS configuration option generateSTClasses is false, and there is no class connector for SomeClass, then attempting to replicate an object that references SomeClass will fail with the error "No client class SomeClass, class generation is disabled", even though that class is already defined in the client as well as the server.

Workaround:

Define a class connector for the class.

Bug 41728 - GemBuilder fileout/filein of GemStone Smalltalk doesn't initialize classes

Product: GemBuilder for Smalltalk/VA

Versions: 5.4.6, 5.4.5, 5.4.4, 5.4.3, 5.4.2, 5.4.1, 5.4, 5.3.2, 5.3.1, 5.3, 5.2.7, 5.2.6, 5.2.5, 5.2.4, 5.2.3, 5.2.2, 5.2.1, 5.1.x

From GemBuilder, when you file out GemStone Smalltalk classes that have class initialize methods, and then file that fileout back in, the classes won't get initialized after the filein. This is because the GemBuilder fileout does not generate a do-it at the end that initializes classes that implement an initialize method.

Workaround:

Figure out which classes need to be sent #initialize, and do it.

Bug 41727 - No query to migrate instances if new version defined from class definition template in GemStone Browser

Product: GemBuilder for Smalltalk/VA

Versions: 5.4, 5.3.2, 5.3.1, 5.3, 5.2.7, 5.2.6, 5.2.5, 5.2.4, 5.2.3, 5.2.2, 5.2.1, 5.1.x

Fixed In: 5.4.1

If you generate a new version of an existing class by filling in the name of the existing class in the NameOfClass field of the GemStone Browser's class creation template (rather than by selecting a class in the class pane), you do not receive the expected dialog box asking if you wish to migrate instances.

Workaround:

Bring up a Class Version Browser and migrate instances explicitly.

Bug 41726 - Session Browser Fails to Update Session Status

Product: GemBuilder for Smalltalk/VA

Versions: 5.4.5, 5.4.4, 5.4.3, 5.4.2, 5.4.1, 5.4, 5.3.2, 5.3.1, 5.3, 5.2.7, 5.2.6, 5.2.5, 5.2.4, 5.2.3, 5.2.2, 5.2.1, 5.1.x

Fixed In: 5.4.6

The Session Browser's Auto and Manual radio buttons are not properly updated when the logged-in session's transaction mode is changed by other means.

Workaround:

Deselect and reselect the session. Its status will be displayed properly until the next change.

Bug 41725 - GBS debugger step from GS into client forwarder fails

Product: GemBuilder for Smalltalk/VA

Versions: 5.4.6, 5.4.5, 5.4.4, 5.4.3, 5.4.2, 5.4.1, 5.4, 5.3.2, 5.3.1, 5.3, 5.2.7, 5.2.6, 5.2.5, 5.2.4, 5.2.3, 5.2.2, 5.2.1, 5.1.x

Platform All

When stepping through GemStone Smalltalk execution, the GemBuilder debugger will be unable to "step into" a client forwarder send.

Workaround:

"Step over" the client forwarder send, or place a "halt" in client code that will enable you to debug the client forwarder send.

Bug 41724 - Embedded ClientForwarder can't print itself

Product: GemBuilder for Smalltalk/VA

Versions: 5.4.6, 5.4.5, 5.4.4, 5.4.3, 5.4.2, 5.4.1, 5.4, 5.3.2, 5.3.1, 5.3, 5.2.7, 5.2.6, 5.2.5, 5.2.4, 5.2.3, 5.2.2, 5.2.1, 5.1.x

Platform All

An embedded ClientForwarder can't print itself. Instead, it tries to replicate OPrintStream, resulting in the exception:

'Unhandled exception: Class creation error
- attempt to create a Smalltalk class for GemStone OPrintStream'

Workaround:

As a workaround for this bug, file the following method in as SystemUser and commit it.

```
category: 'printing'
method: ClientForwarder
printOn: aStream

       "Forward #printString instead of #printOn:, to avoid replicating
the
        argument, a PrintStream."

       aStream nextPutAll: self printString
%
```

Bug 41723 - Session Browser may not show correct transaction mode

Product: GemBuilder for Smalltalk/VA

Versions: 5.4.5, 5.4.4, 5.4.3, 5.4.2, 5.4.1, 5.4, 5.3.2, 5.3.1, 5.3, 5.2.7, 5.2.6, 5.2.5, 5.2.4, 5.2.3, 5.2.2, 5.2.1, 5.1.x

Platform All

Fixed In: 5.4.6

The GemBuilder Session Browser has two radio buttons, labeled "Automatic" and "Manual", for setting and displaying a session's transaction mode. The Session Browser does not automatically update these buttons to reflect changes in a session's transaction mode made programmatically (that is, without using the browser). The display will be refreshed to show the current mode for a session if you deselect and reselect the session's entry in the browser.

Beginning in GemStone/S version 5.1.1, some applications make use of a newly-introduced third transaction mode, known as "transactionless" mode. The Session Browser is not designed to display this state. When the Session Browser is refreshed as described above, it indicates "Manual" for a session in transactionless mode.

Workaround:

You can use a workspace to check on the transaction mode by executing the following using "client print-it":

```
  GBSM currentSession transactionMode
```

Possible responses are #autoBegin, #manualBegin, and #transactionless.

Bug 41722 - "Create in GS" creates twice if class connector already present

Product: GemBuilder for Smalltalk/VA

Versions: 5.4, 5.3.2, 5.3.1, 5.3, 5.2.7, 5.2.6, 5.2.5, 5.2.4, 5.2.3, 5.2.2, 5.2.1, 5.1.x

Platform All

Fixed In: 5.4.1

Performing a "Create in GS" operation on a client Smalltalk class that does not already exist in GemStone can create two versions of the class in GemStone if an unconnected Class Connector for that class already exists at the time the operation is performed.

Such an unconnected Class Connector can exist, for example, as an artifact of a sequence of events similar to the following:

1. Before logging into GemStone, the application creates a class connector for a class that exists in the client but not in GemStone.

2. On GemStone login, the connector will fail to connect.

3. User chooses not to remove failed connectors.

Workaround:

Before performing "Create in GS", delete any preexisting Class Connector for the Smalltalk class.

Bug 41721 - ClassInstVar changes aren't flushed or faulted

Product: GemBuilder for Smalltalk/VA

Versions: 5.4.6, 5.4.5, 5.4.4, 5.4.3, 5.4.2, 5.4.1, 5.4, 5.3.2, 5.3.1, 5.3, 5.2.7, 5.2.6, 5.2.5, 5.2.4, 5.2.3, 5.2.2, 5.2.1

Platform All

ClassInstVarConnectors do connect properly on the first login, but changes aren't propagated because of issues with marking a class as dirty.

Workaround:

A workaround would be to use a value model like an Association instance in the classInstVar to hold the value that would normally be put in the classInstVar. Code would then access with #value and #value: in both the client and server code to ensure changes are propagated as the association is dirtied.

Bug 41720 - Stack corruption returning from ClientForwarder send

Product: GemBuilder for Smalltalk/VA

Versions: 5.2.7, 5.2.6, 5.2.5, 5.2.4, 5.2.3, 5.2.2, 5.2.1

Platform With 32-bit GemStone/S only

Occasionally, under unknown circumstances, the value returned back into GemStone from a client forwarder send will be a two element array containing the expected result, rather than the result itself. When this happens, the first element of the array is the selector of the client forwarder send, and the second element is the expected result. This usually subsequently results in a "Does not understand" error in GemStone.

Workaround:

In GemStone code, check if the result of a client forwarder send is a two element array, and if so, use the second element as the result.

Bug 41719 - If server session timezone is changed, DateTime replication uses old timezone

Product: GemBuilder for Smalltalk/VA

Versions: 5.4.6, 5.4.5, 5.4.4, 5.4.3, 5.4.2, 5.4.1, 5.4, 5.3.2, 5.3.1, 5.3, 5.2.7, 5.2.6, 5.2.5, 5.2.4, 5.2.3, 5.2.2

Platform All

When a GBS session flushes a Timestamp (in VW) or DateTime (in VA), this has the side affect of initializing the cached value of the server session's TimeZone. This TimeZone is stored in the server instance of DateTime.
If that GBS session then changes the session's current TimeZone on the server, then flushes another Timestamp/DateTime, it will have the original TimeZone when it should have the second TimeZone.

Workaround:

Log out after changing the session's timezone.

Bug 41718 - Possible to lose modifications through concurrent operations

Product: GemBuilder for Smalltalk/VA

Versions: 5.4.6, 5.4.5, 5.4.4, 5.4.3, 5.4.2, 5.4.1, 5.4, 5.3.2, 5.3.1, 5.3, 5.2.7, 5.2.6, 5.2.5, 5.2.4, 5.2.3, 5.2.2, 5.2.1

Platform All

Impact: Critical

If multiple Smalltalk Processes concurrently manipulate the same replicate, it is possible for a change to that replicate to be lost. For this to happen, specific circumstances must be occur in a quite narrow time window.

Specific circumstances:

In most versions of GBS, if one Process modifies a replicate while another process is making the replicate into a stub, the modification can be lost. Replicates become stubs due to either an explicit request or as the result of server execution if the object has changed on the server and its fault policy is #lazy.

In a few versions of GBS, if one Process modifies a replicate while another Process is initiating server execution, the modification can be lost.

When using the GBS automatic dirty-marking feature in VA Smalltalk and VisualWorks 7.x, it is possible for GBS to make these failures impossible.
In VisualWorks 5i, making automatic dirty-marking immune to these failures would slow execution down, so we do not anticipate ever fixing this problem in VisualWorks 5i.

All applications which manually mark objects dirty are vulnerable to this problem.

We recommend that all customers use VisualWorks 7.x or VA Smalltalk, and that that their applications use automatic dirty-marking. If you must use VisualWorks 5i or manual dirty-marking, the application must be very careful when accessing sessions and their replicates from more than one Smalltalk Process.

Workaround:

No workaround

Bug 41717 - Invalid class connectors can make session unusable

Product: GemBuilder for Smalltalk/VA

Versions: 5.4, 5.3.2, 5.3.1, 5.3, 5.2.7, 5.2.6, 5.2.5

Platform All

Fixed In: 5.4.1

nvalid class connectors can cause login failures. Upon login, if GemBuilder finds a class connector for which it cannot successfully connect, the user is presented with a dialog asking the user if he or she would like to remove the invalid connector. If the user chooses to say 'No', subsequent openings of the GemStone classes browser will attempt to use that invalid connector and cause errors. It is possible that even if the user chooses 'Yes', errors could occur if the the invalid connector was encountered inside of a flush to the server.

Workaround:

If you encounter a failed connector, select 'Yes' to remove the connector, then logout and log back in. If you wish to keep that connector, resolve the failure before opening the class browser.

Bug 41716 - Debugger shows incorrect variables in ExecutableBlock methods

Product: GemBuilder for Smalltalk/VA

Versions: 5.2.7, 5.2.6, 5.2.5

Platform When running with 32-bit servers

The context inspector in the debugger will show the incorrect contextual variables if the selected stack frame is a method on ExecutableBlock. Instead of showing the values of the variables within the block, it will show the variables from the previous stack frame, or outer context. The receiver inspector will show self as the ComplexBlock instead of the outer context of the block.

Bug 40321 - Debugger shows fowarder information rather than server object

Product: GemBuilder for Smalltalk/VA

Versions: 5.4.2, 5.4.1, 5.4, 5.3.2, 5.3.1, 5.3, 5.2.7, 5.2.6

Platform VA

Fixed In: 5.4.3

When debugging in a server context, selecting a server object displays

forwarder information, rather than the object itself. Double clicking
on the variable will brings up an inspector that displays the object.

After image save and restart, the default debugger is reset by VisualAge.
To verify, execute:
  EsImageStartUp debuggerClass

If it returns StsDebugger, or anything other than GbxDebugger, debugging
GBS/GemStone code may be subject to incorrect behavior due to this bug.

Workaround:

Execute:
 GBSM enableGbsDebugger

After starting the image

Bug 35422 - Class Instance Variable Check On Create/Compile in ST Raises Error

Product: GemBuilder for Smalltalk/VA

Versions: 5.4.5, 5.4.4, 5.4.3, 5.4.2, 5.4.1, 5.4, 5.3.2, 5.3.1, 5.3, 5.2.7, 5.2.6, 5.2.5, 5.2.4

Fixed In: 5.4.6

If you try to do a Create in ST or Compile in ST for a GemStone class that has a capitalized class instance variable, you will get a dialog asking you if you want to add the capitalized class instance variable or not. If you answer 'No', an error will be raised, saying that 'A non-class object was generated'.

Bug 30902 - Stack growth due to GemStone exceptions handled and resumed in client

Product: GemBuilder for Smalltalk/VA

Versions: 5.4.6, 5.4.5, 5.4.4, 5.4.3, 5.4.1, 5.4, 5.3.2, 5.3.1, 5.3, 5.2.7, 5.2.6, 5.2.5, 5.2.4, 5.2.3, 5.2.2, 5.2.1, 5.2

Platform All

Sequential GemStone exceptions that are handled on the client may cause stack growth and consequent memory use. This is due to the way GBS processes a GbsError, invokes handlers, and then does a continue; subsequent errors will be raised on the stack and processed similarly, and so on, causing the stack to grow with each subsequent error.

Bug 28047 - "Invalid operation during callback" errors from VisualAge

Product: GemBuilder for Smalltalk/VA

Versions: 5.4.6, 5.4.5, 5.4.4, 5.4.3, 5.4.2, 5.4.1, 5.4, 5.3.2, 5.3.1, 5.3, 5.2.7, 5.2.6, 5.2.5, 5.2.4, 5.2.3, 5.2.2, 5.2.1, 5.2, 5.1.4, 5.1.3sp1

Platform VA

A design limitation in VisualAge causes Semaphore>>wait to fail
with the error:

 'Invalid operation during callback'

if the operation occurs during a UI callback, and the semaphore
must wait.

GBS relies on semaphores for normal operation. If a UI callback
must unstub objects, or invoke GemStone code, and another thread

is currently accessing that GbsSession, this error may occur.

Workaround:

The simplest way to work around this error is to fork execution at some point so that by the time GBS has to go through a semaphore, it is no longer running in the UI process. Examine the stack traces of occurrences of this error and determine an appropriate place to fork. This can often be achieved by wrapping the entire contents of such a method with:

```
 [
   <contents of method>
 ] fork
```

Often, this can be isolated to a small number of places in the application.

There are a couple restriction on this workaround - the senders of this method cannot be relying on the return value from this method. In many cases, the sender is the VisualAge method CwCallbackRec>>callWith:callData:, which doesn't use the return value of the application's callback method, so it works (assuming there aren't other senders in the application that do rely on the return value).

Another restriction is there can't be any returns (^) at all from inside the forked block. So if the method contains a control flow construct such as:

```
 x == y ifTrue: [
   <do one thing>
   ^self ].
 <do another thing>
```

... it needs to be recoded to avoid the return statement, like:

```
 x == y
   ifTrue: [
     <do one thing> ]
   ifFalse: [
     <do another thing> ]
```

One potential problem with this workaround of simply forking, though, is the possibility of introducing multiple threads into the application where you might not want it. When the block is forked off, the UI will immediately become usable again, even though the block may still be executing. Ask yourself the following questions: Is it possible that another UI request will be issued before that block finishes executing? If so, will that be OK?

If that's not OK, then the next part of the workaround is to introduce a work queue. Rather than simply saying "[<work>] fork", instead the workaround will be of the form "someQueue submit: [<work>]", and the queue will make sure that blocks submitted to it are executed one at a time.

Comments:

Bug 25565 - Defunct stub errors occur under VA

Product: GemBuilder for Smalltalk/VA

Versions: 5.4.6, 5.4.5, 5.4.4, 5.4.3, 5.4.2, 5.4.1, 5.4, 5.3.2, 5.3.1, 5.3, 5.2.7, 5.2.6, 5.2.5, 5.2.4, 5.2.3, 5.2.2, 5.2.1, 5.2, 5.1.4

Platform VA

Impact: Critical

Under VisualAge Smalltalk GBS applications, generally in runtime applications under heavy use, it's possible to get unfounded defunct stub errors. These are intermittant timing related problems, relating to the combination of the way VA handles weak dictionaries, signal handling, and other factors.

The occurrence of these errors has been greatly reduced in GBS versions 5.2 and later.

Bug 16711 - Use of VA debugger "stack" -> "next stack display range" results in walkback

Product: GemBuilder for Smalltalk/VA

Versions: 5.4.6, 5.4.5, 5.4.4, 5.4.3, 5.4.2, 5.4.1, 5.4, 5.3.2, 5.3.1, 5.3, 5.2.7, 5.2.6, 5.2.5, 5.2.4, 5.2.3, 5.2.2, 5.2.1, 5.2, 5.1.4, 5.1.3, 5.1.2

Platform VA

When using a VA debugger on a GemStone stack overflow, with only a partial stack showing, using "stack" -> "next stack display range" can result in a walkback. This is only likely to happen when you are debugging a GemStone infinite recursion bug.

Workaround:

Increase the number of stack frames that the debugger displays in each range by sending the following message:

GbxDebugSession defaultStackSizeLimit: 500 "Or something sufficiently large"

Bug 16037 - Infinite Loop in "Show User Info" with insufficient authorization

Product: GemBuilder for Smalltalk/VA

Versions: 5.2.7, 5.2.6, 5.2.5, 5.2.4, 5.2.3, 5.2.2, 5.2.1, 5.2, 5.1.4, 5.1.3, 5.0

Platform VA

Fixed In: 5.4.x, 5.3.x

If you view user information using the menu pick sequence Admin Tools -> Browse GemStone Users -> Show User Info from VisualAge System Transcript, but lack sufficient privileges to view the selected user, the system issues a notifier informing you that "You do not have authority to view" the selected user.

After selecting the OK button to dismiss the notifier, the notifier keeps returning.

Workaround:

Break out of the loop by first selecting the VA "User Break" button and then selecting the OK button in the authorization error window.

Bug 15253 - Symbol List Browser: Update Error When Selected Symbol is Obsolete

Product: GemBuilder for Smalltalk/VA

Versions: 5.4.2, 5.4.1, 5.4, 5.3.2, 5.3.1, 5.3, 5.2.7, 5.2.6, 5.2.5, 5.2.4, 5.2.3, 5.2.2, 5.2.1, 5.1.x

Platform All

Fixed In: 5.4.3

A GemStone Symbol List Browser continues displaying a symbol that has been removed in another tool (programmatically or through another Symbol List Browser) until the Symbol List Browser is updated. If the now-removed symbol is selected when the browser is updated, a notifier results:

No method was found for the selector #value when sent to nil with arguments contained in anArray().

Workaround:

Deselect the removed symbol before updating the Symbol List Browser.

Bug 14652 - GBS/VisualAge GemStone Bag Inspector does not list elements

Product: GemBuilder for Smalltalk/VA

Versions: 5.4.6, 5.4.5, 5.4.4, 5.4.3, 5.4.2, 5.4.1, 5.4, 5.3.2, 5.3.1, 5.3, 5.2.7, 5.2.6, 5.2.5, 5.2.4, 5.2.3, 5.2.2, 5.2.1, 5.2, 5.1.4, 5.1.3, 5.0

Platform VA

Inspecting a GemStone Bag on VisualAge produces a GemStone Bag Inspector that does not list elements contained within the Bag.

Workaround:

Inspectors on IdentityBag or IdentitySet do show the elements. One way to invoke a useful inspector is to send the bag the message "asIdentityBag" and inspect the result.

Another way to inspect elements of a Bag is to open an inspector on the Bag, then inspect its "dict" instance variable and execute the message "self keys". This produces an IdentitySet Inspector that allows you to inspect elements of the Bag.

Bug 11138 - Improper authorizations prevent Symbol List Browser opening

Product: GemBuilder for Smalltalk/VA

Versions: 5.4.6, 5.4.5, 5.4.4, 5.4.3, 5.4.2, 5.4.1, 5.4, 5.3.2, 5.3.1, 5.3, 5.2.7, 5.2.6, 5.2.5, 5.2.4

If a SymbolDictionary is in a segment to which the user does not have read authority, opening the Symbol List browser gives "GbsAuthErrSegRead - An attempt was made to read the object with ID XXXX in segment..." error.

Workaround:

If you know which segment the offending SymbolDictionary is in and you can change the segment authorization, change the authorization and commit the transaction. Example GemStone code:

```
MySegment ownerAuthorization: #write.
System commitTransaction.
```

VSD

Bug 49259 - Live monitoring in GS64 3.5.4 does not display data

Product: VSD

Versions: 5.5

Fixed In: 5.5.1

The VSD Main menu item monitor allows you to view the statistics as they are generated by a live system, updating each time new statistics are written. Due to changes in 3.5.4, live-monitoring write the statmonitor data file to disk, but does not load and display anything in the vsd main windows.

Workaround:

You can explicitly load the statmonitor data file and set auto-update.

The monitor dialog displays the path and name of the statmonitor file being updated. Load this file using the Main window Main > Load Data File... menu item, then click on the Main window File > Auto Update menu item to enable auto update. The statistics will be displayed as they are with live monitoring.

Bug 48393 - Loading a truncated, zipped statistics file may SEGV

Product: VSD

Versions: 5.4, 5.3.1, 5.3, 5.2, 5.1.3, 5.1.2, 5.1.1, 5.1, 5.0.1, 5.0, 4.0.2, 4.0.1, 4.0, 3.5, 3.2

Fixed In: 5.4.2

When a gzipped statistics file is truncated, the VSD loading code may SEGV at the point of the corruption (at the end of the file).

Workaround:

Later versions of VSD will load the intact portion of the file, reporting the error to the console.

Bug 48316 - VSD command line cannot load files containing ’-h’ or other arguments

Product: VSD

Versions: 5.4, 5.3.1, 5.3, 5.2, 5.1.3, 5.1.2, 5.1.1, 5.1, 5.0.1, 5.0, 4.0.2, 4.0.1, 4.0, 3.5, 3.2

Fixed In: 5.4.2, 5.4.1

If a filename passed in on the vsd command line contains the character sequence ’-h’, ’-H’, ’-v’, or ’-V’, for example:

```
unix> vsd statmon-h2.out.gz
```

it is interpreted as the command line argument for the help information or version information, and VSD does not start.

Workaround:

Rename the statmonitor file.

Bug 46787 - Negative time offset on Windows caused VSD to crash

Product: VSD

Versions: 5.3.1, 5.3

Fixed In: 5.4

VSD has the option to set a time offset, so that viewing statistics that were generated in a different time zone can be adjusted to match log files. On windows, setting this offset to a negative value caused VSD to crash.

Bug 46560 - Stone name from 32-bit GemStone/S appears truncated to 15 characters

Product: VSD

Versions: 5.5.2, 5.5.1, 5.5, 5.4.2, 5.4, 5.3.1, 5.3, 5.2, 5.1.3, 5.1.2, 5.1.1, 5.1, 4.0.1, 4.0, 3.2, earlier versions

Platform with 32-bit GSS data

Impact: Informational

Due to a limitation in statmonitor on GemStone/S 32 Bit, only the first 15 characters of the the stone name are recorded in the statmonitor data files. As a result, the Stone name, if longer than 15 characters, appears to be truncated in VSD.

Workaround:

Fixed in GS/S v6.7.1

Bug 46084 - Large unsigned stats with value >= 2^31 wrap-around to negative on Windows

Product: VSD

Versions: 5.1.1, 5.1, 5.0.1, 5.0, 4.0.2, 4.0.1, 4.0, 3.5, 3.2, earlier versions

Platform Windows only

Fixed In: 5.1.2

On Windows only, unsigned statistics are handled in 32-bit fields, and when the value is >= 2^31 (2147483648) the high-order bit is treated as a sign bit and the graph will wrap-around and the value be displayed as negative.

Bug 46078 - Chart values may be incorrect on big-endian platforms

Product: VSD

Versions: 5.1.1, 5.1, 5.0.1, 5.0, 4.0.2, 4.0.1, 4.0, 3.5

Platform Big endian platforms: Solaris/SPARC and AIX

Fixed In: 5.1.2

Impact: Critical

On big-endian platforms, some data points for some statistics display incorrect values.

The displayed values are not radically different from the correct values (e.g., a value that should be 0 may be 4 or 20), and may not be noticable.

VSD uses a union data structure to store values, to allow reading the different statitic data types. There are places in which the 64 bit member was stored and the 32-member was read from, which produces incorrect values on big-endian platforms. Small-endian platforms: Linux, Windows, Mac, and Solaris/x86, are not affected.

Bug 45850 - command line -u option does not work on Windows

Product: VSD

Versions: 5.1.3, 5.1.2, 5.1.1, 5.1, 5.0.1, 5.0, 4.0.2, 4.0.1, 4.0, 3.5, 3.2

Platform Windows only

The -u command line option for VSD enables auto-update. Using this option on the vsd command line, on windows, results in an error.

Workaround:

After VSD opens, use the File menu item Auto Update

Bug 45843 - Files may cease to be visible in file dialog on Unix

Product: VSD

Versions: 5.5.2, 5.5.1, 5.5, 5.4.2, 5.4, 5.3.1, 5.3, 5.2, 5.1.2, 5.1.1, 5.1

Impact: Informational

The VSD file open dialog on UNIX allows you to type in the names of files you would like to find. You can enter directory, file, and * wild card characters. These act as an additional filter on the visible files in the main pane.

If you navigate to another directory, the File names pane is cleared, but the filter remains in effect. This may cause existing files to be filtered out of the view.

This is a TCL issue, related to the TCL file dialog introduced in v5.1 to provide new file opening features.

Workaround:

Close the dialog and reopen. VSD will remember the directory you were viewing and start off in that location.

Bug 45728 - With subsecond sampling, chart timestamps incorrect

Product: VSD

Versions: 5.5.2, 5.5.1, 5.5, 5.4.2, 5.4, 5.3.1, 5.3, 5.2, 5.1.2, 5.1.1, 5.1, 5.0.1, 5.0, 4.0.2, 4.0.1, 4.0, 3.5, 3.2, earlier versions

Platform With server data from versions earlier 3.3

Impact: Informational

When statmonitor uses sample sizes that are less than one second, the timestamps are written to the data files as if the samples were taken at 1 second intervals. This is noted when you execute statmonitor, for example:

```
WARNING: Sample Interval < 1 second
  vsd time base in sample points, not units of real time
```

VSD has no way to determine that these timestamps are not one second intervals. When using subsecond samples and analyzing the resulting data, you must remember the sampling and adjust the time offsets manually.

Workaround:

With GemStone/S 64 Bit 3.3, subsecond timestamps are written to VSD, and VSD versions 5.0 and later will produce accurate timestamps.

Bug 45646 - Changing between Elapsed time and Absolute time in Chart may appear to fail

Product: VSD

Versions: 5.5.2, 5.5.1, 5.5, 5.4.2, 5.4, 5.3.1, 5.3, 5.2, 5.1.2, 5.1.1, 5.1, 5.0.1, 5.0, 4.0.2, 4.0.1, 4.0, 3.5, 3.2, Earlier versions

Platform All

Impact: Informational

When you are viewing a Chart, the Time format menu allows you to change between Elapsed Time in Seconds, or Seconds (depending on the version) and Hour:Minute:Second format.

This time format controls the way the time is displayed, but does not change between relative and absolute offset computations. The time format is subject to the Main Window general setting for "Absolute Timestamps".

With the main Window menu item Main > Absolute Timestamps checked, the meanings are:
  Absolute time in seconds (since an arbitrary point in the past)
  Absolute time in Hour:Minute:Second:

With the main windows menu item Main > Absolute Timestamps not checked (in other words, relative timestamps), the meanings are:
  Elapsed time in seconds since start of this statmonitor file
  Elapsed time in seconds since start of this statmonitor file in Hours:Minutes:Seconds

To switch between relative and absolute timestamps, you must go to the Main Window to toggle Absolute Timestamps, and open a new Chart window.

Bug 44975 - Append file does not work correctly with .gfs files from GBS

Product: VSD

Versions: 5.5.2, 5.5.1, 5.5, 5.4.2, 5.4, 5.3.1, 5.3, 5.2, 5.1.2, 5.1.1, 5.1, 5.0.1, 5.0, 4.0.2, 4.0.1, 4.0, 3.5, 3.2

Platform All

The VSD option File > Append Data File allows you to merge a file into an already loaded file, allowing the two files to be viewed as one large continuous set of statistics.

This doesn't work with the .gfs file format as generated by GBS. Append of another file, even one from the same system being monitored, results in two separate sets of entries.

Bug 44108 - VSD errors if incomplete data lines are read

Product: VSD

Versions: 5.5.2, 5.5.1, 5.5, 5.4.2, 5.4, 5.3.1, 5.3, 5.2, 5.1.3, 5.1.2, 5.1.1, 5.1, 5.0.1, 5.0, 4.0.2, 4.0.1, 4.0, 3.5, 3.2, Earlier versions

When live monitoring, VSD reads data from a disk file as the data is being written. In some configurations, it is possible for VSD to read a data line that is not yet completely written, resulting in a partial line read. VSD considers such a partial line to indicate a corrupt data file.

Bug 41951 - Appending data files does not work correctly with open chart windows

Product: VSD

Versions: 5.5, 5.4.2, 5.4, 5.3.1, 5.3, 5.2, 5.1.3, 5.1.2, 5.1.1, 5.1, 5.0.1, 5.0, 4.0.2, 4.0.1, 4.0, 3.5, 3.2

Fixed In: 5.5.1

If you have a chart window open on a particular statistic from a file, and you append another data file to this file, the existing chart windows is not updated, and you cannot view the combined statistics in a new chart window - the statistics are incorrect. You must close all chart windows that involve the original statistics before valid statistics lines can be viewed.

The exception is when the second, appended file is being actively updated and auto update is on. In this case, open charts are updated with new data.

Workaround:

Close all open Chart windows before appending a data file.

Bug 41903 - Column sort can change file display

Product: VSD

Versions: 5.1.2, 5.1.1, 5.1, 5.0.1, 5.0, 4.0.2, 4.0.1, 4.0, 3.5, 3.2, earlier versions

Fixed In: 5.2

When multiple files are loaded, certain sequences of sorting processes by clicking on column headers also changed which file or files's processes were included in the view.

GemStone/S 64 Bit

Bug 49769 - Crash with thread-safe GCI, if C data lost due to FFI structures faulting out

Product: GemStone/S 64 Bit

Versions: 3.6.2, 3.6.1, 3.6, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5

It is possible for CByteArrays and CPointers that are used by FFI to be committed and faulted out, and lose C data. In this case, GsTsExternalSession >> send:to:withArguments: may pass NULL to GciTsPerform for the last argument; this results in a null pointer exception and a HostCallDebugger.

Workaround:

Applications using GsTsExternalSession should install this patch to avoid hitting this bug.

```
method: GsTsExternalSession
_signalError: aString
  | err |
  (err := GciError new) error: gciErr in: self details: nil .
  (err originalNumber between: 4000 and: 4999) ifTrue:[
    self _closeConnection . "ensure connection is closed after fatal error"
  ] ifFalse:[
    gciErr := GciErrSType new .
  ].
  err signal .
%
```

Bug 49741 - Strings with µ/181 and ÿ/255 do not convert to uppercase correctly

Product: GemStone/S 64 Bit

Versions: 3.6.2, 3.6.1, 3.6, 3.5.7, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.5, 3.3.4, 3.3.3, 3.3.2, 3.3.1, 3.3, earlier versions

Fixed In: 3.6.3

The characters µ/code point 181 and ÿ/code point 255 have uppercase forms that are outside of the range of a Single-byte String. Sending asUppercase to a single-byte String containing either of these characters did not transparently convert to a DoubleByteString, and results in a single-byte String containing the wrong upper case form for either of these characters.

Workaround:

Unicode strings, specifically Unicode16, do not have this issue.

Sending asUppercase to each Character will result in the correct uppercase form.

Bug 49737 - Crash on startup for valid LARGE_MEMORY_PAGE configuration

Product: GemStone/S 64 Bit

Versions: 3.6.1, 3.6, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5

Fixed In: 3.6.2

With the settings:
SHR_PAGE_CACHE_LARGE_MEMORY_PAGE_SIZE_MB=0;
SHR_PAGE_CACHE_LARGE_MEMORY_PAGE_POLICY=2;

On stone startup, the SPC Monitor crashes with a SIGFPE (divide by zero); stone startup will appear to hang.

SHR_PAGE_CACHE_LARGE_MEMORY_PAGE_SIZE_MB = 0 specifies to use the default for your system, so this is a valid setting.

Bug 49715 - Risk of Gem writing an invalid tranlog record

Product: GemStone/S 64 Bit

Versions: 3.6.3, 3.6.2, 3.6.1, 3.6, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5

Fixed In: 3.6.2

It is possible for a Gem to create a BEGIN_DATA tranlog record, with the tranlog record kind field incorrectly set to type 0.

The resulting tranlog containing this invalid record, will error when restored with

```
   readRecord: unhandled recordKind 0
```

Workaround:

The cause of the incorrect tranlog record kind field is unknown.

In versions with the fix, if the Gem passes an invalid tranlog records to the Stone, the Stone will terminate the Gem and not write the tranlog record, to ensure the tranlogs are not corrupted.

Bug 49712 - Gem crash with nested blocks containing Array constructors

Product: GemStone/S 64 Bit

Versions: 3.6.1, 3.6, 3.5.7, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.5, 3.3.4, 3.3.3, 3.3.2, 3.3.1, 3.3, earlier versions

Fixed In: 3.6.2

When Array Constructors { } are used within several levels of block testing/conditional/iteration, it can result in a Gem crash, due to a bug in native code generation.

In most versions, this results in an endless loop in

```
  INVALIDcodePtr
```

and the Gem must be killed using kill -9.

Workaround:

Workaround is to configure the gems and topaz -l with

```
  GEM_NATIVE_CODE_ENABLED=0;
```

The problem does not occur when creating Arrays using (Array with: obj), rather than { obj }.

Bug 49686 - back-version gslist -v results in error messages in log files

Product: GemStone/S 64 Bit

Versions: 3.6.3, 3.6.2

Fixed In: n/a

Impact: Informational

When running an older version of gslist with the -v argument, gslist attempts to connect to server processes to verify they are responsive. gslist -v (without a specific process name) will attempt to connect to all GemStone processes running on that host machine, regardless of the version.

When using a much older version of gslist, the -v connection attempt fails against the latest server version processes. This results in errors in the log file of the form:

```
NetAcceptReadClientIdent failed, peer localhost, reason numread 0 < length 114, timeout
```

There is no problem for the server, other than some clutter in the log file.

Workaround:

You can ignore the error messages. Otherwise, avoid using old version gslist with the -v argument on a host machine that is also running a system on the most recent GemStone versions. You can verify a specific process, rather than all processes on the host machine, using gslist -v stonename.

Bug 49683 - Copydbf fails on decryption of encrypted extent that has been pregrown or restored

Product: GemStone/S 64 Bit

Versions: 3.6.1, 3.6

Fixed In: 3.6.2

When an encrypted extent is pregrown during stone startup, or a backup is restored into an encrypted extent and the extent grows, the unused space is initialized to zeros. If this encrypted extent is then decrypted, these pages of zeros are incorrect identified as garbage; and the copydbf operation that is performing the decryption crashes. In this case, the extents are correct; the error is in copydbf.

Workaround:

A backup and restore into a non-encrypted repository can achieve the end result, as a work around, until upgrade to 3.6.2 or later.

Bug 49665 - System class>>_add:to: errors due to incorrect method name

Product: GemStone/S 64 Bit

Versions: 3.6.1, 3.6, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2

Fixed In: 3.6.2

System class>>_add:to: is a hidden set method that has been retained in the image to support legacy applications; it has been replaced by GsBitmap.

When this method was replaced, the incorrect method name was used; invoking this method errors with a message not understood.

Workaround:

As SystemUser, update the method as follows

```
_add: anObject to: hiddenSetSpecifier

"Will be deprecated, new code should use GsBitmaps"

^ self add: anObject toHiddenSet: hiddenSetSpecifier
```

Bug 49643 - Stone may not be completely shutdown when stopstone returns

Product: GemStone/S 64 Bit

Versions: 3.6.3, 3.6.2, 3.6.1, 3.6, 3.5.7, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.5, 3.3.4, 3.3.3, 3.3.2, 3.3.1, 3.3, earlier versions

Impact: Informational

For a system with a large shared page cache, it is possible for stopstone to return, before the SPC monitor has freed up the shared memory region. This can result in errors if an immediate startstone is done, e.g. from a script.

Workaround:

waitstone can be used to verify that the stone is completely shutdown.

Bug 49634 - Upgrade SortedCollection recompile fails for SimpleBlocks

Product: GemStone/S 64 Bit

Versions: 3.6.1, 3.6, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5

Fixed In: 3.6.2, 3.5.7

When upgrading from 3.2.x to 3.3 or later, methods and blocks need to be recompiled. Recompiling the sortBlocks of SortedCollections can be done automatically by the postconv step, provided they are SimpleBlocks.

However, this recompile was failing for upgrades from 3.2.x to to 3.5.x, due to changes in internal structures; the verification in the method ExecBlock >> recompile was overly general. The postconv utility invokes this, and reported failure to recompile these blocks.

There is no direct upgrade path from 3.2.x to 3.6.x, however, if a repository is upgraded to 3.5.x and then to 3.6.x without addressing the postconv failures, ExecBlock >> recompile failed in these versions as well.

Bug 49623 - GciTransportError, client user action not supported during non-blocking GCI call

Product: GemStone/S 64 Bit

Versions: 3.6.1, 3.6

Fixed In: 3.6.2

Impact: Informational

If a non-blocking GCI execution (such as GciNbExecute, etc.), invokes a client user action, the process will be unresponsive until the client polls for completion, using GciNbEnd, etc.

Starting with v3.6, the the flag GCI_PERFORM_noClientUseraction, is now on by default; this disables client user actions during non-blocking GCI calls, to avoid the risk of client hangs.

GemBuilder for Smalltalk polls frequently, and sets the GCI_PERFORM_noClientUseraction flag such that this bug did not previous exhibit, and the change in v3.6 does not affect behavior.

Other APIs, such as external sessions or Jadeite will now see an error of the form "a GciTransportError occurred (error 2728), client user action not supported during non-blocking GCI call actionNameOop 4202241" or "a GciTransportError occurred (error 2728), client user action not supported during GciNb call with GCI_PERFORM_noClientUseraction in flags, userAction name oop 9856769".

For historical reasons, GsFile is implemented as user actions; as a result, GsFile client operations were subject to this bug, and with the change in v3.6, GsFile client operations are disabled.

Workaround:

Replace calls to GciNbExecute* or GciNbPerform calls with equivalent DEBUG versions that allow you to specify the flags:

GciNbExecute*Dbg(... , int flags);
GciNbPerformNoDebug(... , int flags);

and ensure that the flags argument includes GCI_PERFORM_FLAG_ENABLE_DEBUG.

Bug 49616 - Upgrade from 3.5.6 to 3.6 or 3.6.1 fails

Product: GemStone/S 64 Bit

Versions: 3.6.1, 3.6, 3.5.6

Fixed In: 3.6.2

Upgrade from v3.5.6 to 3.6 or 3.6.1 fails, due to changes in the LdapDirectoryServer class.

Workaround:

Upgrading from 3.5.5 or earlier to 3.6.x will work successfully.

Bug 49577 - Upgrade from 2.x issue for SessionMethods installed for users other than DataCurator

Product: GemStone/S 64 Bit

Versions: 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.5, 3.3.4, 3.3.3, 3.3.2, 3.3.1, 3.3, earlier versions

Fixed In: 3.5

Session Methods are expected to be in the UserGlobals of DataCurator; if they are installed for users other than DataCurator, they are not handled correctly during upgrade.

Session Methods are an undocumented feature allowing session-specific methods to be installed. It is particularly used in the GLASS/Seaside/GsDevKit environment.

Workaround:

Executing the following code allows the upgrade to complete.

You must edit to specify the user name for the user in whose UserGlobals the session methods are installed, and to provide the password for SystemUser (for security, this should never be swordfish). Execute using topaz prior to upgrade.

```
set user SystemUser pass swordfish
login

run
 "Disable session methods to prevent executing methods that need recompile"
  AllUsers
    do: [ :userProfile |
      | assoc |
      userProfile userId = 'DataCurator'        "userId of user in which GLASS is
installed"
        ifTrue: [
          (assoc := userProfile symbolList resolveSymbol: #'UserGlobals') isNil
            ifFalse: [
              | key ug packagePolicy |
              ug := assoc value.
              key := GsPackagePolicy globalName.
              (packagePolicy := ug at: key otherwise: nil) isNil
                ifTrue: [
                  nil error: 'Expected a packagePolicy for ', userProfile userId ]
                ifFalse: [
                  GsFile gciLogServer: 'PRESERVED session methods for ', userProfile 
userId ] ] ]
        ifFalse: [
          (assoc := userProfile symbolList resolveSymbol: #'UserGlobals') isNil
            ifFalse: [
              | key ug packagePolicy homeSymbolDict |
              ug := assoc value.
              key := GsPackagePolicy globalName.
              (packagePolicy := ug at: key otherwise: nil) isNil
                ifFalse: [
                  ug at: key put: nil.
                  (homeSymbolDict := packagePolicy instVarAt: (packagePolicy class 
allInstVarNames indexOf: #homeSymbolDict)) isNil
                    ifTrue: [ GsFile gciLogServer: 'No homeSymbolDict for ', userProfile 
userId ]
                    ifFalse: [
                      GsFile gciLogServer: 'homeSymbolDict named ', homeSymbolDict name 
printString, ' for ', userProfile userId.
                      (homeSymbolDict at: key otherwise: nil) isNil
                        ifTrue: [
                          GsFile gciLogServer: 'No package found in session methods 
homeSymbolDict for ', userProfile userId ]
                        ifFalse: [
                          homeSymbolDict at: key put: nil.
                          GsFile gciLogServer: 'Package in session methods homeSymbolDict 
cleared for ', userProfile userId ]].
                       System commitTransaction ifFalse: [ nil error: 'COMMIT failed' ].
                       GsFile gciLogServer: 'DISABLED session methods for ' , userProfile 
userId ] ] ] ].
^ 'done'
%
logout
```

Bug 49568 - GsFile position: incorrect for r+ file mode

Product: GemStone/S 64 Bit

Versions: 3.6.1, 3.6, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.5, 3.3.4, 3.3.3, 3.3.2, 3.3.1, earlier versions

Fixed In: 3.6.2, 3.5.7

If a GsFile is open for read/write, e.g. when using GsFile class >> openUpdateOnServer: (mode r+), then after invoking atEnd, the file position is incorrect; GsFile >> position will return 0. Subsequent writes will be in the correct location.

If GsFile >> position: is sent, followed by GsFile >> atEnd, a subsequent write may not be in the correct location.

Bug 49566 - Object >> passivate produced incorrect results for some cases

Product: GemStone/S 64 Bit

Versions: 3.6.1, 3.6, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.5, 3.3.4, 3.3.3, 3.3.2, 3.3.1, earlier versions

Fixed In: 3.6.2, 3.5.7

If an instance of ScaledDecimal, Fraction, or FixedPoint has an instance variable containing a LargeInteger, and if that instance of ScaledDecimal, Fraction or FixedPoint appears more than once in the object graph, the passivation of the second reference is incorrect, and the reactivated object will contain an incorrect value/class.

for example,

```
    largeSD := 1222921504606846975 73.45s2.
    coll := { largeSD . largeSD }.
    coll = coll passivate activate
```

Also, when an Object's closure contains ExecBlocks (such as SortedCollections, which contain a sortBlock), the source string for the block is included in the passivated form. If in-memory GC occurs while passivating (this is more likely to occur in later versions of GemStone), the sourceString may be flushed and not included in the passivated form, which creates an error on activation.

Bug 49504 - Symbol GC slow with many dead symbols

Product: GemStone/S 64 Bit

Versions: 3.6.1, 3.6, 3.5.5, 3.5.4, 3.5.3, 3.5.1, 3.5, 3.4.5, 3.4.4

Fixed In: 3.6.2, 3.5.6

When Symbol GC finds a very large number of possible dead symbols, the write set union sweep became unreasonably slow. This was a result of the possibleDeadSymbol structure containing large collision buckets; the objects referenced from the collision buckets were being read, which is unnecessary, since all contained objects would be symbols.

Bug 49450 - Logsender not connected to stone risks transmitting incomplete tranlog records

Product: GemStone/S 64 Bit

Versions: 3.6.1, 3.6, 3.5.7, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4

Fixed In: 3.6.2

In a hotstandby system, when the logsender is not connected to the master Stone, the logsender relies on disk file information to check for when a transaction log record is ready to send to the logreceiver.

However, logical tranlog records may cross a file system block boundary, and this not detected by the logsender’s disk file checks. In this case, the transaction log record that is sent may be incomplete.

See also bug 49449, a related problem with transaction log records and fync timing.

Workaround:

It is strongly encouraged to use the -s flag to avoid this risk; the Stone tracks the logical record boundaries and provides this information to the logsender.

Bug 49449 - Logsender risk of transmitting incomplete tranlog records due to fync timing

Product: GemStone/S 64 Bit

Versions: 3.6, 3.5.5, 3.5.4, 3.5.3, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4

Fixed In: 3.6.1, 3.5.6

n a hotstandby system, the Stone tracks the completed transaction log records, and when a record has been written, the logsender reads from the on-disk transaction logs and transmits the tranlog records to the logreceiver.

When the write is complete, however, there is a risk that the disk fsync was not finished, in which case the record that is read from disk by the logsender and sent to the logreceiver may be incomplete.

See also bug 49450, a related problem with logical transaction log records.

Bug 49448 - GEM_ABORT_MAX_CRS = 0 does not work as documented

Product: GemStone/S 64 Bit

Versions: 3.6, 3.5.5, 3.5.4, 3.5.3, 3.5.1

Fixed In: 3.6.1, 3.5.6

The configuration parameter GEM_ABORT_MAX_CRS controls the number of commit records that are analyzed on Gem abort, to compute the write set union. If the number of commit records is higher than this limit, all committed in-memory objects must be re-read.

The documentation says that a value of zero means the setting is "unlimited". This limit is not being applied; a value of zero is effectively treated as analyze no commit records.

Also, the documentation incorrectly states that this feature applies to Gems that are in transaction. It actually applies on all aborts, regardless if the Gem is in transaction or not.

Workaround:

To ensure all commit records are always analyzed, use the maximum value, 2147483647.

Bug 49434 - If abort updates over more than GEM_ABORT_MAX_CRS commit records, GBS traversals may not include updates to persistent objects

Product: GemStone/S 64 Bit

Versions: 3.6, 3.5.5, 3.5.4, 3.5.3, 3.5.1

Fixed In: 3.6.1, 3.5.6

On abort, a Gem must move its commit record view from the view acquired on the previous commit or abort, to the latest commit record.

The configuration parameter GEM_ABORT_MAX_CRS allows you to limit the number of commit records that are analyzed on Gem abort; if the number of commit records that must be crossed is larger than this, objects are all invalidated and must be re-read.

However, of the number of commit records that must be crossed is higher than this limit, the subsequent traversal result seen by GBS does not include all of the exported objects.

This means that if there are unmodified committed objects that are replicated to GBS, and these objects have been changed by other sessions, these changes are not visible in the Gem's view after the abort.

The default GEM_ABORT_MAX_CRS is 100000 for local sessions and 33333 for remote sessions, so this bug only applies when the commit record backlog exceeds this number.

Workaround:

If your application sees large commit record backlogs, set GEM_ABORT_MAX_CRS to the maximum, 2147483647.

While the documentations states that a setting of 0 means no limit; this is not working correctly in some product versions. See
bug 49448

Bug 49411 - X509-secured netldi and hostagent processes have memory leaks

Product: GemStone/S 64 Bit

Versions: 3.6, 3.5.5, 3.5.4, 3.5.3, 3.5.1, 3.5

Fixed In: 3.6.1, 3.5.6

The X509-secured GemStone feature includes a specialized version of the NetLDI, and the Host Agent process. These processes had a memory leak in login/logout.

Workaround:

Restarting the X509 NetLDI and the Host Agent will free the memory.

Bug 49406 - ProfMonitor profiling file must be manually removed

Product: GemStone/S 64 Bit

Versions: 3.6, 3.5.5, 3.5.4, 3.5.3, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4

Fixed In: 3.6.1, 3.5.6

Using ProfMonitor or ProfMonitorTree >> monitorBlock: to monitor Smalltalk execution creates a file gem_pid_nnnnnn.pro in the home directory of the user (for an RPC login), or in the working directory for a linked login. This file is not being removed after the report is returned. Large profiles files may use a considerable amount of disk space.

Workaround:

After profiling is complete, manually delete these .pro files files.

Bug 49388 - Reclaim log fails to print reclaim entries

Product: GemStone/S 64 Bit

Versions: 3.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5

Fixed In: 3.6.1, 3.5.6

Entries providing information on reclaim activity, after the first entry, were not printed to the ReclaimGem's log file. This was a consequence of the change to printing millisecond timestamps.

Bug 49382 - Nested transactions that modify kinds of IdentityBag have risk of undetected corruption in tranlogged data

Product: GemStone/S 64 Bit

Versions: 3.6, 3.5.5, 3.5.4, 3.5.3, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.2, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.x, 3.1.x, 3.1

Fixed In: 3.6.1, 3.5.6

Impact: Critical

When collections are modified in GemStone, only the changes are logged to the transaction logs; this incremental logging avoids transaction log bloat. In the case of modifications to certain kinds of collections (subclasses of IdentityBag), when the modification is done within a nested transaction, the incremental logging was not correct.

If these tranlogs were replayed, either after restore from backup, or in a warm or hot standby system, the restored collection may contain nils rather than valid results. Object audit did not detect this, since the elements were nil.

This bug only occurs when the modifications are from within nested transactions, e.g., after invoking beginNestedTransaction; and only for modifications to instances of subclasses of IdentityBag. The affected collections are:

```
IdentityBag
  BucketValueBag
  IdentitySet
    AbstractUserProfileSet
      UserProfileSet
    ClassSet
    GsObjectSecurityPolicySet
    RcIdentitySet
    StringPairSet
    SymbolSet
    UserProfileGroup ( groupName)
  RcIdentityBag ( components)
  RcLowMaintenanceIdentityBag
```

Workaround:

New versions will be released as soon as they are ready. Contact GemTalk Techincal Support for early access or special patch releases.

Bug 49379 - DataCurator permission error on addPerformOnServerCommand: in upgraded repository

Product: GemStone/S 64 Bit

Versions: 3.6

Fixed In: 3.6.1

A feature was added in v3.6, that allowed specific command paths to be allowed for a UserProfile that had the NoPerformOnServer inverse privilege. This feature required adding an instance variable to the UserSecurityData class, which holds sensitive information and can only be viewed or modified by SystemUser.

However, instance of UserSecurityData were not explicitly migrated during the upgrade process. While these instance will auto-migrate, since they are not accessible to DataCurator, they cannot be auto-migrated by DataCurator.

If DataCurator attempted to add a path to the white list for a UserProfile in an upgraded repository, this resulted in a permission error.

Workaround:

Login as SystemUser to perform the addPerformOnServerCommand:. After the UserSecurityData is auto-migrated, DataCurator can maintain the list.

Bug 49365 - Upgrade failures if DataCurator password is not at default, or has NoGsFileOnServer

Product: GemStone/S 64 Bit

Versions: 3.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5

Fixed In: 3.6.1, 3.5.6

Before running the upgradeImage script, you must change SystemUser's password to the default. The upgradeImage script also performs some operations as DataCurator and should automatically update DataCurator's password without the user needing to take action. However, a timing error was introduced so that the DataCurator password is not updated correctly, and this part of upgradeImage fails.

Workaround:

Manually set DataCurator's password to the default before running the upgradeImage step of the upgrade process.

You should also verify that DataCurator does not have the NoGsFileOnServer reverse privilege, and remove it if present.

Bug 49339 - Symbol GC may collect referenced Symbols

Product: GemStone/S 64 Bit

Versions: 3.6, 3.5.5, 3.5.4, 3.5.3, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.x

Fixed In: 3.6.1, 3.5.6

When Symbol garbage collection is enabled (STN_SYMBOL_GC_ENABLED is configured to true), there is a risk that a referenced symbol may be garbage collected.

If the only reference to a symbol is from an object in the possible dead set, and that object is voted to stay alive by a session, during voting after an MFC, then in some cases this symbol may be garbage collected.

Workaround:

STN_SYMBOL_GC_ENABLED is false by default. To avoid risk, only turn this on to enable symbol garbage collection during maintenance periods, when sessions other than the MFC session are not running.

Bug 49328 - Filein via topaz input may break in some cases of GsPackagePolicy enabled

Product: GemStone/S 64 Bit

Versions: 3.6

Fixed In: 3.6.1

In some configurations where GsPackagePolicy is inadvertently enabled, after upgrade to v3.6 the ordinary filein of GemStone code (e.g. using the topaz input command) may be broken. Filein appears to work, but after logout and login, the changes disappear. If the class existed prior to the filein, all methods may be missing after logout/login.

With GsPackagePolicy enabled, the use of the transient symbol dictionary was inadvertently enabled. v3.6 changed the way the transient and persistent symbolLists are handled. The result is that the filed-in changes could end up in the transient symbol dictionary, and thus disappear on logout.

fileouts that include the class definition (output using standard GemStone fileout operators), have a fileout format includes topaz directives to remove all methods on that class. The effects of removeAllMethods are not transient, so after filein and logout, if the class previously existed, it was left with no methods at all.

Workaround:

If GsPackagePolicy was inadverently enabled though previously benign, disable it using:

```
   GsPackagePolicy current disable
```

Bug 49319 - Configuring Shared Memory on Mac "Big Sur"

Product: GemStone/S 64 Bit

Versions: 3.6.3, 3.6.2, 3.6.1, 3.6, 3.5.7, 3.5.6, 3.5.5, 3.4.5

Impact: Informational

On the "Big Sur" release of MacOS (OSx v11.x), the increased security means shared memory configuration, as required for large GemStone installations, has become more restrictive. Setting values in /etc/sysctl.conf is no longer effective. Big Sur is not yet officially supported, but GemStone is expected to run on this platform.

As root, you can manually change the shared memory parameters using /usr/sbin/sysctl; however, changes made in this way are lost on reboot.

To upgrade the configurations permanently, you must set the configuration values in an XML-formatted file with the path and name:
  /Library/LaunchDaemons/com.gemtalksystems.shared-memory.plist

You will need to reboot, for the settings to take effect.

An example plist file for a host with 8 GB of RAM is available at com.gemtalksystems.shared-memory.plist, with the contents below; edit the settings as appropriate for your machine RAM and GemStone configuation.

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN"
"http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
<dict>
  <key>Label</key>
  <string>shmemsetup</string>
  <key>UserName</key>
  <string>root</string>
  <key>GroupName</key>
  <string>wheel</string>
  <key>ProgramArguments</key>
  <array>
    <string>/usr/sbin/sysctl</string>
    <string>kern.sysv.shmmax=6442450944</string>
    <string>kern.sysv.shmall=1572864</string>
  </array>
  <key>KeepAlive</key>
  <false/>
  <key>RunAtLoad</key>
  <true/>
</dict>
</plist>
```

Bug 49280 - Risk of ProcessScheduler _runNextProcess failed errors

Product: GemStone/S 64 Bit

Versions: 3.6

Fixed In: 3.6.1

The ProcessScheduler may encounter an error "ProcessScheduler _runNextProcess failed, new 20 old 20" under some timing circumstances; specifically, if an EINTR occurs, the method _findReadyProcess will return nil.

This scenario may have been possible in earlier versions, but has became more common in 3.6.

Bug 49245 - Missing libssl or libkrb5 causes GCI client process to terminate

Product: GemStone/S 64 Bit

Versions: 3.6, 3.5.7, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.1, 3.5

Fixed In: 3.6.1

When a GCI client (such as GBS) logs in, it must have access to a set of shared libraries. If libssl or libkrb5 was not available, on some platforms this caused the GCI client process to terminate.

Note that in no case will the login succeed; the required shared libraries are required.

Workaround:

Copy all required shared library files to the client. The required shared libraries are described in the Installation Guide for GBS.

Bug 49184 - CodeLibrarianUser and CodeLibrarianUserObjectSecurityPolicy in upgraded repositories

Product: GemStone/S 64 Bit

Versions: 3.6.3, 3.6.2, 3.6.1, 3.6

Impact: Informational

CodeLibrarianUser and CodeLibrarianUserObjectSecurityPolicy were added in v3.5 to support the initial limited support for tonel-format source code and X509-secured GemStone. Further changes to GemStone's code management in 3.6 made this user obsolete.

Applications that were running in v3.5, or that upgraded to 3.5.x as part of an upgrade from an earlier version before another upgrade to v3.6.x, will have had this user and security policy created during upgrade.

In v3.6.x, CodeLibrarianUser will be present in the DeletedUsers colllection, and may be manually deleted. Globals does not contain a link to CodeLibrarianUserObjectSecurityPolicy, but the object security policy is present in SystemRepository.

Bug 49179 - Transient Session SymbolList Changes in 3.6

Product: GemStone/S 64 Bit

Versions: 3.6.3, 3.6.2, 3.6.1, 3.6

Platform All Platforms

Impact: Informational

In earlier versions of GemStone/64, when a session logged in, a session-specific transient copy of the user's symbol list would be created and used for all symbol lookups. This session symbol list copy is accessible via:

```
GsCurrentSessions currentSession symbolList
```

This session symbol list was set up as a copy, so that changes to the persistent user symbol list at "System myUserProfile symbolList" would not impact symbol lookup in the current session. And as a *transient* copy, aborts would not rollback changes made to the session symbol list. This design provided useful separation between the "official" user symbol list and the "working version" of an individual session, but required extra effort when synchronization between the two was required.

In 3.6, this behavior has changed slightly. When first logged in, the session symbol list at "GsCurrentSessions currentSession symbolList" will now instead directly reference the persistent user symbol list at "System myUserProfile symbolList". This will allow changes to the persistent user symbolList to be transparently visible to code that does not need to worry about managing or refreshing the transient session symbol list.

The following methods will restore the original transient behavior of the session symbol list for those sessions that require the old behavior:

```
System class >> refreshTransientSymbolList
GsCurrentSession >> transientSymbolList:
GsCurrentSession >> transientSymbolList
```

Bug 49109 - Slow stone startup with large possibleDead

Product: GemStone/S 64 Bit

Versions: 3.5.7, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4

Fixed In: 3.6

After a markForCollection, the possibleDead set holds objects that are candidates for garbage collection, until all sessions have voted and these objects are determined to be really dead (promoteToDead). If the stone is shut down after the markForCollection, but before promoteToDead, and the possibleDead set is very large, the restart of the stone may take a long time (30 minutes was observed with a 4 billion size possibleDead). This problem appears to be much less likely with possibleDead set sizes less than 2 billion.

To confirm this is the cause of a slow stone startup, execute pstack on the stone process. If the code is in StnLoopRemoveNonExistentFromPossDead, this bug is the cause of the slowness.

Workaround:

Avoid shutting down while the possibleDead set is very large. If shutdown is unavoidable, be prepared for the delay; the stone will start up correctly after the possibleDead are checked.

Bug 49106 - Abort of a nested transaction may not roll back changes to IdentityBag

Product: GemStone/S 64 Bit

Versions: 3.5.3, 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4

Fixed In: 3.5.4

Changes that are made to an instance of a kind of IdentityBag, within a nested transaction, were not always correctly rolled back by an abort. If the outer transaction committed, it was possible for the change that was meant to be aborted and removed, to become committed.

Bug 49104 - Nested transaction abort does not undo change to an Object's object security policy

Product: GemStone/S 64 Bit

Versions: 3.5.3, 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4

Fixed In: 3.5.4

The actions of sending #objectSecurityPolicy: to an object, to change the objectSecurityPolicy associated with the object, is not rolled back if it is done within a nested transaction that is subsequently aborted. A later commit of the outer transaction can make this change persistent.

Bug 49101 - Result of listInstances on objects that are equal/not identical may be incorrect

Product: GemStone/S 64 Bit

Versions: 3.5.3

Fixed In: 3.5.4

An internal comparison when building the results of a listInstances:, in the method Repository >> _buildAllRefsResul, used = rather than == to perform the comparison. If the input argument array includes multiple objects, and two (or more) of these objects were equal but not identical to each other, the list of the results of the listInstances could have missing or incorrect entries.

Workaround:

Perform listReferences scans separately for equal but not identical objects.

Bug 49092 - IdentityBag adds in nested transactions may not be tranlogged correctly

Product: GemStone/S 64 Bit

Versions: 3.5.3, 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4

Fixed In: 3.5.4

Impact: Critical

If additions are made to an IdentityBag or a subclass of IdentityBag within a nested transaction, the additions may not be logged correctly in the transaction logs.

If the database is later restored from backup and the transaction logs replayed, or on automatic recovery after an unclean shutown, these added elements will be missing, and the variable that tracks the size of the collection will be incorrect.

Workaround:

The fix is in the C code, and requires a new version of GemStone. Versions with the fix should be released soon; contact GemTalk Technical Support for assistance.

Bug 49091 - GsExternalSession disconnected by GsSocket class >> closeAll

Product: GemStone/S 64 Bit

Versions: 3.5.3

Fixed In: 3.5.4

In version 3.5.3, to support the added method GsExternalSession>>waitOnReadready, an instance of GsExternalSession creates a GsSocket representation of its underlying socket to the gem on login.

If your application also performs GsSocket class >> closeAll, this method closes this socket, which will disconnect the GsExternalSession.

In v3.5.4 and later releases, the method GsSocket class >> closeAll does not close instances created using GsSocket class >> fromFileHandle:.

Workaround:

If you are not using GsExternalSession >> waitForReadReady, you can edit the method GsExternalSession>>_postLogin:, to comment out the final two lines:

```
    sock := GsSocket fromFileHandle: fd .
    self dynamicInstVarAt: #_socket put: sock
```

If you are using waitForReadReady, do not use GsSocket class >> closeAll.

Bug 49023 - pstack broken on Mac

Product: GemStone/S 64 Bit

Versions: 3.5.3, 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4

Fixed In: 3.5.4

The pstack script relies on the OS-level command gdb or lldb.

In versions of the Mac that no longer have gbd installed, for GemStone versions prior to v3.5.3, pstack does not work.

With the added security required by Apple for later Catalina releaes, and implemented for GemStone in v3.5.3, lldb would get an EPERM error attempting to attach to a process to get a stack trace.

Bug 48965 - Indexes with optionalPathTerm error on removing element without term

Product: GemStone/S 64 Bit

Versions: 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4

Fixed In: 3.5.3

GemStone indexing allows you to specify the optionPathTerms option, which allows you to add objects to an indexed collection that do not have the indexed path term.

Attempting to remove an object that did not have the indexed path term, however, could encounter an error.

Workaround:

File in the following as SystemUser, and commit.

```
run
        GsOptionalPathTerm removeSelector: #'_nextObjectFor:'.
        GsOptionalPathTerm removeSelector: #'_errorInvalidOffset:'.
%

method: GsOptionalPathTerm
_findAllValuesForIdenticalRootObject: rootObject
       " if index directly on NSC elements or anObject is nil "

       | ivOffset |
       ivOffset := self _ivOffsetFor: rootObject.
       ivOffset
               ifNil: [
                       "no values in index for rootObject"
                       ^ IdentityBag new ].
       ^ super _findAllValuesForIdenticalRootObject: rootObject
%
category: 'Accessing'
method: GsOptionalPathTerm
_nextObjectFor: anObject atInstVar: anIvOffset
       "Returns the object at the instance variable that corresponds to the receiver"
       | ivOffset |
       ivOffset := self _ivOffsetFor: anObject.
       ivOffset ifNil: [ ^ nil ].
       ^ super _nextObjectFor: anObject atInstVar: anIvOffset
```

```
%

category: 'Adding'
method: GsOptionalPathTerm
addMappingsForObject: anObject root: rootObject logging: aBoolean
       "Add dependency list entries for anObject."

       | ivOffset |
       (nil == anObject or: [ self size == 0 ])
               ifTrue: [ ^ self ].
       (ivOffset := self _ivOffsetFor: anObject) == nil
               ifTrue: [ ^ self recordNilOnPathForRoot: rootObject ].
       ^ super addMappingsForObject: anObject root: rootObject logging: aBoolean
%

category: 'Removing'
method: GsOptionalPathTerm
removeMappingsFor: anObject root: rootObject lastOne: aBoolean logging: doLogging
  "Remove entries in the btree and dependency lists for anObject."

       | ivOffset |
       (nil == anObject or: [ self size == 0 ])
               ifTrue: [ ^ self ].
       (ivOffset := self _ivOffsetFor: anObject) == nil
               ifTrue: [ ^ self removeNilOnPathForRoot: rootObject ].
       ^ super removeMappingsFor: anObject root: rootObject lastOne: aBoolean logging:
doLogging
%
category: 'Private'
method: GsPathTerm
_findAllValuesForIdenticalRootObject: rootObject
       " if index directly on NSC elements or anObject is nil "

       | key tmpList |
       (self indicatesIndexOnNscElements or: [ nil == rootObject ])
               ifTrue: [ key := rootObject ]
               ifFalse: [ key := self _nextObjectFor: rootObject ].
       tmpList := IdentityBag new.
       self updateBtree btreeRoot _findAllValuesForIdenticalKey: key into: tmpList.
       ^ (tmpList occurrencesOf: rootObject) <= 1
%
category: 'private'
method: PathTerm
_findAllValuesForIdenticalRootObject: rootObject
       " if index directly on NSC elements or anObject is nil "

       | key tmpList |
       (self indicatesIndexOnNscElements or: [ nil == rootObject ])
               ifTrue: [ key := rootObject ]
               ifFalse: [ key := self _nextObjectFor: rootObject ].
       tmpList := IdentityBag new.
       self updateBtree btreeRoot _findAllValuesForIdenticalKey: key into: tmpList.
       ^ (tmpList occurrencesOf: rootObject) <= 1
%
category: 'Indexing Support'
method: UnorderedCollection
_isLastOccurrenceInIndexObjects: anObject
```

```
 "Returns true if the given object is maintained in the indexing objects for one
occurrence."

 | rootTerms pathTerm key val val2 num parentTerm |
 rootTerms := self _indexedPaths rootTerms.     " find a path term with a mapping in the
index dictionary "
 1 to: rootTerms size do: [ :i |
   pathTerm := rootTerms at: i.
   (pathTerm _isObsoletePathTerm)
     ifFalse: [
       pathTerm isRangeEqualityIndexLastPathTerm
         ifTrue: [
           pathTerm offset == 1
             ifTrue: [ ^ pathTerm _findAllValuesForIdenticalRootObject: anObject ]
             ifFalse: [
               pathTerm hasIndexDictionary
                 ifTrue: [
                   pathTerm := pathTerm getParentTerm.
                   val  := pathTerm updateDict
                     at: anObject
                     term: pathTerm
                     otherwise: nil.
                   (BucketValueBag _hasInstance: val)
                     ifTrue: [
                       " see if more than one mapping "
                       pathTerm indicatesNsc
                         ifTrue: [
                           num := val occurrencesOf: self.
                           num < val size
                             ifTrue: [
                               " if anObject is contained in other NSCs "
                               ^ false ].
                           pathTerm := pathTerm getParentTerm.  " get path term before
this one "
                           val2 := pathTerm updateDict
                             at: self
                             term: pathTerm
                             otherwise: nil.
                           (BucketValueBag _hasInstance: val2)
                             ifTrue: [ ^ val2 size == num ]
                             ifFalse: [ ^ true ] ]
                         ifFalse: [ ^ (val occurrencesOf: self) <= 1 ] ]
                     ifFalse: [ ^ true ] ]
                 ifFalse: [ ^ (self occurrencesOf: anObject) <= 1] ] ]
         ifFalse: [
           pathTerm hasIndexDictionary
             ifTrue: [
               " get key to look up in index dictionary "
               (pathTerm indicatesIndexOnNscElements or: [ nil == anObject ])
                 ifTrue: [ key := anObject ]
                 ifFalse: [
                   " see if a path with '*.*' in it "
                   pathTerm indicatesNsc
                     ifTrue: [
                       val := pathTerm updateDict
                         at: anObject
                         term: pathTerm getParentTerm
```

```
                          otherwise: nil.
                        ^ (BucketValueBag _hasInstance: val) not ]
                      ifFalse: [ key := pathTerm _nextObjectFor: anObject ] ].
                val := pathTerm updateDict at: key term: pathTerm otherwise: nil.        "
look up the mapping in the index dictionary "
                (BucketValueBag _hasInstance: val)
                  ifTrue: [
                    " see if more than one mapping "
                    (num := val occurrencesOf: anObject) <= 1
                      ifTrue: [ ^ true ]
                      ifFalse: [
                        " see if multiple occurrences are due to more than one object
                        referencing self "
                        " if there is a parent term, it is a SetValuedPathTerm "
                        parentTerm := pathTerm getParentTerm.
                        parentTerm == nil
                          ifTrue: [ ^ false ].
                        val2 := pathTerm updateDict
                          at: anObject
                          term: parentTerm
                          otherwise: nil.
                        (BucketValueBag _hasInstance: val2)
                          ifTrue: [ ^ (val2 occurrencesOf: self) == num ]
                          ifFalse: [ ^ false ] ] ]
                  ifFalse: [ ^ true ] ]
              ifFalse: [ ^ (self occurrencesOf: anObject) <= 1] ] ] ].
 ^ true " if we get this far, there are only incomplete indexes "
%
```

Bug 48885 - topaz prompt for password not supported on login

Product: GemStone/S 64 Bit

Versions: 3.5.7, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4

Impact: Informational

In earlier releases, it was possible to skip the set password command in topaz, in which case the topaz login command would prompt you for the password.

To support other password authentication schemes, this particular sequence is no longer implemented, and results in a login error.

Workaround:

Entering set password, but omitting the argument, will prompt for a password immediately (not on login). For example,

```
topaz> set username DataCurator
topaz> set password
GemStone Password?
<enter password>
topaz> login
```

Bug 48879 - Cross-version external sessions limited to specific other versions

Product: GemStone/S 64 Bit

Versions: 3.5.4, 3.5.3

Fixed In: 3.6, 3.5.5

GsTsExternalSession allows you to start an external session against a Stone running a different version of GemStone. This feature requires that the other version also support the cross-version login feature.

The version verification in 3.5.3 and 3.5.4 was limited and will fail for later versions (e.g., against a 3.5.5 stone).

Also note that the version requires exactly '3.6.0', and does not accept '3.6'.

Workaround:

We recommend upgrading to 3.5.5 or later to use cross-version external sessions.

For testing, you may login as SystemUser and edit GciTsLibrary >> version:

Bug 48875 - Memory leaks in NetLDI

Product: GemStone/S 64 Bit

Versions: 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4

Fixed In: 3.5.3

There were several memory leaks that occurred on requests to the NetLDI. Over time, the memory footprint of the NetLDI process could become very large.

When not running in guest mode, there was a 16 bytes per-request leak in v3.4.x and v3.5.x.

In all NetLDI modes, there was a 2k byte per-request leak in v3.5.x.

Workaround:

Restarting the NetLDI will clear memory.

To ensure leaks such as this are caught, versions with the fix for this bug includes a cache statistic HeapKBytes, with the size of the process's heap in kilobytes; and statmonitor by default records statistics on NetLDI processes. Continually increasing HeapKBytes in a NetLDI demonstrates the problem.

Bug 48841 - Issues running GemStone on latest versions of Mac Catalina

Product: GemStone/S 64 Bit

Versions: 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, older versions

Fixed In: 3.5.3

Impact: Informational

When the GemStone distribution is downloaded from the GemTalk website, users running on the most recent versions of OSX 10.15 (Catalina) may see the executables blocked as a security risk, and it may be difficult to run GemStone software.

When software is downloaded from a web application, the Mac GateKeeper now will set LSFileQuarantineEnabled on, which results in executables being quarantined.

Workaround:

LSFileQuarantineEnabled is not set for applications downloaded using curl or ftp that is not from a web application.

For example

curl --output GemStone64Bit3.5.2-i386.Darwin.zip
https://downloads.gemtalksystems.com/pub/GemStone64/3.5.2/GemStone64Bit3.5.2-i386.Darwin.zip

You can verify using

```
    codesign --verify --verbose <executable>
```

for example,

```
    prompt> codesign --verify --verbose $GEMSTONE/sys/stoned
```

Bug 48836 - statmonitor -X started from STN_STATMONITOR_ARGS does not stop when Stone stops

Product: GemStone/S 64 Bit

Versions: 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4

Fixed In: 3.5.3

Impact: Informational

The statmonitor -X argument runs a startmonitor that collects host statistics only, and does not attach to a Stone.

When statmonitor -X is started during Stone startup by including it in the argument to STN_STATMONITOR_ARGS or similar configuration parameters, the statmonitor was launched, but would of course not naturally stop when the Stone was stopped.

Workaround:

Do not use -X in the arguments to STN_STATMONITOR_ARGS; you must manually track down and kill statmonitor processes started in this way.

In 3.5.3 and later, -X is disallowed in STN_STATMONITOR_ARGS.

Bug 48827 - GemStone does not run on Ubuntu 20.04

Product: GemStone/S 64 Bit

Versions: 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.8, 3.3.7, 3.3.6, earlier versions

Fixed In: 3.5.3

Impact: Informational

Due to OS changes in mmap(), existing versions of GemStone cannot run on the latest version of Ubuntu, v20.04.

Workaround:

Upgrade to a version of GemStone that is supported on Ubuntu 20.04, such as v3.5.3 or later.

Bug 48805 - DateAndTimes now invariant: offset: and beRounded behavior change

Product: GemStone/S 64 Bit

Versions: 3.6.3, 3.6.2, 3.6.1, 3.6

Impact: Informational

With the addition of special type SmallDateAndTime, instances of DateAndTime, which were previously modifiable, are now invariant. SmallDateAndTime instances, since the value is encoded in the OOP, are inherently not modifiable, and DateAndTime and SmallDateAndTime require consistent behavior.

The method offset:, which previously modified and returned the receiver, returns a new instance of DateAndTime with the new offset in v3.6. If your code expects the receiver to be modified, this may result in silent failures. Since this is an ANSI method, deprecation or error was not appropriate.

The method beRounded also modified the receiver, and will throw an invariant object error for DateAndTime and a shouldNotImplement for SmallDateAndTime. Code should use the method rounded, which returns a new instance.

There are other changes and ways DateAndTime behavior has changed in v3.6, refer to the Release Notes for v3.6 for details.

Bug 48789 - listReferences: results do not respect order of argument

Product: GemStone/S 64 Bit

Versions: 3.5.2, 3.5.1, 3.5

Fixed In: 3.5.3

The method System>>listReferences: takes an array of classes and returns an array of arrays of instances. The order of subarrays within the results array no longer matches the order in the argument.

Workaround:

Limiting listReferences: arguments to a single class avoids issues.

The method allReferences: performs the same function, but returns sub-Arrays that include the class and a GsBitmap of results, allowing you to easily match class to results.

Bug 48734 - gslist -p returns invalid process ID when process is not running/status killed

Product: GemStone/S 64 Bit

Versions: 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.x, earlier versions

Fixed In: 3.5.2

gslist -p returns the processId only of one or more GemStone processes. The processId comes from the GemStone locks files, and thus is returned for processes that have the status killed and are no longer running. For processes that have status killed, the PID may have been reused for a different process, including a non-GemStone process. The results from gslist -p in this case is wrong.

Workaround:

Always use the -c argument (which clears killed processes) in addition to the -p. For example

```
gslist -c -p gs64stone
```

Bug 48706 - Hotstandby requires default-named transaction log files

Product: GemStone/S 64 Bit

Versions: 3.5.7, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.x, 3.1.0.6, 3.1.x

Fixed In: 3.6

Transaction log file names by default are tranlogNN.dbf, but this can be configured to use a different prefix using the configuration parameter STN_TRAN_LOG_PREFIX.

The logsender component of a hotstandby system did not work correctly if STN_TRAN_LOG_PREFIX was set, and the transaction logs generated by the master stone did not have the default name tranlogNN.dbf.

Workaround:

Avoid using STN_TRAN_LOG_PREFIX in hotstandby systems.

Bug 48580 - Repository scans and possibleDead objects

Product: GemStone/S 64 Bit

Versions: 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4

Impact: Informational

Following a markForCollection, the Stone has a list of possibleDead objects which it provides to sessions on abort. These objects must be voted on by the sessions, before each object in the possibleDead can be determined to be truly dead and reclaimed, or still referenced, retained and not reclaimed.

If a repository scan operation, such as objectAudit, listInstances, etc. is done while there are possible dead objects in the repository, the scan in versions up to v3.4.5 does not scan the possible dead. If objects in the possible dead are determined by voting to be still referenced, this means the scan result could have missed some valid result objects.

In later versions, the repository scan operation may return objects in the results, that will later be finalized and reclaimed. If the results of the scan are retained through the garbage collection cycle and the session commits or aborts, or if the results are persisted, these objects may not existed or that turn out to be a different actual object (if the OOP is reused for another object).

Workaround:

Always do a markForCollection and reclaimAll, so that there are no possibleDead, to get the completely reliable results for repository scans.

Bug 48559 - Changing result of isSpecial for classes for specials, such as Character and SmallInteger

Product: GemStone/S 64 Bit

Versions: 3.6.3, 3.6.2, 3.6.1, 3.6, 3.5.7, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.8, 3.3.7, 3.3.6, earlier versions

Impact: Informational

Historically, the method isSpecial returns true when sent to objects that are instances of Class such as Character, where the Class itself is not Special, but the instances of that class would be special. For example, Character isSpecial returns true, in spite of the class named Character not actually being a special instance.

Starting with v3.6, the method Behavior >> isSpecial that overrides the general implementation is removed, allowing the general implementation in Object to be inherited, and the method returns false for Classes of specials. The nature of this change made deprecation unfeasible. Ongoing development in pre-3.6 code should be aware and write code accordingly.

With v3.6 and 3.5.3, to check if Classes have instances that are special, you should use the method areInstancesSpecial. You should review your code before upgrading to 3.6, and replace uses of isSpecial for classes to use areInstancesSpecial.

Workaround:

Uses of isSpecial for classes should be changed to areInstancesSpecial.

The kernel change is to remove the method Behavior >> isSpecial, so that it inherits from Object. While base modifications are discouraged, you may define Behavior >> isSpecial with the previous implementation, which will restore the previous behavior, to avoid disruption your application.

Bug 48533 - GsFile >> removeServerFile:, removeClientFile: do not handle filenames with ? or *

Product: GemStone/S 64 Bit

Versions: 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.2, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, earlier versions

Fixed In: 3.5.2

The GsFile methods removeServerFile: and removeClientFile: were not able to find and remove files that contained the literal characters * or ?.

Note that these methods did and do not support wild card matching to remove files.

Workaround:

Execute code such as the following:

```
System performOnServer: 'rm fileWith?or*'
```

Bug 48520 - Periodic 30-second hangs with small STN_MAX_SESSIONS

Product: GemStone/S 64 Bit

Versions: 3.5.1

Fixed In: 3.5.2

If STN_MAX_SESSIONS is explicitly set to a small value (such as 10), there is likely to be a periodic hang for as long as 30 seconds, after each 10 to 20 commits.

The problem is related to the speed in which the completed tranlog AIO buffers are recycled and available for use for subsequent commits. The STN_MAX_SESSIONS setting controls how many tranlog AIO buffers are allocated, and with sufficient buffers this problem does not occur. The default setting for STN_MAX_SESSIONS, 40, is recommended and with this setting, the problem does not occur.

Workaround:

Avoid setting STN_MAX_SESSIONS to a small value. Even if the license limit is lower than the default of 40 (such as with the community license), the shared page cache allocates tranlog AIO buffers based on the STN_MAX_SESSIONS settings.

Bug 48506 - Removing dynamic instance variable from large SequenceableCollection corrupts size

Product: GemStone/S 64 Bit

Versions: 3.5.1, 3.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.2, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, earlier versions

Fixed In: 3.5.2, 3.4.5

Impact: Critical

Removing a dynamic instance variable may cause a SequenceableCollection to have a corrupted size, becoming incorrectly very large.

This only applies to large collections: a pointer collection larger than about 2K or a byte collection larger than about 16K. Objects that are too large to fit on a single page are implemented internally using a tree-based structure.

Workaround:

Affected collections must be manually resized.

Bug 48489 - gslist -x does not include exe entry for -n process on -m host

Product: GemStone/S 64 Bit

Versions: 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.2, 3.3.1

Fixed In: 3.5.2

gslist -x reports complete details for GemStone processes; -m reports on processes on remote nodes, and -n specifies to limit the report to a specific named process, so you can specify to only see the details for the stone, for example.

When these three options are all used, the "exe=<executable>" information is not included in the gslist output.

Bug 48477 - Upgrade from 3.3.x to 3.5.x fails

Product: GemStone/S 64 Bit

Versions: 3.5.1, 3.5

Fixed In: 3.5.2

Upgrading a GemStone repository from version 3.3.x to version 3.5.x fails with an error such as

```
Error 2010 , a MessageNotUnderstood occurred (error 2010), a Integer class does not
understand  #'_selectedPrimeGreaterThan:' (MessageNotUnderstood)
```

Workaround:

In $GEMSTONE/upgrade/integer.gs, comment out the lines

```
   removeallmethods Integer
   removeallclassmethods Integer
```

Bug 48459 - Page server processes left behind when remote cache times out

Product: GemStone/S 64 Bit

Versions: 3.5

Platform All Platforms

Fixed In: 3.5.1

Page server processes are left behind on the stone machine if the associated remote cache times out.   The processes are idle and not consuming CPU, but will tie up some system resources unless terminated.

Workaround:

Stopping and restarting the stone will terminate these processes.

Bug 48447 - find reference path search can miss references via class instance variables

Product: GemStone/S 64 Bit

Versions: 3.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.2, 3.3.1, 3.3, earlier versions

Fixed In: 3.5.1, 3.4.5

Finding the reference path to an object did not follow reference paths from class instance variables if the class itself was unreferenced. If the only reference to an search object was from a class instance variable, the search object could incorrectly be reported as unreachable/dead.

Bug 48425 - Remote log file naming using #log: NRS directive should include executable and pid

Product: GemStone/S 64 Bit

Versions: 3.6.3, 3.6.2, 3.6.1, 3.6, 3.5.7, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.2, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.x, 2.x, earlier versions

When logging in remotely, with the Gem on a node other than the Stone's node, a remote cache is started, and in additon to the Gem, a number of other processes are started up on the Gem and Stone nodes. If the NRS for the Gem includes the #log: directive, which specifies the name of the log file, this pattern is also used for the log files for these other processes.

If the #log: directive argument does not include the %N pattern (the executable, such as gemnetobject or pcmon), and the $P pattern (the PID of the process), then, depending on the version, the process log files may be located in the home directory, rather than the intended directory, have default names rather than names following the log pattern; and/or process log files may have a trailing descriptive, such as .log_pcmon.

Workaround:

When logging in remotely, ensure that any #log: directives in the Gem NRS or GEMSTONE_NRS_ALL includes both %P and %N.

Bug 48369 - Remote process support logs may not be written to preferred location

Product: GemStone/S 64 Bit

Versions: 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.2, 3.3.1, 3.3, earlier versions

Fixed In: 3.5.1

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, the NRS for the client process, and -D setting for startnetldi on the server and, in a remote gem configuration, the client node startnetldi -D. This allows flexibility to specify paths for the client and server log files from the respective NetLDIs, or from the client process.

In earlier versions, the expansion of the %D with the startnetldi -D argument was done early and could result in invalid log file locations on the server node.

v3.5 included changes, including -D becoming a default directory, and the precedence of settings was calculated differently. While Gem logs were located correctly for most configurations, 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).

Workaround:

Configuring the location of process log files may need adjustments, in the Gem's NRS, the GEMSTONE_NRS_ALL, and/or the -D arguments to startnetldi. Contact GemTalk Techincal Support for assistance with the options for a specific version.

Bug 48358 - Missing reports in traversals from GBS

Product: GemStone/S 64 Bit

Versions: 3.5

Fixed In: 3.5.1

Impact: Critical

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 reports, 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, causing the client object to have a state inconsistent with the server state.

Bug 48340 - Repository>>continuousRestoreFromArchiveLogs: does not work on extent copy

Product: GemStone/S 64 Bit

Versions: 3.5.7, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.2, 3.3.1, 3.3, 3.2.17, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.3, 3.1.0.2, 3.1.0.1, 3.1

Platform All Platforms

Fixed In: 3.6

Documentation in the System Administration Guide presents two approaches to initially configuring a slave system for hot standby:

1. Take extent copies of the primary, copy these to the slave system, and startstone -R -N before executing #continuousRestoreFromArchiveLogs:.

2. Take a fullBackup of the primary, copy the backup file(s) to the slave system, and restore the backup on the slave before executing #continuousRestoreFromArchiveLogs:.

For now, the first approach does does not work: the #continuousRestoreFromArchiveLogs: returns the expected output message and error (4048), but tranlogs *will not* be getting replayed as expected.

Workaround:

Start the slave system with an out-of-box extent0.dbf and restore a full backup from the primary before executing #continuousRestoreFromArchiveLogs:

Bug 48311 - Collection fanout in repositories that did not run postconv after conversion from 32-bit

Product: GemStone/S 64 Bit

Versions: 3.6.3, 3.6.2, 3.6.1, 3.6, 3.5.7, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.2, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.x, 2.x, earlier versions

Impact: Informational

Large collections in GemStone are represented internally in a tree structure, with nodes designed to fit on a GemStone page. Since the page size in GemStone/S 64 Bit is larger than the page size in 32-bit GemStone/S, the conversion process included a step, postconv, that converted the 32-bit collection fanout to the new 64-bit fanout.

If the postconv conversion step is skipped, the old fanout will continue to be used and is fully supported. However, there may be some performance penalty as more leaf node pages may need to be read to access elements of the collection.

Object audit will report a warning if it detects collections (other than NSCs) that are in the old fanout form.

```
Object Audit: Audit successfully completed; no errors were detected.
[WARNING] nnnn objects found with the oldLrgFanout
  These objects are saved in hiddenSet Reserved1 (26)
```

and could be optimized by converting to 64bit fanout

(the message may vary in later versions of GemStone/S 64 Bit)

If you do not have a problem with collection performance, the warning can be disregarded.

Workaround:

The collections can be rebuilt, by creating a new collection, adding each content object to the new collection, and doing a become:. The specific details will depend on the collection class.

Note that due to the different ways NSCs (such as IdentityBag) are constructed, it is not simple to distinguish an NSC with an old fanout from one that happens to be sparse.

Bug 48276 - Scan errors on repositories upgraded from early GemStone versions with sparse security policies

Product: GemStone/S 64 Bit

Versions: 3.5.1, 3.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4

Fixed In: 3.5.2, 3.4.5

Repositories that originated in early versions of GemStone, such as 32-bit GemStone/S, and that had less than 20 Segments (since relabled ObjectSecurityPolicies), were assigned an ObjectSecurityPolicy with ID 20 during upgrade to 2.x, to support a nil security policy. If there were less than 20 security policies in the original repository, the intermediate slots for object security policy IDs were nil.

Code changes in v3.4 for repository scan code did not handle nil slots for object security policies, and operations such as objectAudit and listInstances would error.

Workaround:

Creating dummy object security policies will avoid errors. For example:

```
1 to: SystemRepository size do: [:i |
   (SystemRepository at: i) isNil ifTrue: [GsObjectSecurityPolicy new.]].
```

Bug 48224 - Incorrect execution results due to compiler block optimization

Product: GemStone/S 64 Bit

Versions: 3.5, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.2, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, earlier versions

Fixed In: 3.5.1, 3.4.4

There is an error in the compiler's optimization of certain expressions including blocks and nil, such that this could return incorrect results.

For example, this is an incorrect result:

```
topaz 1> run
   | object |
   object := 5 .
   (object ifNil:[ nil ]) == object
%
false
```

Bug 48220 - Invalid results from System>>hostCpuUsage

Product: GemStone/S 64 Bit

Versions: 3.6.3, 3.6.2, 3.6.1, 3.6, 3.5.7, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.2, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, earlier versions

Fixed In: 3.5.1, 3.4.4

The method System>>hostCpuUsage returns an Array of integers describing the percent active, idle, user, system, and I/O wait.

In versions earlier than v3.4.4/v3.5.1, the return values on some platforms may have been entirely 0, or a set of values that did not add up to 100. This was due to incorrect rounding in the calculations.

Note that the first call to this within a gem, and back-to-back calls, provide invalid results (entirely idle CPU) on some platforms. On Linux, hostCpuUsage is based on reading files in /proc. For calls that are too close together, the kernel will not have updated the statistics information; you will need to wait 10ms-100ms between calls to get valid CPU usage numbers.

Bug 48214 - Commit Record Backlog and C heap growth during restore

Product: GemStone/S 64 Bit

Versions: 3.5, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.2, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, earlier versions

Fixed In: 3.5.1, 3.4.4

When restoring transaction logs, such as in a warm standby, where there is a large commit record in the tranlog being restored, the a commit record for the restore may take some time to be released. This can result in a commit record backlog.

The thread that is reading records continues to run, which causes excessive C heap growth in the stone process.

Bug 48211 - STN_FREE_SPACE_THRESHOLD > 4GB is truncated

Product: GemStone/S 64 Bit

Versions: 3.5, 3.4.3, 3.4.2, 3.4.1, 3.4

Platform All Platforms

Fixed In: 3.5.1, 3.4.4

Setting STN_FREE_SPACE_THRESHOLD to a value greater than 2^32-1 (4294967295) will result in the actual value used being truncated to the low-order 32 bits of the number, which will tend to be significantly lower than the intended value.

Workaround:

Limit the setting of this configuration parameter to 4 GB / 4294 MB / 4294967 KB or less.

Bug 48206 - Installs on Mac/Mojave can generate rpcinfo error

Product: GemStone/S 64 Bit

Versions: 3.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5

Platform Mac OS / Mojave

Impact: Informational

Under currently unknown conditions, 3.5 installs under Mac OS / Mojave can fail with the following error:

```
rpcinfo: can't contact portmapper: rpcinfo: RPC: Unable to send; errno = Broken pipe
```

Workaround:

Prior to installation, execute:

```
sudo launchctl start com.apple.rpcbind
```

Bug 48188 - copydbf -Z option is broken

Product: GemStone/S 64 Bit

Versions: 3.5, 3.4.3, 3.4.2, 3.4.1, 3.4

Fixed In: 3.5.1, 3.4.4

The copydbf option -Z, which copies using LZ4, is broken.

Workaround:

Use gzip compression

Bug 48148 - GciStoreTravDoTravRefs_ may error, impacting linked GBS sessions

Product: GemStone/S 64 Bit

Versions: 3.5

Fixed In: 3.5.1

Impact: Critical

GemBuilder for Smalltalk interacts with the server using specialized GCI calls, such as GciStoreTravDoTravRefs_, that perform several functions in one call. Due to code changes in GS64 v3.5, if the numNotReplicated buffer gets too large, GciStoreTravDoTravRefs_ fails with Error 2101, oop 0 does not exist.

As a result, linked GBS sessions are unreliable with GemStone/S 64 Bit v3.5 servers, particularly if many objects are replicated and there are many object changes.

Workaround:

Avoid using linked logins with GBS 8.4 and GS64 v3.5.

Bug 48098 - Tranlog full with STN_COMMITS_ASYNC may result in lost record

Product: GemStone/S 64 Bit

Versions: 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4

Fixed In: 3.5.1

Impact: Critical

With STN_COMMITS_ASYNC, if the tranlogs become full and the Stone has to wait for more space to become available, the tranlog record that was waiting to be may not actually get written, depending on the particular point in which the tranlog directories full conditions are encountered.

A later replay of the transaction logs may encounter errors, depending on the nature of the lost record, and be unusable.

Workaround:

Ensure that STN_COMMITS_ASYNC is set to false; this is enforced with v3.5 and 3.4.4.

It is always strongly recommended to avoid tranlog full conditions.

Bug 48011 - Dynamic instance variables are not migrated

Product: GemStone/S 64 Bit

Versions: 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.2, 3.3.1, 3.3

Fixed In: 3.5

The instance migration process by default migrates named instance variables. Dynamic instance variables were not migrated.

Workaround:

Implementing a customized method migrateFrom:instVarMap: in the new class version (the destination class) allows you to make any instance variable mappings required.

Bug 47988 - Change in results from System>>descriptionOfSession: on non-existant sessions

Product: GemStone/S 64 Bit

Versions: 3.6.3, 3.6.2, 3.6.1, 3.6, 3.5.7, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3

Platform All Platforms

Impact: Informational

For a session that does not exist, System>>descriptionOfSession: used to return an array of nils. The new version of System>>descriptionOfSession: now returns an array of nils, zeros, empty strings, and a false as follows:

```
 topaz 1> run
 System descriptionOfSession: 333
 %
 a Array
#1 nil
#2 0
#3 <empty string>
#4 0
#5 0
#6 0
#7 0
#8 false
#9 0
#10 0
#11 <empty string>
#12 0
#13 0
#14 0
#15 0
#16 0
#17 nil
#18 0
#19 0
#20 0
#21 0
```

```
 #22 nil
  topaz 1>
```

This change was made to provide additional information for a proto-session while still in the process of logging in, such as gems that have connected to the stone but have not yet registered their userProfile with the stone. You can now get the gem processId, gem hostIp, and other fields for such a session.

Element 10 (sessionId) of the result will be 0 if the session does not yet "officially" exist, or positive if the session is officially recognized by the stone.

Bug 47979 - CHeader has limitations in parsing header file syntax

Product: GemStone/S 64 Bit

Versions: 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.8, 3.3.7, 3.3.6, 2.2.9, older versions

Fixed In: 3.5

Some C header file syntax cannot be read by GemStone's CHeader class. This includes, but is not limited to, the syntax used in /usr/include/zlib.h and its include files, in recent versions of Linux. These header files cannot be processed by CHeader.

An an error such as this may be reported.
   unexpected token: #'identifier' -> 'va' near line 1758 in file /usr/include/zlib.h (Error)

Workaround:

Using the native C preprocessor on the header files before using CHeader can avoid issues. This preprocessing is built into CHeader for versions in which this bug is fixed.

Bug 47967 - GCI applications on Ubuntu 18

Product: GemStone/S 64 Bit

Versions: 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.2, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, earlier versions

Platform Linux/Ubuntu 18 only

Fixed In: 3.5, 3.4.4

Impact: Informational

Changes in gcc with Ubuntu 18.04 created issues with building GemStone GCI applications.

While earlier versions of GemStone run without known problems on 18.04, these configurations have not been certified. Building GCI applications on or for Ubuntu 18.04 is not supported with v3.4.3 and earlier.

Bug 47916 - Logsender padding when compressing tranlog records could cause buffer overrun

Product: GemStone/S 64 Bit

Versions: 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.2, 3.3.1, 3.3

Fixed In: 3.5, 3.4.4

The logsender does compression of the tranlog records (by default) prior to sending them to the logrecevier in a hot standby configuration. The logsender was adding unnecessary padding to these compressed records. In building the buffer for transmittal, the cumulative padding could cause the buffer size to be exceeded.

Workaround:

Disable compression by using the startlogsender -u argument.

Bug 47891 - Symbol Garbage Collection not working reliably

Product: GemStone/S 64 Bit

Versions: 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.2, 3.3.1, 3.3

Fixed In: 3.5, 3.4.4

The process that implements the garbage collection of unreferenced Symbols includes several steps, to ensure that there is no risk of incorrectly garbage collecting a symbol that is in use. One of these steps could inadvertently commit interior nodes of the collection that held references to the possible dead symbols, which meant they were not unreferenced and so where not actually garbage collected.

Workaround:

While there may be a slight overhead, GemStone will work correctly with unused symbols. Avoiding the creation of unnecessary symbols remains best practice, as in previous versions.

Bug 47885 - Statmonitor automatic start failed with -X

Product: GemStone/S 64 Bit

Versions: 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.2, 3.3.1, 3.3

Fixed In: 3.5, 3.4.4

Starting up statmonitor with the -X provides host-only monitoring. With the -X, no stone or cache name is permitted as an argument.

Using a -X in the configuration parameters that automatically start statmonitor, such as STN_STATMONITOR_ARGS, caused an error since these methods automatically include the stone or cache name as an argument.

Workaround:

As a workaround, $GEMSTONE/sys/runstatmonitor can be edited to add these lines, which deletes the stonename from the command line args if the second arg is -X

```
#Workaround bug 47885

if [ "x$2" = "x-X" ]; then
  shift
fi
```

Bug 47884 - GcGems not running after upgrade

Product: GemStone/S 64 Bit

Versions: 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.2, 3.3.1, 3.3

Fixed In: 3.5, 3.4.4

When the stone is started with the new executables during upgrade, the Stone attempts to start the Reclaim and Admin GcGem. These gems are not yet able to login due to the version mismatch. The Stone did not reattempt to start the GcGems later.

Workaround:

After upgrade is complete, shut down and restart the Stone. You may also explicitly restart the GcGems using System startAllGcGems.

Bug 47878 - GsSysLog writeErrorMessage fails with ArgumentError

Product: GemStone/S 64 Bit

Versions: 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.2, 3.3.1, 3.3

Platform All Platforms

Fixed In: 3.5, 3.4.4

Execution of the method GsSysLog writeErrorMessage fails with:

```
-----------------------------------------------------
GemStone: Error        Nonfatal
a ArgumentError occurred (error 2718), Invalid priority, LOG_ERROR Error Category: 231169 [GemStone] Number:
2718  Arg Count: 2 Context : 36416769 exception : 36413441
Arg 1: [20971265 sz:9 cls: 110849 Symbol] LOG_ERROR
Arg 2: [20 sz:0 cls: 76289 UndefinedObject] nil
```

Workaround:

Edit code in $GEMSTONE/upgrade/GsSysLog.gs for GsSysLog writeErrorMessage and replace #LOG_ERROR with #LOG_ERR. File-in as SystemUser and commit.

Bug 47875 - Backups may run in transaction, causing CR backlog

Product: GemStone/S 64 Bit

Versions: 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.4, 2.4.x, 2.3.1.6, 2.3.x, 2.2.5.4, 2.2.x, 2.x

Platform All Platforms

Fixed In: 3.1

Normally, when doing a fullBackup*, the session is initially placed in auto transaction mode and in-transaction while some preliminary structures and shadowed pages are written to the backup file. The session then switches to manual transaction mode and outside of transaction for the rest of the backup while writing out the data pages.

But if there are no shadowed pages (indicated by VSD stat PagesNeedReclaimSize equal to zero), there is a small chance that the session will stay in auto transaction mode and in-transaction during the entire backup, which could cause problems with CR backlogs.

Also note that if you take a backup while the stone is still in restore mode (before executing the commitRestore), then the session will also stay in auto transaction mode and in-transaction during the entire backup

Workaround:

1. Don't take a backup while in restore mode.

2. Shutdown the reclaim GcGem before starting the backup. You can restart the reclaim GcGem when you see the VSD stat ProgressCount increasing.

Bug 47868 - GsExternalSession fails with gemNRS: string argument

Product: GemStone/S 64 Bit

Versions: 3.4.3, 3.4.2

Fixed In: 3.5, 3.4.4

GsExteranalSession login parameters can accept an instance of GsNetworkResourceString or an String containing NRS as an argument. The use of strings for the gemNRS: is broken.

Workaround:

File in the following as SystemUser, and commit.

```
category: 'Parameters'
method: GsExternalSession
gemNRS: anNRS
  "Set the GemService parameters for the logon to the value
   of anNRS, which may be a String or a GsNetworkResourceString instance."

  parameters gemService: anNRS asString .
  (anNRS isKindOf: GsNetworkResourceString) ifTrue:[
    self dynamicInstVarAt: #gemHost put: anNRS node .
  ].
%
```

Bug 47846 - Utf8 >> withAll: disallowed with ByteArray arguments

Product: GemStone/S 64 Bit

Versions: 3.6.3, 3.6.2, 3.6.1, 3.6, 3.5.7, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.3.9

Impact: Informational

The method Utf8>>withAll: has been added, which restricts the composition of Utf8 instances to avoid malformed Utf8. However, applications that rely on ByteArray inputs containing Utf8 characters may have used withAll: as part of the process of decoding the input, which errors. This was not the intended usage of Utf8.

Workaround:

To convert a ByteArray containing UTF-8 encoded text into a string, in v3.5 and later, you can use the methods:
    ByteArray >> decodeFromUTF8ToString
    ByteArray >> decodeFromUTF8ToUnicode

This replaces patterns such as:
  (Utf8 withAll: <aByteArrayContainingEncodings>) decodeToString

Bug 47844 - Logreceiver race condition can cause slave stone tranlog read failure

Product: GemStone/S 64 Bit

Versions: 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.2, 3.3.1, 3.3, 3.2.17, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.x

Platform All Platforms

Fixed In: 3.5, 3.4.4

There is a race condition in the logreceiver logic that can cause the slave stone to attempt to read from a newly transferred tranlog before any data has been permanently flushed to disk. This will generate errors that look like this (for tranlog ID XXXXX)

```
  WARNING: for fileId XXXXX,
    filename = <tranlog_directory>/tranlogXXXXX.dbf
  This file is not a GemStone transaction log file:
```

error in read() , <tranlog_directory>/tranlogXXXXX.dbf failed with errno=9,EBADF, Bad file number (file is not open, or cannot be read/written)

Workaround:

You will need to restart the "SystemRepository continuousRestoreFromArchiveLogs:*" command on the slave stone.

Bug 47823 - 3.4 and 3.4.1 shared libraries not compatible with 3.4.2

Product: GemStone/S 64 Bit

Versions: 3.4.2

Fixed In: 3.4.3

Impact: Informational

GemTalk officially requires that clients use the same version of the gci shared libraries as the stone version. However, often shared libraries remain compatible between server versions.

With v3.4.2, using v3.4 or 3.4.1 shared libraries on Windows appears to work, but GsFile operations hang.

Workaround:

Ensure you are using the correct versions of the client libraries.

Bug 47817 - Kiling a remote shared page cache risks bringing the Stone down

Product: GemStone/S 64 Bit

Versions: 3.4.2, 3.4.1, 3.4

Fixed In: 3.4.3

If a remote shared page cache is killed using SIGTERM (which normally performs a clean shutdown), there are code paths in which it does not correctly wait for all threads to detach from the Stone's cache. This results in the Stone exit.

Workaround:

The risk of hitting this bug is proportional to the number of active gems running on the cache at the time of the kill. To minimize the risk, terminate all the gems first, before killing the cache.

Bug 47816 - Stuck session in login cannot be cleaned up

Product: GemStone/S 64 Bit

Versions: 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.x, earlier versions

Fixed In: 3.4.3

If a UserProfile is configured to record login times, and some problem occurs during login such that this session stops processing (a stuck session), it may leave the UserSecurityData for this session locked. This problem cannot be cleared up using the normal stopSession: and stopZombieSession: methods.

Bug 47806 - listInstances:limit: does not apply limit if there are instances in memory

Product: GemStone/S 64 Bit

Versions: 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.2, 3.3.1, 3.3.x

Fixed In: 3.4.3

Repository>>listInstances:limit: finds both persistent instances and instances in memory, returning no more results than the limit specified. However, this limit was not being applied to instances in memory, so the results could exceed the limit.

Bug 47801 - fullBackupCompressed file size limit computed based on uncompressed size

Product: GemStone/S 64 Bit

Versions: 3.4.2, 3.4.1, 3.4

Fixed In: 3.4.3

The MBytes: limit argument to multiple file fullBackupCompresed is applied to the uncompressed data size. This results in smaller files being written for all files except the last, and the large remainder of the backup being written to the single final file.

Workaround:

Increase the MBytes limit.

Bug 47797 - Reclaim configuration parameters revert to speed-optimized values

Product: GemStone/S 64 Bit

Versions: 3.4.2, 3.4.1, 3.4, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.2, 3.3.1, 3.3

Fixed In: 3.4.3, 3.3.9

During reclaimAll or finalization after other garbage collection operations, some Reclaim GcGem configurations are set to values that optimize performance by using all system resources. This is intentional; however, these are not being reset to the application-defined settings afterwards.

The following parameters are affected:
  #deadObjsReclaimedCommitThreshold
  #objsMovedPerCommitThreshold
  #sleepTimeBetweenReclaimMs
  #sleepTimeWithCrBacklogMs
  #reclaimDeadEnabled

As of v3.3.x, the values in the GcUser's UserGlobals are used at startup only; runtime changes to these values are done using methods in System class.

Workaround:

After reclaimAll, MFC, or Epoch, manually reset these values by executing:
   System setReclaimConfig: configName toValue: theValue

If the desired parameter values are set in the GcUser's UserGlobals, you may also restart the ReclaimGem.

Bug 47796 - Does not understand #_selectiveAbort in #_resolveRcConflictsWith: during commit

Product: GemStone/S 64 Bit

Versions: 3.4.2, 3.4.1, 3.4

Platform All Platforms

Fixed In: 3.4.3

There is a problem with RC commit retry processing that can cause a doesNotUnderstand error on the method #_selectiveAbort in #_resolveRcConflictsWith:. The call stack will be similar to this:

```
  (Object) >> _doesNotUnderstand:args:envId:reason: @8 line 13
 [] in RcKeyValueDictionary >> _resolveRcConflictsWith: @37 line 32
 IdentityKeyValueDictionary (KeyValueDictionary) >> keysAndValuesDo: @19 line 15
 RcIndexDictionary (RcKeyValueDictionary) >> _resolveRcConflictsWith: @21 line 31
 System class >> _resolveRcConflicts @17 line 19
 System class >> _resolveRcConflictsForCommit: @4 line
 [] in System class >> _localCommit: @28 line 42
 ExecBlock0 (ExecBlock) >> onException:do: @2 line 66
 System class >> _localCommit: @16 line 44
 GsCommitList >> nbCommit: @12 line 17
 GsCommitList >> commit: @4 line 7
 System class >> _commit: @8 line 16
 System class >> commitTransaction @5 line 7
```

Workaround:

File-in the following code as SystemUser and commit:

```
! fix 47796
! file-in as SystemUser and commit
!
method: PrivateObject
_primitiveSelectiveAbort
"Performs an abort operation on the receiver. That is, if the object is
 committed, it removes any changes made by the current transaction and allows
 access to the committed state of the object.

 The error #rtErrSelectiveAbort is thrown if the receiver has depMap entries."



self _primitiveFailed: #_primitiveSelectiveAbort .
self _uncontinuableError
%

method: PrivateObject
_selectiveAbort

"Performs an abort operation on the receiver. That is, if the object is
 committed, it removes any changes made by the current transaction and allows
 access to the committed state of the object."

^ self _primitiveSelectiveAbort.
%
```

Bug 47792 - findAllReferencePathsToObject: does not reliably scale

Product: GemStone/S 64 Bit

Versions: 3.5.7, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.2, 3.3.1, 3.3

Impact: Informational

In larger repositories with more complex object graphs, findAllReferencePathsToObject: may not return results within a reasonable amount of time, in spite of the improvements in memory requirements in 3.3. There are some sets of conditions under which the implementation of find*References paths cannot scale.

The "findAll" methods have been deprecated in v3.4 and later. findReferencePath* can be used serially to find references.

Bug 47768 - Statmonitor subsecond sampling generates incorrect timestamps

Product: GemStone/S 64 Bit

Versions: 3.4.2, 3.4.1, 3.4

Fixed In: 3.4.3

In most cases, the timestamps that statmonitor collects from GemStone are rounded to the nearest whole number of seconds before being recorded. This allows them to be easily understood with the VSD display.

With sub-second timestamp intervals, however, the seconds field should be truncated, not rounded. This rounding resulted in timestamps that were incorrect by up to 1 second.

Since VSD skips out of order timestamps, the charts displayed by VSD did not include all samples.

Bug 47753 - Very large GEM_MAX_SMALLTALK_STACK_DEPTH can result in SEGV on AlmostOutOfStack on x86_64

Product: GemStone/S 64 Bit

Versions: 3.4.2, 3.4.1, 3.4, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.x

Fixed In: 3.4.3, 3.3.9

With a very large GEM_MAX_SMALLTALK_STACK_DEPTH for Gems running on x86_64 processors (Linux and Solaris/X86), when an AlmostOutOfStack occurs, the Gem cashes with a SEGV.

The SEGV is in stack memory due to the native code's use of 8 bytes of C stack per Smalltalk frame.

Workaround:

Set GEM_MAX_SMALLTALK_STACK_DEPTH to 30000 or less.

Bug 47733 - Topaz 64-bit on Windows, printing of 64 bit integers is incorrect

Product: GemStone/S 64 Bit

Versions: 3.4.2, 3.4.1, 3.4, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2

Fixed In: 3.4.3, 3.3.9

GemStone distributes a 64-bit topaz executable, as well as the 32-bit topax executable, on Windows clients in v3.2 and later. In the 64-bit topaz, but not in the 32-bit, printing of integers is incorrect.

For example:

```
topaz 1> run 3715007610
%
-579959686
```

Workaround:

Use the 32-bit topaz, in %GEMSTONE%/bin32/topaz.exe

Bug 47732 - Problems with copydbf -i with files on NFS-mounted disks

Product: GemStone/S 64 Bit

Versions: 3.4.2, 3.4.1, 3.4, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.2, 3.3.1, 3.3

Fixed In: 3.4.3, 3.3.9

With a backup .dbf file on an NFS-mounted disk, copydbf -i could return an error such as an illegal page kind, or report no results. The backup itself may still be valid and restore correctly.

Bug 47720 - Filenames with non-ASCII characters not handled by Windows client

Product: GemStone/S 64 Bit

Versions: 3.4.2, 3.4.1, 3.4, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.2, 3.3.1, 3.3, 3.2.17, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.x, 3.2, 3.x

Platform Windows client

Fixed In: 3.4.3, 3.3.9

When a filename on Windows contains characters with codePoints over 127, GsFile operations to open the file on the client return nil. Other GsFile operations are also expected to fail on Windows with filenames with codePoints over 127.

Note that a number of related issues with filenames containing characters with codePoints over 127 have been addressed in later versions of GemStone.

Bug 47718 - CIFS filesystems on Linux do not work

Product: GemStone/S 64 Bit

Versions: 3.6.3, 3.6.2, 3.6.1, 3.6, 3.5.7, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.2, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, earlier versions

Impact: Informational

GemStone extents cannot be created on a Linux CIFS file system. This results in an Stone crash in earlier versions, and has been disallowed in more recent versions.

Bug 47712 - Sign of UTC offset in header is incorrect and should be inverted

Product: GemStone/S 64 Bit

Versions: 3.4.2

Platform All Platforms

Fixed In: 3.4.3

Information in the header on the start time/date of a GemStone process now includes UTC offset information. For example, from a topaz session header started in the Pacific Daylight Time zone:

```
PROCESS ID: 18591     DATE: 08/28/2018 11:20:20 PDT  (UTC +7:00)
```

However, the sign of the offset is incorrect and should be inverted. This example should read:

```
PROCESS ID: 18591     DATE: 08/28/2018 11:20:20 PDT  (UTC -7:00)
```

Workaround:

Invert the sign of the UTC offset.

Bug 47681 - Object and page audit errors after restore of NonTranloggedGlobals

Product: GemStone/S 64 Bit

Versions: 3.4.2, 3.4.1, 3.4

Fixed In: 3.4.3

When a repository is using NotTranloggedGlobals, and performs a backup and restore from logs, there are some circumstances in which OOPs end up neither allocated nor on the free OOP list.

This causes object and page audit to report errors:

objectAudit:
    The object [nnnn] does not exist, but is neither dead nor in the FreeOop List.

pageAudit:
    Error: Found oops that do not exist and are not in the root page freeOops

These OOPs will not get used and are effectively lost; otherwise, this error is benign.

The exact circumstances that produce this issue are not yet determined.

Workaround:

Making a programmatic backup and restoring it rebuilds the free oop list and fixes the problem.

Bug 47679 - NonTranloggedGlobals can cause tranlog restore to fail

Product: GemStone/S 64 Bit

Versions: 3.4.1, 3.4

Fixed In: 3.4.2

Impact: Critical

Under certain sequences of operations, when using NonTranloggedGlobals when the oopHighWaterMark is increasing, the oopHighWaterMark is not updated corrrectly. This does not cause the restoreFromLogs to fail, however, the stone cannot subsequently start up.

If the repository's oopHighWaterMark is sufficiently high that new oops are not allocated, this bug is not exposed.

Bug 47677 - Very large gem TOC disables native code

Product: GemStone/S 64 Bit

Versions: 3.6.2, 3.6.1, 3.6, 3.5.7, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.2, 3.3.1, 3.3, 3.2.17, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.x, 3.2, 3.1.x

Platform Linux

Impact: Informational

Using a very large setting for GEM_TEMPOBJ_CACHE_SIZE, 10GB or larger, results in issues for the native code

generator, which reports the warning message:
native code DISABLED, offset from code memory to libgcilnk.so exceeds 32bits

In 3.2.x, this caused logins to fail with the error
Fatal error in native code generation, emit_call_stub, jmp displacement exceeds 32bits Error Category: [GemStone] Number: 4151 Arg Count: 1

Workaround:

In 3.2.x, reduce the GEM_TEMPOBJ_CACHE_SIZE to less than 10GB, or disable native code generation by setting GEM_NATIVE_CODE_ENABLED = 0;

The warnings in 3.3.x and later can be ignored, although there may be performance differences in the absence of native code generation.

Bug 47663 - GsExternalSession logout errors with multiple stones of the same name

Product: GemStone/S 64 Bit

Versions: 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.2, 3.3.1, 3.3

Fixed In: 3.5, 3.4.4

The internal method GsExternalSession _isOnMyStone incorrectly assumed unique stone names. If the external session is logged into a stone on a different node that has the same name as a stone that is also running on the Gem's node, after the wait for logout would timeout.

Bug 47659 - Nested Transactions can create references to non existent objects

Product: GemStone/S 64 Bit

Versions: 3.4.1, 3.4, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.2, 3.3.1, 3.3, 3.2.17, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.x, 3.2, 3.1.x

Fixed In: 3.4.2

Impact: Critical

There is a sequence of processing within a nested transaction that has the ability to create a committed reference to an object that does not exist.

This involves a nested transaction followed by a failed commit of the outer transaction; then an abort and another commit.

The problem is related to objects that were not correctly sorted between the closures of the inner and outer commits, such that state was not correctly handled by the abort.

Bug 47647 - Stone runs hot when RcTransQueueSize is high

Product: GemStone/S 64 Bit

Versions: 3.4.1, 3.4

Platform All Platforms

Fixed In: 3.4.2

The stone will consume excessive amounts of CPU when handling a large number of sessions doing commits involving RC (reduced conflict) objects during the same time window.

This can be detected by noting a high value for statmonitor/vsd stat UserTime along with stat RcTransQueueSize > 50.

This may result in degraded stone performance during this time window, but there are no other ill effects.

Bug 47646 - Stone memory leak in commit record bitmap caching

Product: GemStone/S 64 Bit

Versions: 3.4.1, 3.4

Platform All Platforms

Fixed In: 3.4.2

There is a slow C heap memory leak in the handling of bitmap caching for commit records. This can be observed by monitoring the stone statistic BmCHeapPages. The rate of increase is proportional to the number of commits performed (stone statistic TotalCommits).

Workaround:

Disable the mechanism by setting in the stone config file:

```
STN_COMMIT_RECORD_BM_CACHING = FALSE;
```

Impacts on the setting are outlined in the config file comment:

```
  # STN_COMMIT_RECORD_BM_CACHING
  #  When true, enables caching at the commit point of page allocation
  #  information needed when disposing a commit record.
  #  This can reduce I/Os during commit record dispose when the commit record
  #  backlog is high, or when there is a lot of page preemption occurring
  #  in the shared cache.
  #  When enabled, the maximum commit rate is slightly lower
  #  and commit latency is slightly higher because more work is done in
  #  the commit critical region in stone.
```

Bug 47636 - Backups taken under tranlog0.dbf after using /dev/null cannot be restored

Product: GemStone/S 64 Bit

Versions: 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.2, 3.3.1, 3.3

Platform All Platforms

Fixed In: 3.4.2

Normally GemStone starts with tranlog1.dbf as the first tranlog in a series. But after using /dev/null for tranlogs and then switching back to normal tranlogs, the first log will be tranlog0.dbf.

If a backup is taken while tranlog0.dbf is in effect, you will be unable to restore this backup. When subsequently attempting to restore tranlogs or commit the restore, the system will respond as though the restoreFromBackups: had never occurred:

```
  topaz 1> run
  SystemRepository restoreFromBackup: 'backupOnT0.dat'
  %
  finshed restore commitRestore
  --- MM/DD/2018 HR:MN:SC.XXX Logging out
```

```
The restore from backup completed, with XXXXXXX objects restored.
 Ready for restore from transaction log(s). (error 4046)
topaz> login
....
topaz 1> run
SystemRepository restoreStatus
%
Restore is not active.
topaz 1> run
SystemRepository restoreFromArchiveLogs: '/my/archive/tlogs'
%
 -----------------------------------------------------
GemStone: Error Nonfatal
a ImproperOperation occurred (error 2717), , You cannot restoreLogs without first executing restoreFromBackup
topaz 1>
```

Workaround:

Increment the tranlog ID by executing "SystemRepository startNewLog" before taking the backup.

Bug 47630 - GsHostProcess file descriptor leak

Product: GemStone/S 64 Bit

Versions: 3.4.1, 3.4, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.2, 3.3.1, 3.3

Fixed In: 3.4.2, 3.3.7

GsHostProcess creates three pipes. The finalization of in-memory instances of GsHostProcess did not properly close these pipes.

Bug 47612 - Problems updating IcuLibraryVersion

Product: GemStone/S 64 Bit

Versions: 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.2, 3.3.1

Platform All Platforms

Impact: Informational

The GS/64 Programming Guide, in Chapter 5 under heading "Updating IcuLibraryVersion", gives the following instructions:

2. Login as SystemUser and execute
   Globals at: #IcuLibraryVersion put: newVersionString
   Commit and logout.

3. Login as DataCurator, ...

But when attempting to login as DataCurator during step 3, the login will fail with the following error:

```
-----------------------------------------------------
GemStone: Error        Fatal
Error during login: IcuLibraryVersion version <new version> , first login to stone used version <old version> Error
Category: 231169 [GemStone] Number: 4147  Arg Count: 1 Context : 20 exception : 20 Arg 1: 20

Login failed due to errors.
```

To avoid this error, shutdown and restart the stone after step 2.

Workaround:

Shutdown and restart the stone after changing (Globals at: #IcuLibraryVersion) and committing during step 2.

Bug 47583 - listInstancesInPageOrder:toFile: reports out of memory

Product: GemStone/S 64 Bit

Versions: 3.4.1, 3.4, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.2, 3.3.1, 3.3, 3.2.17, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.x, 3.2

Fixed In: 3.4.2, 3.3.7

When running Repository >> listInstancesInPageOrder:toFile:, and there are no instance of the class for which listInstances is being run, the operation reports a memory error:

a AlmostOutOfMemory occurred (notification 2208), Host memory allocation failed; there is insufficient swap space and/or RAM.

Bug 47565 - Exclusive lock failure can cause tranlog space full error

Product: GemStone/S 64 Bit

Versions: 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.2, 3.3.1, 3.3.x

Fixed In: 3.3.7

Creating a new tranlog involves multiple steps:

1) file created with open(fn, O_CREAT|O_TRUNC|O_RDWR)
2) file closed
3) file opened again with open(fn, O_RDWR)
4) file write lock requested using fcntl(fd, SETFL, ...)

Between steps 2 and 3, it is possible for another process to access the tranlog file. In this case, in step 4 the lock request fails with the error EBADF/exclusive open: File is open by another process.

The handling of this error incorrectly reports the problem as tranlog space is full.

Bug 47540 - copydbf reporting GemStone version information with version mistmatch may fail or SEGV

Product: GemStone/S 64 Bit

Versions: 3.4.1, 3.4, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.2, 3.3.1, 3.3

Fixed In: 3.4.2, 3.3.9

copydbf -i reports the version of GemStone that was used to create the backup, and always reports this information correctly for a backup created with the same version of GemStone (that is, the executable copydbf is from the same version of GemStone as the backup).

However, when using copydbf in versions 3.3.x and 3.4.x with older backups, swizzling errors were reported and no information was available. With 3.4.x copydbf and a 2.4.x backup, the copydbf SEGVed.

Bug 47509 - Incorrect results for BtreePlus indexed range query

Product: GemStone/S 64 Bit

Versions: 3.4.2, 3.4.1, 3.4

Fixed In: 3.4.3

When using the new indexing infrastructure, the BtreePlus index, and making a range query of the form a <= value <= b, the incorrect operator was being applies for one of the two logical comparisons. This resulted in incorrect results.

Bug 47499 - Stone crashes with semaphore problems on Linux / systemd

Product: GemStone/S 64 Bit

Versions: 3.6.3, 3.6.2, 3.6.1, 3.6, 3.5.7, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.2, 3.3.1, 3.3, 3.2.17, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.x, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.4, 2.4.x, 2.3.1.6, 2.3.x, 2.2.5.4, 2.2.x, earlier versions

Platform Linux using systemd

Impact: Informational

Customers running Linux using the systemd software suite may be subject to GemStone system crashes caused when the semaphore array used by GemStone for interprocess communication is prematurely deleted. The actual errors will vary according to which GemStone process accesses the semaphore array first, but can include:

In the stone log:

```
  ERROR:Semaphore failure errno 43 (stnSemIdx = 240)
  SHUTDOWN command was generated by Stone failed in goToSleep, semaphore failed.
  [followed by normal stone shutdown messages]
```

In a gem log:

```
  Shared memory SemClear idx 240 failure:  for lock: noLock
  [followed by catastrophic error handling messages from HostCallDebugger]
```

During the subsequent system shutdown, the following error will appear in the SPC monitor log as it attempts to remove the already-deleted semaphore array:

```
   | semctl(131072, 0, IPC_RMID, ...) failed with errno=22,EINVAL, Invalid argument
   | (programmer error)                                              |
   |  Gemstone could not destroy the array of semaphores.
```

****

This problem occurs because the systemd suite is configured to automatically delete any semaphore arrays created by a unix session when that session logs out. So any stone started by that unix session will crash when it logs out.

To disable this mechanism, modify the systemd logind.conf file to set RemoveIPC under [Login] to 'no' (note that the default setting is 'yes'):

From /etc/systemd/logind.conf:

```
[Login]
#NAutoVTs=6
#ReserveVT=6
#KillUserProcesses=no
. . .
#IdleAction=ignore
#IdleActionSec=30min
#RuntimeDirectorySize=10%
RemoveIPC=no
#UserTasksMax=12288
```

Workaround:

Reconfigure systemd as described earlier.

Bug 47452 - Adds/removes from RcLowMaintenanceIdentityBags can be slow

Product: GemStone/S 64 Bit

Versions: 3.4.1, 3.4

Platform All Platforms

Fixed In: 3.5, 3.4.2

Adding and removing elements from an RcLowMaintenanceIdentityBag can be much slower than adding/removing from an RcIdentityBag.

Workaround:

Use an RcIdentityBag until after upgrading to a version containing the fix for this bug (3.4.2, 3.5).

Bug 47445 - RcQueue may rarely and incorrectly return a nil element

Product: GemStone/S 64 Bit

Versions: 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 2.4.8, 2.4.7, earlier versions

Fixed In: 3.4.2

There is a sequence of events in which an RcQueue may return a nil rather than the next element in the queue. No data is lost in this scenario.

The sequence required to get this case is where a session is performing a cleanupQueue and encounters a concurrency conflict against another session that is performing a read. The cleanup session aborts and retries the cleanupQueue, which succeeds. However, the cleanup session has written the removalSequence numbers when it should not, and a read by the other session may now get a nil from the RcQueue.

Bug 47423 - Login after restoring backup fails with ICU library error

Product: GemStone/S 64 Bit

Versions: 3.6.3, 3.6.2, 3.6.1, 3.6, 3.5.7, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1.2, 3.5.1.1, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4

Platform All Platforms

Impact: Informational

If you are restoring a backup using a linked topaz session, and the IcuLibraryVersion of your backup file is different from that of the extent you are restoring into (usually the initial extent0.dbf), an extra step is required to complete the restore.

When you first logged in the topaz session, before the restore, the topaz process loaded the ICU library version specified in the initial, empty extent. When the restoreFromBackup completes, the session logs out. However, if the restored backup specifies a different ICU library version, login will fail with an error similar to the following, since the linked topaz process still has the original ICU library version loaded.

```
[ERROR] ICU library error, this process previously loaded 58.2, now IcuLibraryVersion=54.1
-----------------------------------------------------
GemStone: Error Fatal
StrPrimInit failed, ICU library error, this process previously loaded 58.2, now IcuLibraryVersion=54.1

Error Category: [GemStone] Number: 4149 Arg Count: 1 Arg 1: nil
```

Workaround:

Exit the linked topaz and start topaz again.

Bug 47409 - performOnServer: blocked SIGTERM

Product: GemStone/S 64 Bit

Versions: 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8, 3.2.7, earlier versions

Fixed In: 3.4.2

During the vfork() call invoked by System class >> performOnServer:, all signals are blocked (this was done to avoid a deadlock). If a third party software package relies on these signals, it will not operate correctly when invoked from GemStone Smalltalk code.

In versions in which this bug is fixed, SIGTERM is no longer blocked, though other signals still are. If other signals are required, contect GemTalk Techincal Support to discuss.

Bug 47401 - Unable to perform a partial restore of a repository with minor corruption/objects that do not exist

Product: GemStone/S 64 Bit

Versions: 3.4, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 2.4.8, 2.4.7, earlier versions

Fixed In: 3.4.1, 3.3.7

Impact: Informational

f a backup file has corruption which prevents some objects from being restored, the restore terminates with a non-fatal error and the backup cannot be restored. While it is preferable to only restore non-corrupted backups, there are cases in which being able to restore with some missing or bad objects is valuable.

In v3.4.1, the restore tolerates some corrupted records and continues to complete the restore. On completion of a non-clean restore, the restore reports the error BKUP_ERR_RESTORE_FAILURE/4152. To continue with the restore, continue with restoring transaction logs and execute commitRestore. To determine the nature and extent of the problems, perform an objectAudit immediately after the commitRestore.

Bug 47396 - GsBitmap writeToFile: on an empty bitmap triggers AlmostOutOfMemory error

Product: GemStone/S 64 Bit

Versions: 3.4

Platform All Platforms

Fixed In: 3.5, 3.4.1

Using method GsBitmap>>writeToFile: on an empty bitmap triggers an AlmostOutOfMemory error. For example:

```
run
    GsBitmap new writeToFile: <filename>
%
ERROR 2208 , a AlmostOutOfMemory occurred (notification 2208), Host memory allocation failed; there is insufficient swap space and/or RAM. (AlmostOutOfMemory)
```

Workaround:

Check GsBitmap>>size and if zero, skip the writeToFile:.

Bug 47387 - #instancesNonPersistent and other class options may be "sticky"

Product: GemStone/S 64 Bit

Versions: 3.6.3, 3.6.2, 3.6.1, 3.6, 3.5.7, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, earlier versions

Impact: Informational

#instancesNonPersistent can be used as an element in the options: argument of a subclass creation method. New instances of the class are then created such as they cannot be made persistent; committing a reference to them from the repository is disallowed. There are a number of other similar options that affect instance behavior, including #subclassesDisallowed, #disallowGciStore, #traverseByCallback, #dbTransient, and #instancesInvariant.

These options are automatically inherited by subclasses, new versions, and after executing subclass creation methods that do not version the class, but modify the existing class as a side-effect. Attempting to remove an option by defining a new version of the class without that options symbol may not have the desired effect.

If you wish to turn off #instancesNonPersistent or other options, you must specify #noInheritOptions as the first element in the options: argument array when executing the subclass definition method that either redefines the class or defines the subclass.

Bug 47364 - Security error when browsing implementors with session methods enabled and no CodeModification privilege

Product: GemStone/S 64 Bit

Versions: 3.4, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.2, 3.3.1, 3.3, 3.2.17, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.3, 3.1.0.2, 3.1.0.1, 3.1, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.4, 2.4.x, 2.3.1.6, 2.3.x, 2.2.5.4, 2.2.x

Platform All Platforms

Fixed In: 3.4.1, 3.3.7

If session methods have been enabled for a UserProfile that does not have CodeModification privilege, if this user attempts to browse implementors of any method, using either GBS or ClassOrganizer>>implementorsOf:, it will trigger the following error:

```
-----------------------------------------------------
GemStone: Error Nonfatal
a SecurityError occurred (error 2151), reason:rtErrNoPriv, An attempt was made to do a privileged operation for which no privilege had been granted. Error Category: 231169 [GemStone] Number: 2151 Arg Count: 1 Context : 32046593 exception : 32044545
Arg 1: [32040705 sz:10 cls: 76545 UserProfile] anUserProfile
```

For GS version 2.X the error will be formatted slightly differently:

```
-----------------------------------------------------
GemStone: Error Nonfatal
An attempt was made to do a privileged operation for which no privilege had been granted. Error Category: 231169 [GemStone] Number: 2151 Arg Count: 1 Context : 15536641
Arg 1: [15045121 sz:10 cls: 76545 UserProfile] an UserProfile
```

Workaround:

Browse implementors from a GS user with appropriate privileges, or that has not had session methods enabled.

Bug 47344 - Abort may not advance view if not in transaction and many CRs behind

Product: GemStone/S 64 Bit

Versions: 3.4, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.2, 3.3.1, 3.3, 3.2.17, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.x, 3.2, 3.1.x, 3.0.x

Fixed In: 3.4.1, 3.3.7

If a session is outside of transaction and many commitRecords beind the current commitRecord, a System >> abortTransaction does not advance the session's view.

Bug 47325 - GsFile contentsAndTypesOfDirectory: does not handle UTF8 filenames correctly

Product: GemStone/S 64 Bit

Versions: 3.4, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3

Fixed In: 3.4.1, 3.3.9

The methods GsFile class>>contentsAndTypesOfDirectory:onClient: and GsFile class>>contentsOfDirectory:onClient: did not correctly handle filenames that contained characters outside of the extended ASCII range. The method returned instances of utf8, which had to be decoded before further use.

Bug 47314 - Debugger single-stepping could get stuck

Product: GemStone/S 64 Bit

Versions: 3.4.1, 3.4, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8, 3.2.7, earlier versions

Fixed In: 3.4.2, 3.3.7

When single stepping through code in a debugger, it was possible for stepping to get stuck and be unable to advance.

For example, in GBS, select the following and GS-debug-it:

```
| dict | dict := KeyValueDictionary new. dict
```

```
    at: #notPresent
    ifAbsent: [^self].
^75
```

Step through the method using the "step" button; when it gets to the non-local return of self, it will not advance.

Bug 47295 - GciFetchSize ambiguous return value for objects of size 0

Product: GemStone/S 64 Bit

Versions: 3.6.3, 3.6.2, 3.6.1, 3.6, 3.5.7, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.2, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.x, 2.x, earlier versions

The GCI function GciFetchSize() returns 0 both to indicate an error, and as a valid return when querying for an object of size 0.

To avoid perturbing existing GCI applications, this method will not be updated to provide different return values for these two cases.

Workaround:

Use GciErr() to determine if it is an error.

Bug 47291 - GEM_PGSVR_UPDATE_CACHE_ON_READ can cause "Repository root page is corrupted..." errors for remote logins

Product: GemStone/S 64 Bit

Versions: 3.4, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.2, 3.3.1, older versions

Fixed In: 3.4.1, 3.3.7

When the remote gem configuration parameter GEM_PGSVR_UPDATE_CACHE_ON_READ is set to true, there is a risk that logins from remote gems may encounter the error 'Repository root page is corrupted or contains a disk media defect'.

This does not indicate an actual repository problem; there is a code path in the page servers, that does not correctly handle the case when the requested root page is no longer in the Stone's cache.

Workaround:

Changing the GEM_PGSVR_UPDATE_CACHE_ON_READ to FALSE avoids the problem entirely.

Decreasing STN_CHECKPOINT_INTERVAL, to checkpoints are more frequent, will reduce the chances of a problem. The Stone writes the root pages into the cache on checkpoints so more frequent checkpoints increases the probability that the requested root page will still be in the cache.

Bug 47275 - GEM_PGSVR_USE_SSL is unreliable and should not be used

Product: GemStone/S 64 Bit

Versions: 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.2, 3.3.1, 3.3

Fixed In: 3.4.1

There are code paths that are broken when the configuration parameter GEM_PGSVR_USE_SSL is set to TRUE. There are thread safety issues in the socket code, making the multi-threaded pgsvr end of the ssl connection unreliable.

Workaround:

Do not set GEM_PGSVR_USE_SSL to true.

Bug 47244 - Login attempts when logins suspended generates clutter in topaz/gem and page manager logs

Product: GemStone/S 64 Bit

Versions: 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 2.4.8, 2.4.7, earlier versions

Platform All Platforms

Fixed In: 3.5

Impact: Informational

When attempting to login while logins are suspended, the client will actually attempt to login 20 times under the covers (once per second) before giving up. Each of these hidden login attempts will generate messages to both the topaz/gem log and the page manager log.

The topaz/gem log will see entries like:

```
topaz> login
[Info]: LNK client/gem GCI levels = 34001/34001
--- 10/23/2017 15:41:59.575 PDT Logging out login failed due to error 4053(logins suspended), retrying for 20 seconds
[Info]: LNK client/gem GCI levels = 34001/34001
--- 10/23/2017 15:42:00.577 PDT Logging out
[Info]: LNK client/gem GCI levels = 34001/34001

(repeat last 2 lines 18 times)

-----------------------------------------------------
GemStone: Error Fatal
Login failed: logins are currently disabled., Error Category: 231169 [GemStone] Number: 4053 Arg Count: 1 Context : 20 exception : 20 Arg 1: 20

Login failed due to errors. topaz>
```

While the page manager log will contain entries like:

```
  Warning: PageSetLostOt didn't find session 6

(repeated 20 times)
```

Bug 47234 - gslist crashes if GS64 2.x Stone or NetLDI is running

Product: GemStone/S 64 Bit

Versions: 3.4

Fixed In: 3.4.1

If a Stone or NetLDI from older, 2.x version of GemStone/S is running on a host machine, the v3.4 gslist gets a segmentation fault.

Bug 47197 - Mac High Sierra issues related to apfs file system

Product: GemStone/S 64 Bit

Versions: 3.4, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.2, 3.3.1, 3.3, 3.2.17, 3.2.16, earlier versions

Platform Mac only

Fixed In: 3.4.1, 3.3.7

Impact: Critical

The Mac High Sierra version, installed on SSD drives, may automatically convert the drive to their new APFS file system. GemStone did not recognize this file system and was unable to start.

Workaround:

Set
STN_ALLOW_NFS_EXTENTS=TRUE; in the stone config file.

Bug 47187 - listReferences could find references from a large NSC that did not directly contain the object

Product: GemStone/S 64 Bit

Versions: 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.2, 3.3.1, 3.3.x

Fixed In: 3.4, 3.3.7

If a large UnorderedCollection (NSC) contained an object that referenced a search object, but did not contain the search object directly, the results of a listReferences: or fastListReferences: could incorrectly include the NSC itself, in addition to the correct referencing object.

Bug 47133 - Suspended checkpoints will silently resume after an Epoch GC

Product: GemStone/S 64 Bit

Versions: 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.2, 3.3.1, 3.3, 3.2.17, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2

Platform All Platforms

Fixed In: 3.4, 3.3.7, 3.2.18

Impact: Critical

When checkpoints have been suspended using "System suspendCheckpointsForMinutes:", if an Epoch GC should run during this time window, the suspension is cancelled and checkpoints will resume, without any warning messages in the stone log. If a checkpoint should subsequently occur while extent copies are being taken, the resulting extent copies could be corrupted and therefore unusable.

Workaround:

Suspend Epoch GC using: "System disableEpochGc" before executing the "System suspendCheckpointsForMinutes:" command, and then resume them using "System enableEpochGc" after the suspension time window or when checkpoints are resumed using "System resumeCheckpoints"

Bug 47131 - Out of range #at: into very large ByteArray triggers corrupt object error

Product: GemStone/S 64 Bit

Versions: 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.2, 3.3.1, 3.3

Fixed In: 3.4, 3.3.7

With a large ByteArray (a ByteArray with a size > 16272), if an argument to at: or at:put: is greater than the size of the ByteArray, it triggers a corrupt object error rather than the expected OffsetError. A corrupt object error prevents commit.

Workaround:

The following filein patches the ByteArray at: method.

```
category:  'Patch bug 47131'
method: ByteArray
_at: anIndex

"Original ByteArray>>at: method -- demoted to provide
 index range check before calling the primitive

 Returns the value of an indexed variable in the receiver.
 The argument anIndex must not be larger than the size of the
 receiver, and must not be less than 1.

 Generates an error if anIndex is not a SmallInteger or is out of
 bounds, or if the receiver is not indexable."



(anIndex _isInteger)
  ifTrue: [^ self _errorIndexOutOfRange: anIndex]
  ifFalse: [^ self _errorNonIntegerIndex: anIndex].
self _primitiveFailed: #at: args: { anIndex } .
self _uncontinuableError
%

category:  'Patch bug 47131'
method: ByteArray
at: anIndex

"Replacement ByteArray>>at: method -- original moved to ByteArray>>_at: "

(anIndex > self size) ifTrue: [ ^ self _errorIndexOutOfRange: anIndex ].
^ self _at: anIndex
%
```

Bug 47116 - Symbols with trailing NUL bytes in Unicode Comparison Mode

Product: GemStone/S 64 Bit

Versions: 3.6.3, 3.6.2, 3.6.1, 3.6, 3.5.7, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2

Symbols with trailing NULs (Characters with codePoint 0), are not handled correctly in equality comparisons in a repository that is in Unicode Comparison Mode.

Two symbol that have exactly the same characters, but in which one has one or more trailing NULs, will return false for =, >, and <. However, the return from >= and <= will be true.

Workaround:

The private method _unicodeEqual:, added in v3.4, avoids the inconsistent equality comparison.

Bug 47107 - Risk of SEGV when accessing hidden classes, such as done by CanonicalObjectManager

Product: GemStone/S 64 Bit

Versions: 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, earlier versions

Fixed In: 3.4

Sending a message to the results of the private primitive method Object >> _primitiveAt: has a risk of SEGV, when used with with instances of internal, hidden classes LargeObjectNode oe NscNode that compose a large byte-based object. The risk further depends on bit patterns of the object.

Instance of internal, hidden classes are normally not accessible from the Smalltalk image, but may be returned by maintenance methods, such as the contents of system sets, find references, System >> _writtenObjects, etc.

The Object Canonicalization Framework performs some low-level accesses, and is exposed to this bug.

Workaround:

Code that sends _primitiveAt: is expected to check the size using _primitiveSize, and not invoke _primitiveAt: for indexes outside that range. Overriding the _primitiveSize for the NscNode and LargeObjectNode internal classes avoids problems when using the results of _primitiveSize to iterate using _primitiveAt:, such as in CanonicalObjectManager.

This code must be executed as SystemUser.

```
method: @211969
_primitiveSize
  ^ 0
%
method: @209409
_primitiveSize
  ^ 0
%
commit
```

Bug 47054 - Read-only configuration files cause unclear failure and inconsistent online addExtent behavior

Product: GemStone/S 64 Bit

Versions: 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, earlier versions

Fixed In: 3.4

If the stone's executable and system configuration files are read-only, the stone will not start, but does not provide a clear error message.

While having a stone's system configuration file writable and the executable file read-only allow stone to start, adding an extent will not write the updated DBF_EXTENT_NAMES to the configuration file, creating problems on restart.

Bug 47021 - GsExternalSession>>lastResult may be incorrect with multiple sessions

Product: GemStone/S 64 Bit

Versions: 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2

Fixed In: 3.4, 3.3.7

GsExternalSession >> lastResult is used to fetch the results of execution.

This method fetched the result from the session that had the most recent previous access, normally a method such as isResultAvailable. However, in an environment with multiple instance of GsExternalSession performing work, if the most recent previous session access was to a different session, GsExternalSession >> lastResult returned the results for that session, not for the receiver.

Workaround:

Sending self _setSessionId ensures that the receiver's session is the current session in the GCI.

Bug 47003 - Floating point printing may include unnecessary trailing digits

Product: GemStone/S 64 Bit

Versions: 3.6.3, 3.6.2, 3.6.1, 3.6, 3.5.7, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.2, 3.3.1, 3.3

In v3.3, the default printing of floating points was changed from using the format string '%.16E' to using the format string '%.16g', to avoid the string including unnecessary trailing digits (that is, to print optimally). This change unfortunately introduced a bug in missing precision (bug 46855).

It also did not always print optimally. For example, in versions prior to 3.3,
9.44 asString -> 9.4399999999999995E00
9.45 asString -> 9.4499999999999993E00

In 3.3 and later:
9.44 asString -> 9.44
9.45 asString -> 9.449999999999999

While both these printing formats for 9.45 are entirely correct, they are not optimal, since 9.45 has sufficient precision. Each of the following returns the same object, the SmallDouble with OOP 9380547663850006118.
'9.45' asNumber
'9.4499999999999993E00' asNumber
'9.449999999999999' asNumber

Workaround:

As in versions prior to 3.3, printing using asStringUsingFormat: allows you to specify exactly how many decimal places to display, performing decimal rather than floating point rounding.

Bug 46990 - Random new performance

Product: GemStone/S 64 Bit

Versions: 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2

Fixed In: 3.4

Impact: Informational

Random new defaults to creating an instance of Lag25000CmwcRandom. This class is designed to be truly random for long sequences, and the process of creating the initial instance and seed is slow, on the order of 30 ms.

If you require a Random for single use, it it more efficient to use HostRandom or Lag1MwcRandom.

Starting with v3.4, Random new answers the HostRandom singleton, and Random seed: answers a new Lag1MwcRandom.

Bug 46989 - TimeZone cache corrupted during upgrade to 3.2.0 or later

Product: GemStone/S 64 Bit

Versions: 3.5.7, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.2, 3.3.1, 3.3, 3.2.17, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2

Platform All Platforms

Impact: Informational

When upgrading a repository from prior to 3.2.0 to later than 3.2.0, the TimeZone cache can be corrupted, causing failures on lookups using TimeZone>>for:.

Workaround:

To fix:

1. Login as SystemUser
2. Execute "TImeZone cache rehash"
3. Commit

Bug 46975 - findPatternNoCase: fails with Unicode16 argument

Product: GemStone/S 64 Bit

Versions: 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.2, 3.3.1, 3.3, 3.2.17, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.x, 3.2

Fixed In: 3.4

Calling findPatternNoCase: with a String receiver and an element in the argument Array that is a Unicode16 will fail with an illegal pattern error.

This occurs when the argument is an instance of String (which only includes Characters with codePoints under 256), while the Unicode string argument has Characters with codePoints greater than 127 (i.e., Unicode16).

For example:

run
'après ski' asString findPatternNoCase: (Array with: ('après' asUnicodeString)) startingAt: 1
%
a ArgumentError occurred (error 2017), anArray('apr..s') is an illegal pattern for string comparisons.

Workaround:

login as SystemUser and execute:

```
run
String removeSelector: #findPatternNoCase:startingAt:
%
commit
```

Bug 46915 - Files created using GsFile using String containing non-ASCII characters not accessible

Product: GemStone/S 64 Bit

Versions: 3.5.7, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.2, 3.3.1, 3.3

Platform All Platforms

Impact: Informational

Prior to GS/64 3.3, there was a bug in GsFile (44786) where on a unix environment using a UTF-8 locale, if a file was created using a filename that was a String containing non-ASCII characters, the resulting file would have a corrupted filename that was not a valid UTF-8 string. Since 3.3, these files are now not directly accessible from GsFile. Attempts to read/append them using the same String will fail with a file not found error.

Workaround:

Rather than using:

GsFile open: <stringContainingNonAsciiChars>

use:

GsFile open: (Utf8 withBytes: <stringContainingNonAsciiChars> asByteArray)

Bug 46900 - Persistent GsFile instances can lose transient state within same session

Product: GemStone/S 64 Bit

Versions: 3.6.3, 3.6.2, 3.6.1, 3.6, 3.5.7, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.2, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.x, 3.1.0.6, 3.1.0.5, 3.1.x, 3.0.x

Platform All Platforms

Fixed In: N/A

Impact: Informational

GsFile instances have always been transient objects that can only be used within a single session. If a GsFile was made persistent, attempts to reuse it in another session would trigger an error due to the loss of the transient state information about the file in use.

But due to a design change in GS/64 3.0, the behavior has changed to where in the *same* session, if a persistent GsFile is flushed from local memory due to GC activity and later faulted back in, it will also lose its transient state information and trigger the error:

```
  --------------------------
  a ImproperOperation occurred (error 2364), The object <aGsFile>
  that has some associated session state has lost that transient state.
  Examples of classes that use session state are GsSocket and GsFile.
  --------------------------
```

This behavior is also mentioned in the class comment for GsFile:

```
  --------------------------
  Beginning with Gs64 v3.0, instances of GsSocket automatically have
  their C state closed when the instance is garbage collected or
  when a persistent instance drops out of memory.
  --------------------------
```

Workaround:

Make sure that GsFile instances do not become persistent. This typically happens by having them referenced by other committed objects. Use of System>>sessionStateAt:[put:] is one way to access GsFile instances in a safe manner that will avoid having them become inadvertently committed/persistent.

Bug 46879 - withAll: may truncate character codePoints of a DoubleByteString

Product: GemStone/S 64 Bit

Versions: 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.17, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, older versions

Fixed In: 3.4

When sending String >> withAll: with an argument of some particular specific structures of DoubleByteString argument, codePoints in the result may be truncated to less than 256, and the result was an instance of String.

Bug 46855 - Floating point printing may miss significant digits

Product: GemStone/S 64 Bit

Versions: 3.6.3, 3.6.2, 3.6.1, 3.6, 3.5.7, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.2, 3.3.1, 3.3

In v3.3, the default printing of floating points was changed from using the format string '%.16E' to using the format string '%.16g'. This allows printing of many floats to be limited to the digits that were of interest, However, this could result in missing precision.

Doubles can have up to 17 significant decimal digits. The E format prints one digit before the decimal point and "precision" digits after, while the g format, however, only prints "precision" digits total, so only 16.

To illustrate, in versions prior to 3.3,
7.6000000000000001 asString -> 7.6000000000000005E00

and in v3.3 and later:
7.6000000000000001 asString -> 7.600000000000001

This change to 16 digits results in printing with insufficient digits to resolve the correct float.

```
'7.6000000000000001' asNumber
    returns OOP 9360281465526838902
'7.6000000000000005E00' asNumber
    returns OOP 9360281465526838902
'7.600000000000001' asNumber
    returns OOP 9360281465526838918
```

Workaround:

using #asStringLegacy provides the printing as in earlier versions.

Bug 46817 - Premature gem exit during login can reduce max session count

Product: GemStone/S 64 Bit

Versions: 3.3.3, 3.3.2, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.3, 3.1.0.2, 3.1.0.1, 3.1

Platform All Platforms

Fixed In: 3.3.4

If a gem exits prematurely during login before all the handshaking between the gem and stone is completed, the session counter on the stone is not properly decremented. If enough gems exit like this, then it is possible for later logins to fail with "the maximum number of users is already logged in" even though the real max session count has not been reached.

Workaround:

Restart the stone.

To avoid the problem, analyze any client login failures to determine the cause and resolve the underlying issue. These are likely to be network issues between the client, gem, and/or stone. Note that this problem occurs *before* the gem log has been established, so will not be available for diagnosis.

Bug 46750 - GsFile>>position: failures for arguments over 2 billion

Product: GemStone/S 64 Bit

Versions: 3.3.3, 3.3.2, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8, 3.2.x, 3.2, 3.1.0.6, 3.1.0.x, 3.1, 3.0.1, 3.0

Platform All Platforms

Fixed In: 3.3.4

When using GsFile>>position:, if a position argument equal to or greater than 2^31 (2147483648) is specified, the method will return nil (normally used to indicate an error) and subsquent calls to GsFile>>position will return incorrect values. Reads and/or writes will still be performed correctly.

If at some point GsFile>>position: is called with a value less than 2^31 (2147483648), GsFile>>position will again return correct values until the position exceeds 2^31.

Bug 46727 - Stone may hang on out of disk in tranlog directories

Product: GemStone/S 64 Bit

Versions: 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.x, earlier versions

Fixed In: 3.4

When the system runs out of space in the tranlog directory, and the timing is such that a checkpoint in a newly created tranlog cannot complete, the stone may hang. This final partially written tranlog is not usable for restart, and must be deleted.

Bug 46723 - Idle Gems not terminated by STN_GEM_TIMEOUT

Product: GemStone/S 64 Bit

Versions: 3.3.3, 3.3.2, 3.3.1, 3.3, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.x, 3.1.0.6, 3.1.0.5, 3.1.x, 3.0.x

Fixed In: 3.3.4, 3.2.16

The configuration parameter STN_GEM_TIMEOUT is designed to terminate gems that have had no interaction with the stone for some period of time. There was a logic error that prevented the check from running, with the result that idle gems were not terminated

Bug 46694 - Corrupt object / relops: encrypt (no Unicode collator) if fail to rebuild indexes on change to UnicodeComparisionMode

Product: GemStone/S 64 Bit

Versions: 3.6.3, 3.6.2, 3.6.1, 3.6, 3.5.7, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2

Impact: Informational

In Unicode Comparison Mode, introduced in v3.2, all string comparison operations (both traditional string and symbol and unicode string) use an ICU library collator, unlike in standard mode in which native GemStone code is used to perform comparisons.

When changing an existing application from the standard mode to Unicode Comparison Mode, you must take steps to ensure any structures relying on comparisons are rebuilt; indexes on strings must be rebuilt, sorted collections on strings may need to be re-sorted, and dictionaries with string keys may need to be rebuilt.

While the risk is primarily that lookups may fail to find objects, if you do not rebuilt indexes it reports a corrupt object error as below. This error is in the underlying system comparison and is not detected by auditIndexes.

ERROR 2261 , a InternalError occurred (error 2261), The object with object ID 20 is corrupt. Reason: 'relops: encrypt (no Unicode collator) -- Unicode strings encountered' (InternalError)

Workaround:

As part of changing to Unicode Comparison Mode, carefully ensure your application has had all string-sort based structures rebuilt.

Bug 46655 - Cannot change objectSecurityPolicy of dbTransient objects

Product: GemStone/S 64 Bit

Versions: 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2

Fixed In: 3.4, 3.3.7

If you have a previously committed dbTransient object (connected to a persistent object and so visible to other sessions), committing changes to the dbTransient object's security policy does not result in this change being visible to other sessions, or for new sessions logging in.

Bug 46646 - Sign of -0.0

Product: GemStone/S 64 Bit

Versions: 3.6.3, 3.6.2, 3.6.1, 3.6, 3.5.7, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.x, 3.0.x, 2.x, earlier versions

Impact: Informational

The expression

```
-0.0 sign
```

returns 0. This is according to the ANSI Smalltalk standard, 5.6.2.33, "Answer 1 if the receiver is positive, 0 if the receiver equals 0, and -1 if it is negative".   Pharo, and some interpretations of IEEE 754, disagree and expect this expression to return -1.

Bug 46645 - listReferences: fails to find object in large IdentityBags and IdentitySets

Product: GemStone/S 64 Bit

Versions: 3.3.3, 3.3.1, 3.3, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.x, earlier versions

Fixed In: 3.3.4, 3.2.16

If an object is in an IdentityBag or IdentitySet with more than about 1015 or 2030 elements, respectively, a listReferences: or fastListReferences: operation will not detect the reference.

This is related to the internal large object node structure that implements NSCs.

Bug 46552 - Symbol >> keywords added semicolons to unary tokens

Product: GemStone/S 64 Bit

Versions: 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.x, earlier versions

Fixed In: 3.4

Symbol >> keywords parses a given symbol intended to represent a keyword message selector, and returns an Array of keywords. This method did not operate correctly when the argument was not a valid keyword selector; if the argument was a unary selector or the receiver included a space, a colon was added to the preceding token.

In addition, it returned an empty array when any tokens in the receiver were not legal selectors, e.g. included illegal characters or started with digits. While this is the behavior per strict reading of the message comment, it did not conform to the behavior of other Smalltalk dialects.

Bug 46539 - GemStone's pstack cannot be used with nonzero kernel.yama.ptrace_scope

Product: GemStone/S 64 Bit

Versions: 3.6.3, 3.6.2, 3.6.1, 3.6, 3.5.7, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.5, 3.3.4, 3.3.3, 3.3.2, 3.3.1, 3.3, earlier versions

Platform Linux

Impact: Informational

GemStone distributes a pstack utility, distinct from the more limited Linux pstack command. GemStone's pstack is a wrapper for the gdb debugger.

gdb is disallowed from attaching to a process when kernel.yama.ptrace_scope=1,and as kernel.yama.ptrace_scope=0 is a significant security hole, by default this is set to 1 in recent Linux versions. pstack therefore often cannot be used.

In versions of GemStone before 3.4, stack traces using kill -USR1 and on a fatal error were also disabled. In 3.4 and later, you can get a stack traces using kill -USR1 or on hostCallDebugger, when the NetLDI is in guest mode so the real and effective users of the process are the same.

In versions of GemStone up to and including 3.5.6 and 3.6.1, GemStone could not get stacks for processes started when the NetlDI is in authentication mode, so the real and effective userIds are not the same.

Workaround:

sending
   kill -USR1 <pid> prints the desired stacks to the process log, in v3.4 and later in guest mode.

sending (as root user)
  sysctl kernel.yama.ptrace_scope=0 will allow stacks, with the introduction of a general security risk.

Bug 46518 - waitstone may return error while stone in startup

Product: GemStone/S 64 Bit

Versions: 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1., 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.x, 3.0.x, 2.4.8, 2.4.7, 2.4.6, earlier versions

Fixed In: 3.4

It is possible for the waitstone to exit with an error, rather than waiting, if the stone is in startup.

Bug 46506 - GciRtlLoad with null path fails on Windows and for 32-bit GCI applications

Product: GemStone/S 64 Bit

Versions: 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, earlier versions

Fixed In: 3.4

When the path argument to GciRtlLoad is NULL, it uses a default path, which is the lib directory under GEMSTONE.

On Windows, the shared libraries are distributed in the bin directory, and GciRtlLoad does not find them, and fails.

With 32-bit applications, the library path is lib32, which likewise results in load failure.

Workaround:

Copy the shared library .dlls to the %GEMSTONE%\lib directory on Windows or the $GEMSTONE/lib directory for 32-bit UNIX. The .so filenames include the bits, so will not overwrite the 64-bit libraries.

Bug 46494 - Upgrade issues; cannot flexibly manage ICU library version on AIX

Product: GemStone/S 64 Bit

Versions: 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2

Platform AIX only

Fixed In: 3.4

GemStone uses the open-source ICU libraries to support sophisticated handling of sorting and collation of strings containing Characters outside of the ASCII range. As the Unicode standard is updated and versioned, the ICU open source libraries are also versioned. This can impact GemStone structures that rely on fixed collation order.

Data structures such as SortedCollections that contain Unicode strings, dictionaries in which the keys are Unicode strings, or has indexes on Unicode strings, may be affected. In Unicode Comparison Mode, all strings use ICU comparisons and structures including traditional Strings are also affected. Applications that do not use ICU-based comparisons in this way are not impacted, and can use the latest ICU library versions without risk.

Rather than rebuilding affected data structures, upgraded GemStone applications can continue to use the earlier version of the ICU libraries. The distribution for GemStone v3.3.1 and later includes both ICU v51.2 and 54.1, and additional checking is done to ensure that the version of the ICU shared libraries that are loaded into the Gem matches the version of the ICU that was used for collating in the repository.

While on other platforms, GemStone can use repository settings or environment variables to control the specific version of the ICU library that is loaded into the Gem, AIX does not support deferred library loading. This means additional manual work is required on AIX.

Workaround:

On AIX, GemStone will load the latest library version; for 3.3.x this is v54.1. If you do not have any ICU comparisons, before upgrading you can configure your application to use this ICU version by executing as SystemUser:

```
 Globals at: #IcuLibraryVersion put: '54.1'
```

ICU library loading is handled automatically on AIX in v3.4 and above. For applications that should continue to use the ICU version 51.2, which was current for GemStone v3.2.x, do the following:

```
cd $GEMSTONE/lib ; chmod +w .
mv libicudata.54.1.so libicudata.54.1.so.save
ln -s libicudata.51.2.so libicudata.54.1.so
mv libicui18n.54.1.so libicui18n.54.1.so.save
ln -s libicui18n.51.2.so libicui18n.54.1.so
mv libicuuc.54.1.so libicuuc.54.1.so.save
ln -s libicuuc.51.2.so libicuuc.54.1.so
```

Note that applications that contain collated Unicode strings from version 3.1.x must rebuild these structures.

Bug 46478 - Removing UserGlobals symbol dictionary can cause errors

Product: GemStone/S 64 Bit

Versions: 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.2, 3.3.1, 3.3

Fixed In: 3.4

Removing UserGlobals can cause problems with some methods, such as Behavior >> categoryNames. This can result in problems with tools such as GBS.

Bug 46475 - changeClassTo: with fewer instance variables may have problems

Product: GemStone/S 64 Bit

Versions: 3.3.1, 3.3, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.x, 3.2, 3.1.15, 3.1.0.6, 3.1.0.5, 3.1.x, 3.0.x, 2.4.8, 2.4.7, 2.4.6, 2.x

Fixed In: 3.3.3, 3.2.16

When changeClassTo: is used to change an instance to a compatible type of class that has fewer instance variables than its current class, the handling of the surplus instance variables is not always correct.

Instances of IdentityBag become corrupt.

Instances of some kinds of collection shift the contents that were in named instance variables into the unnamed instance variables. For IdentitySet, the collection size remains the same, so the contents effectively gain values that were in the named instance variables and lose some of the previous contents.

Other collections, such as Set, are unaffected.

Non-collection classes also move the contents of the named instance variables into unnamed slots, however, since these classes are not indexable, these slots are inaccessible (except using _primitiveAt:).

See also bug 45956 on changeClassTo: behavior with respect to the handling of named and indexed instance variables.

Bug 46419 - Client exit during login results in clean exit/log deletion

Product: GemStone/S 64 Bit

Versions: 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.x, 3.1.0.6, 3.1.x, earlier versions

Fixed In: 3.3.3

During the login process, after the gem log is created but before the login completes, if the client exits then the Gem sees this as an error. However, since the client may have elected to exit, this may not actually be an error; so this category of error is treated as a clean exit, to avoid false positives.

This means that if there is an actual fatal error in the client during the login, this is still considered a clean Gem exit, not a Gem error. To make diagnosis easier, as of v3.3.3 the Gem log, which would normally be deleted on a clean exit, is kept for the case where the client process cannot be contacted.

Bug 46404 - Page reclamation during tranlog restore can hit Page Cache Fault

Product: GemStone/S 64 Bit

Versions: 2.3.1.6, 2.3.x, 2.2.5.4, 2.2.x

Platform All Platforms

Fixed In: 2.4

There is a small risk that during tranlog restore the stone can fail with a Page Cache Fault, with the special message:

non-data page in StnGarReclaim

Workaround:

Upgrade to version 2.4.0 or later.

Bug 46398 - Divide by float zero and other operations using floats do not signal exceptions

Product: GemStone/S 64 Bit

Versions: 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.x, 3.0.x

Fixed In: 3.4

Impact: Critical

Operations that would return a Float that cannot produce reasonable results return an exceptional float, such as PlusInfinity or PlusQuietNaN, rather than signaling an exception.

For example,
   100 / 0.0 results in PlusInfinity
    -100.0 sqrt results in MinusQuietNaN

In 2.x and earlier, division by floating point zero did raise an exception, while other operations returned exceptional floats.

Workaround:

When performing operations involving floats, explicitly check for exceptional float return values (see Number >> kind).

In version 3.4 and later, various kinds of FloatingPoint exceptions, including divide by zero, can be enabled. See v3.4 Release Notes/FloatingPoint signalling Exceptions.

Bug 46384 - System _tempObjSpaceMax can return negative values on larger GEM_TEMPOBJ_CACHE_SIZE

Product: GemStone/S 64 Bit

Versions: 3.3.1, 3.3, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.x, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.4.8, 2.4.4.7, earlier versions

Fixed In: 3.3.3, 3.2.16

When GEM_TEMPOBJ_CACHE_SIZE is set to a large value (over about 2800000), then System _tempObjSpaceMax returns an incorrect negative value.

Bug 46381 - Object >> storeOn: fails for large strings

Product: GemStone/S 64 Bit

Versions: 3.3.1, 3.3, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.x, 3.0.x, 2.4.8, 2.4.7, 2.4.6, 2.4.x, older versions

Fixed In: 3.3.3, 3.2.16

When the method Object >> storeOn: is executed for a string of more than 9999 bytes, it encounters a typo in the Smalltalk code and errors.

Bug 46358 - AES encryption methods, invalid salt size may corrupt memory

Product: GemStone/S 64 Bit

Versions: 3.3.2, 3.3.1, 3.3, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2

Fixed In: 3.3.3, 3.2.16

Encryption methods such as aesEncryptWith256BitKey: require specific sizes for the key and salt. If an incorrect salt size was provided, it did not raise an exception as it should, and this condition could corrupt memory, causing topaz to crash.

Bug 46315 - DateTime asStringISO8601 reported time/timezone incorrectly

Product: GemStone/S 64 Bit

Versions: 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, older versions

Fixed In: 3.3.3

The method DateTime >> asStringISO8601 should report the time as adjusted for the DateTime's timezone, followed by the timezone offset.

Instead, it was reporting the time adjusted for the repository's timezone, but including the timezone offset for the DateTime's timezone. This produced incorrect results when executed for DateTimes that were not in the current timezone.

Bug 46290 - Idle gems in transactionless mode could be terminated by sigAbort

Product: GemStone/S 64 Bit

Versions: 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.2, 3.3.1, 3.3, 3.2.17, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.x, 3.2

Fixed In: 3.4

If a transactionless session is executing bytecodes, it will respond to sigAbort by silently aborting and continue execution.

However, if the session is entirely idle, the session will not notice the sigAbort or sigLostOT, from the Stone, and may be killed by the Stone as part of lostOT handling

Bug 46265 - Native code may be disabled on Solaris/SPARC

Product: GemStone/S 64 Bit

Versions: 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1

Platform Solaris/SPARC

Fixed In: n/a

Impact: Informational

Under some conditions, a requested memory address may be out of range on login. This results in native code being disabled, with a warning message such as:

native code error, format1call dispTooBig , methodId:20, ipOffset 0, native code disabled for remainder of session

Solaris/SPARC is being deprecated for production use.

Workaround:

This error can be ignored.

Bug 46262 - gslist -m does not include Stones or Caches

Product: GemStone/S 64 Bit

Versions: 3.3.1, 3.3, 3.2.15

Fixed In: 3.3.3, 3.2.16

gslist -m reports information for server processes running on remote nodes. Due to this bug, only netldi process information is reported; Stones, caches, and logsender/receiver processes are not included.

Bug 46252 - Operations may hang when oopHighWaterMark over 4 billion

Product: GemStone/S 64 Bit

Versions: 3.3.3, 3.3.2, 3.3.1, 3.3, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.3, 3.1.0.2, 3.1.0.1, 3.1, 3.0.1, 3.0

Platform All Platforms

Fixed In: 3.3.4, 3.2.16

Impact: Critical

In very large repositories with a oop high water mark greater than 4 billion, operations that scan oops over the entire repository may hang and run hot.

You can determine your oop high water mark by executing

```
  System _oopNumber oopNumberHighWaterMark.
```

or by checking the stone startup information in the stone log:

```
  Repository startup statistics:
    Pages Need Reclaiming =XXXXX
    Free Oops=XXXXX
    Oop Number High Water Mark=XXXXXXXX
```

The affected operations includes:
  MFC and subsequent processing such as write set union sweep
  restore from backup
  object audit
  list instances
  count instances
  find object

To diagnose this problem, take a stack trace (using either gdb/dbx or "kill -USR1") of the hanging gem. This will include, in the C stack:

```
   UtlFindFirstBitSet
```

Most cases of this bug are fixed in 3.3.1 and later. However, restore from backup is still exposed.

Workaround:

In some cases, you can kill the session and try the operation again.

Bug 46173 - GsFile operations can hang on seteuid / __nptl_setxid GLIBC bug

Product: GemStone/S 64 Bit

Versions: 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.3, 3.1.0.2, 3.1.0.1, 3.1, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.4, 2.4.x

Platform Linux only

Impact: Informational

Customers running Linux GLIBC versions prior to 2.12 may be subject to a GLIBC bug (bug 3270) which can cause GsFile operations calling the OS seteuid( ) operation to deadlock. Using pstack to view the stack of the hanging gem, if you see these entries then you may have hit this bug:

```
    __nptl_setxid () from /lib64/libpthread.so.0
    seteuid () from /lib64/libc.so.6
```

The following web pages have further discussion on this issue:

https://sourceware.org/bugzilla/show_bug.cgi?id=3270

https://sourceforge.net/p/gfarm/tickets/84/

https://github.com/xrootd/xrootd/issues/60

Workaround:

Upgrade your Linux machine to use a newer version of GLIBC. You should upgrade to version 2.12 or later.

The following C code includes a test that can be run to conclusively determine if your machine is subject to this bug.

Bug 46166 - Invalid class instance variable specification does not prevent class creation

Product: GemStone/S 64 Bit

Versions: 3.3, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.3, 3.1.0.2, 3.1.0.1, 3.1, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.4, 2.4.x, 2.3.1.6, 2.3.x, 2.2.5.4, 2.2.x, 2.x

Fixed In: 3.3.1, 3.2.14

If a subclass creation expression includes invalid class instance variables, then it returns an error; but the class was created and left in a modifiable state.

Subclass creation creates the class instance variables in a secondary step, after the class has been created; and did not clean up if class instance variable addition was not successful.

Bug 46148 - Repeated replay of transaction log can cause corruption

Product: GemStone/S 64 Bit

Versions: 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.x, 3.0.x, 2.4.8, 2.4.7, 2.4.6, 2.4.x, 2.x, 1.x

Fixed In: 3.2.15

Impact: Critical

If a partial transaction log is restored into a repository, and then the completed transaction log is restored again into this same repository, some records in the later part of the tranlog may not be replayed, which results in corruption.

The scenario in which this may occur is in a warm backup system, where the current transaction log of the production system is inadvertently replayed into the standby while the production system is still writing records to it. After the production system completes writing to that transaction log, if the partially-replayed log is re-restored, this bug is exposed.

Workaround:

Ensure that your system does not replay partial logs.

Bug 46140 - Inconsisent return type for remove* and add* methods

Product: GemStone/S 64 Bit

Versions: 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.x, 3.2, 3.1.15, 3.1.0.6, 3.1.0.5, 3.1.x, 3.0.x, 2.4.8, 2.4.7, 2.4.6, 2.x

Fixed In: 3.4

RcIdentityBag remove: returned the receiver, but per ANSI should return the argument.

Set and Bag removeAll: return the receiver; while ANSI does not specify, for consistency with other GemStone collection removeAll: handling should return the argument.

Bug 46133 - "Senders of" does not find selectors on which a breakpoint is set

Product: GemStone/S 64 Bit

Versions: 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.2, 3.1.0.1, 3.1, 3.x

Fixed In: 3.3

Finding all senders of a particular selector will not find selectors if the method that includes the sender has a breakpoint on that step point.

Bug 46116 - Unreferenced Class may not get GCed if class variable holds reference to instance

Product: GemStone/S 64 Bit

Versions: 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.3, 3.1.0.2, 3.1.0.1, 3.1, 3.0.1, 3.0

Fixed In: 3.3.1

If a application class has a class variable, and that variable references an instance of the class, and the size of the instance is 0, then the internal handling of the class during voting does not perform reference checks, it is simply voted to not garbage collect. Under these cirumstances, the class will not get garbage collected even though there are no other references to the class or any instance of the class.

Because of limitations in findReferences when the target is a class, the problem is not easily detected.

Workaround:

Clear out references from the class variable.

Bug 46066 - gslist does not clear locks if another process reused PID

Product: GemStone/S 64 Bit

Versions: 3.6.3, 3.6.2, 3.6.1, 3.6, 3.5.7, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.x, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.4, 2.4.x, 2.3.1.6, 2.3.x, 2.2.5.4, 2.2.x, earlier versions

Fixed In: 3.2.13 (linux only)

Normally, when server processes die but leave their lock files behind, gslist -v reports these as "killed", and gslist -c will clear the old lock files.

However, if another process starts up and uses the PID that the server process was using, then gslist mistakenly believes that it's the same process. gslist -v reports it as "frozen", and gslist -c will not clear the lock file since the process is running.

Workaround:

Manually delete the lock file. These are normally in /opt/gemstone/locks/<processname> ..LCK, but the location may be /usr/gemstone/locks/ on older systems, or may be configured using GEMSTONE_GLOBAL_DIR.

Bug 46047 - AdminGem shutdown prevents operations that need GC lock from running

Product: GemStone/S 64 Bit

Versions: 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.x

Fixed In: 3.3.1

Impact: Informational

If the AdminGem is not running, then garbage collection steps from previous operations cannot complete. The error message returned from the markForCollection operation does not indicate the nature of the problem, it reports the lock was denied with the reason "vote state is doneVoting".

Workaround:

You should normally leave the AdminGem running. Start the AdminGem using

```
System startAdminGem
```

This can be done as part of the MFC sequence; executing it when the AdminGem is already running does not cause problems.

Bug 46035 - Repository>>timeToRestoreTo: replaced by restoreToPointInTime:

Product: GemStone/S 64 Bit

Versions: 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.x

Platform All Platforms

Impact: Informational

In GS/64 3.1, the method:

```
Repository>>timeToRestoreTo:
```

was changed to:

```
Repository>>restoreToPointInTime:
```

The name was changed to make this method consistent with the naming convention used by other restore methods.

Workaround:

Obsolete code using Repository>>timeToRestoreTo: should be changed to call Repository>>restoreToPointInTIme: instead.

Bug 45961 - System timeNs incorrect return of LargeInteger; value is smaller in 3.3 and later

Product: GemStone/S 64 Bit

Versions: 3.5.7, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.x, 3.1.x, 2.x, earlier versions

The method timeNs has been documented to return a SmallInteger, but actually has returned a LargeInteger.

As of version 3.3, the highest bits are discarded to ensure the result is a SmallInteger.

Given the SmallInteger range and nanosecond resolution, this creates a result range of over 36 years. However, values returned from earlier versions will be larger than values returned in v3.3 and later. For example, at a particular specific time, timeNs could return 1454362338115845960 in v3.2.12, and 301440833508998984 in v3.3.

Bug 45956 - changeClassTo: problematic behavior when target class has different size

Product: GemStone/S 64 Bit

Versions: 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.x, 3.0.x, 2.x, earlier versions

Fixed In: 3.4

changeClassTo: can be used to change the class of an object, between indexable and non-indexable classes (that is, between a class that allows or does not allow unnamed instance variables, to the converse), and to a class that has more or fewer named instance variables.

The behavior for on some of these changes was not correct with regards to overflow instance variable values, particularly between indexed and named instance variable slots.

When converting from an instance with indexable instance variables to a class that was not indexable, if there are more named instance variables in the target class, these slots were filled with nils, and the indexed contents were left in place. The resulting object was larger by that difference.

See also bug 46475 on changeClassTo: behavior, in which some kinds of collections may be corrupted/data lost.

Bug 45925 - STN_FREE_SPACE_THRESHOLD lower than GcUser #reclaimMinFreeSpaceMb can block reclaim

Product: GemStone/S 64 Bit

Versions: 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.x, 3.2, 2.4.8, 2.4.7, 2.4.6

Fixed In: 3.4

Reclaim activity is deferred when the free space on the system falls below the setting of the GcUser parameter #reclaimMinFreeSpaceMb. If this value is higher than the STN_FREE_SPACE_THRESHOLD and the amount of free space on the system falls between the two values, the system can become "stuck" where the free space is below the #reclaimMinFreeSpaceMb (so reclaim activity is deferred) but above the STN_FREE_SPACE_THRESHOLD (so the stone never adds additional space).

During normal operation on a quiet system, this can cause reclaim activity to be deferred indefinitely until other system activity finally causes the free space to drop below the STN_FREE_SPACE_THRESHOLD.

During tranlog replay, since there's no other system activity to reduce the free space, reclaim will be deferred forever. This will cause the tranlog replay to hang until the 30 minute timeout expires and shuts down the stone.

Workaround:

Keep the setting for reclaimMinFreeSpaceMb below the configuration setting for STN_FREE_SPACE_THRESHOLD. Don't use the default reclaimMinFreeSpaceMb setting of zero, which causes the system to calculate a default value that will often be higher than the STN_FREE_SPACE_THRESHOLD.

Bug 45892 - Syntax errors due to GemConvertArrayBuilder setting during upgrades / file-ins / compiles

Product: GemStone/S 64 Bit

Versions: 3.6.3, 3.6.2, 3.6.1, 3.6, 3.5.7, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.2, 3.1.0.1, 3.1, 3.0.1, 3.0

Platform All Platforms

Impact: Informational

In GS/64 3.0, the syntax for Array constructors was changed from using square brackets with commas (for example: #[a, b, c]) to using curly brackets with periods (for example: {a . b . c}). This was done to avoid conflicts with the use of square brackets for ByteArray constructors (for example: #[1 2 3]) for compatibility with other Smalltalks.

The System configuration parameter #GemConvertArrayBuilder was introduced to inform the compiler to treat #[...] as a pre-3.0 Array constructor and to compile it accordingly. But note that this will cause problems if processing post-3.0

code containg #[...] ByteArray constructors.

For example, if compiling code using the pre-3.0 Array constructor #[a , b, c], and #GemConvertArrayBuilder is incorrectly set to nil/false, a compiler error similar to the following will be generated:

```
 GemStone Smalltalk Compiler Errors:
   #[a, b, c]
 * ^1 ^2 *******

1: [1035] expected a right bracket (]) 2: [1034] unexpected token
```

Inversely, if compiling post-3.0 code containing the ByteArray constructor #[1 2 3] and #GemConvertArrayBuilder is incorrectly set to true, a compiler error similar to the following will be generated:

```
 GemStone Smalltalk Compiler Errors: #[1 2 3]
 * ^1
******* 1: [1034] unexpected token
```

When compiling code (for example, during upgrades or file-ins) be aware of which type of constructor is used in the source code and make sure that #GemConvertArrayBuilder is set accordingly. In general this means that for pre-3.0 code #GemConvertArrayBuilder should be set to true, and for post-3.0 code the parameter can be left unset or set to false/nil. If not correctly set, compiler errors will be generated on any #[ ] constructors.

Workaround:

Make sure #GemConvertArrayBuilder is set properly (true for pre-3.0 code, unset/nil/false for post-3.0 code).

Bug 45880 - Debugger cannot step into accessor methods

Product: GemStone/S 64 Bit

Versions: 3.6.3, 3.6.2, 3.6.1, 3.6, 3.5.7, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.x, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.4, 2.4.x, 2.3.1.6, 2.3.x, 2.2.5.4, 2.2.x, earlier versions

Impact: Informational

Methods that contain only a simple instance variable access cannot be stepped into. This limitation was introduced by design, to improve efficiency and simplify internal processing, although it does disrupt the flow of debugging.

Bug 45839 - Conversion from 6.7 fails version test

Product: GemStone/S 64 Bit

Versions: 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.4, 2.x

Fixed In: 2.4.7

The conversion process from 32-bit GemStone/S to GemStone/S 64 Bit includes version checking; this does not include checking for version 6.7, so conversion from GS/S 6.7 to GemStone/S 64 Bit versions earlier than 2.4.7 will error.

Workaround:

The conversion target 2.x version should be 2.4.7 or later.

Bug 45807 - IndexManager usageReport may get MNU #nextWord with Portable Streams installed

Product: GemStone/S 64 Bit

Versions: 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.2, 3.1.x, 3.0.x

Fixed In: 3.3

The code invoked by the IndexManager >> usageReport method calls PositionableStream >> nextWord. However, nextWord is implemented only in the legacy stream collections (PositionableStreamLegacy and subclasses), not in the portable streams (PositionableStreamPortable and subclasses). So in a system in which Portable Streams are installed, IndexManager >> usageReport encountered an MNU.

Systems that originate in versions earlier than 3.0 will have portable streams installed by default.

Bug 45744 - Inconsistent lock status after failed commit on temporary locked object

Product: GemStone/S 64 Bit

Versions: 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.3, 3.1.0.2, 3.1.0.1, 3.1, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.4, 2.4.x, 2.3.1.6, 2.3.x, 2.2.5.4, 2.2.x, 2.x, 1.x

Fixed In: 3.3

If you perform the following sequence:
  create a temporary object
  lock this temporary object using methods such as System writeLock:, etc.
  attempt a commit, but encounter a commit failure.
  abort the session

Then the lock on the temporary object remains, but is in a invalid state. The lock can't be removed, and methods to access lock status such as sessionLocks are inconsistent.

Workaround:

Removing the lock before aborting avoids problems. Using System removeLocksForSession allows you to clear the state.

Bug 45692 - Error message "Primitive number -1 (or -2) does not exist in the virtual machine" confusing

Product: GemStone/S 64 Bit

Versions: 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.2, 3.1.0.1, 3.0.1, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.3.1.6, 2.2.5.4

Platform All Platforms

Fixed In: 3.3

The following error message is incorrect:

```
  GemStone: Error Nonfatal
  a InternalError occurred (error 2083), Primitive number -1 does not exist in the virtual machine.
  Error Category: [GemStone] Number: 2083 Arg Count: 1
  Arg 1: -1
```

The negative number (-1 or -2) is *not* the number of the primitive being called but an internal status indicator:

  -1 = primitive does not exist

-2 = primitive should not be used

The correct message is:

Non-existent primitive called: status = -1. This usually indicates a mismatch between the smalltalk image and virtual machine.

One frequent cause of this error would be restoring a backup made on one version of GemStone into a system configured for a different version of GemStone.

Bug 45665 - Additional link flag needed when compiling on SUSE 12

Product: GemStone/S 64 Bit

Versions: 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.x, earlier versions

Fixed In: n/a

Impact: Informational

GCI applications get a link failure on SUSE 12.

An additional linker flag is required for compiling GCI applications on SUSE 12:

-Wl,-traditional

Bug 45654 - Cache Statistic PercentCpuIdle may have invalid value on AIX

Product: GemStone/S 64 Bit

Versions: 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.3, 3.1.0.2, 3.1.0.1, 3.1, 3.0.1, 3.0, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.x, 2.3.1.6, 2.3.x, 2.2.6.3, 2.2.6, 2.2.5.4, 2.2.x, 2.x

Fixed In: 3.2.9, 2.4.7

The calculation for the cache statistic PercentCpuIdle is subject to a rounding error, which results in values of -1 being recorded in the statmonitor output data.

VSD displays this as an unsigned int, incorrectly displaying a value of 4294967295.

Bug 45653 - Statmonitor can record session with PID = 4294967295 (-1)

Product: GemStone/S 64 Bit

Versions: 2.3.1.7, 2.3.1.6, 2.2.5.4

Platform All Platforms

Fixed In: 2.4

There is a small window of time after a session has logged out when statmonitor can inadvertently record stats for a session with the same session ID but with a process ID of 4294967295, which is actually a -1 treated as an unsigned integer.

These can be safely ignored.

Bug 45604 - startcachewarmer crashes if stone name argument is given an NRS

Product: GemStone/S 64 Bit

Versions: 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.3, 3.1.0.2, 3.1.0.1, 3.1, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.x, 2.3.1.6, 2.3.x, 2.2.5.4, 2.2.x, 2.x

Fixed In: 3.3

While many GemStone arguments accept an NRS instead of a stone name, the startcachewarmer script requires the stone name only. Using an NRS for the -s stone name argument results in a HostCallDebugger.

In v3.2.8.1 and later, it will not crash, but will not report an error or perform cache warming.

Workaround:

Ensure the stone name is used, not an NRS.

Bug 45586 - Possible SEGV on execution of method upgraded from 3.1.x and not recompiled

Product: GemStone/S 64 Bit

Versions: 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2

There are minor changes in the bytecode generated from method compliation in v3.1.x and 3.2.x, related to the changes that allow a step point at the beginning of a method.

A case has been observed in which a SEGV in a Gem is apparently related to this difference in the bytecodes. While this is a very rare situation and is not believed to be at risk for causing any other problems, for reliability it is recommended that all application methods be recompiled as part of the 3.1.x to 3.2.x upgrade process.

Workaround:

Recompile the method.

Bug 45574 - Delay issues with Softbreak

Product: GemStone/S 64 Bit

Versions: 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.0.4, earlier versions

Fixed In: 3.3

There are a number of ways in which SoftBreak does not interrupt Delays correctly or otherwise behaves incorrectly. The specific behavior varies between 3.1.x and 3.2.x and is affected by other circumstances, such as linked vs. RPC topaz.

Bug 45566 - SUSE 11 using older glibc; LDAP not usable

Product: GemStone/S 64 Bit

Versions: 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5

Platform Linux SUSE 11

As of version 3.2.5, GemStone was compiled on a platform using glibc version 2.12. However, SUSE v11 sp3 uses an older version of glibc, v2.11. The GemStone LDAP libraries have a dependency on a symbol in glibc v2.12; attempting to load the LDAP libraries on a system using glibc v2.11 encounters a library load error.

As a result, LDAP is not usable on SLES 11SP3 with glibc 2.11.x. We believe that this is the only issue caused by the older glibc version; our tests have shown no other issues.

SLES 11 has been removed from the list of certified platforms.

Bug 45518 - encodeUsing: with nil argument resulted in SEGV

Product: GemStone/S 64 Bit

Versions: 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.3, 3.1.0.2, 3.1.0.1, 3.1, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.4, 2.4.x, 2.3.1.6, 2.3.x, 2.2.5.4, 2.2.x

Fixed In: 3.2.8

Sending encodeUsing: to a kind of String or MultiByteString with an argument of nil resulted in a SEGV.

Bug 45473 - Hot standby connection may timeout with large number of tranlogs

Product: GemStone/S 64 Bit

Versions: 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.x

Fixed In: 3.3

There is a 5-second timeout in the internal code that scans the received tranlog files. With a very large number of tranlogs in the master systems tranlog directory, this timeout may be exceeded, which causes the logsender-logreceiver connection to report a timeout.

Workaround:

Periodically move or remove older tranlogs.

Bug 45447 - Restore error 4049/"unable to re-originate log" is uninformative

Product: GemStone/S 64 Bit

Versions: 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.3, 3.1.0.2, 3.1.0.1, 3.1, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.x, 2.3.1.6, 2.3.x, 2.2.5.4, 2.2.x, 2.x, 1.x

Fixed In: 3.2.8

When a tranlog restore finds a tranlog of the intended name already exists, it produces this error message, which was not useful to understand the problem.

Workaround:

Ensure you do not have extra existing tranlogs in the tranlog directories for the repository into which you are restoring.

Bug 45409 - Object >> isCommitted may return true after failed commit

Product: GemStone/S 64 Bit

Versions: 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.3, 3.1.0.2, 3.1.0.1, 3.1, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.x, 2.3.1.6, 2.3.x, 2.2.5.4, 2.2.x, 2.x, 1.x

Fixed In: 3.2.7

After a failed commit, temporary objects that would have been committed are in a special state, normally resolved by abort or by a retry of the commit. After the failed commit but before further commit or abort, sending #isCommitted to these temporary objects returns true, which is incorrect since the objects were not actually committed.

Workaround:

After a failed commit, application code must resolve the issues in order to either commit successfully or abort (or log out). The code that resolves the commit issues should not rely on the results of isCommitted.

Bug 45403 - Symbol garbage collection not usable in v3.2.x

Product: GemStone/S 64 Bit

Versions: 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2

Platform All

Fixed In: 3.3

Symbol garbage collection is enabled by setting STN_SYMBOL_GC_ENABLED to TRUE, in versions 3.1 and later. By default, this is false. There are bugs in the primitive code supporting this option; it is unlikely to complete correctly with a non-trivial number of symbols or with some corner cases of symbol references.

Workaround:

Do not enable symbol garbage collection; while there is slight overhead, GemStone will work correctly with unused symbols. Avoiding the creation of unnecessary symbols remains best practice, as in previous versions.

in v3.2.7, STN_SYMBOL_GC_ENABLED may not be set to TRUE.

Bug 45386 - GsHostProcess class >> fork: does not handle single quotes

Product: GemStone/S 64 Bit

Versions: 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2

Fixed In: 3.3

GsHostProcess class >> fork: accepts a String argument containing a command line. This string may include quotes per the command line syntax; double quotes were handled correctly, but using single quotes resulted in syntax errors.

For example:

```
  GsHostProcess fork: '/bin/bash -c ''echo foo/bar'''
```

resulted in:

```
  foo/bar': -c: line 0: unexpected EOF while looking for matching `''
  foo/bar': -c: line 1: syntax error: unexpected end of file
```

Bug 45367 - StringKeyValueDictionary at:ifAbsent and at:otherwise: returns an error on a nil key

Product: GemStone/S 64 Bit

Versions: 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.2, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.3.1.6, 2.2.5.4

Fixed In: 3.3, 3.2.13

The StringKeyValueDictionary methods at:otherwise: and at:ifAbsent: provide handling instructions for cases in which the lookup value is not found (respectively, a value and a no-argument block that returns a value). However, for a lookup argument of nil, instead of following the instructions, these methods throw an invalid argument error.

Bug 45320 - Debugging: Step Thru on an interval #do: will skip down to the end of the code

Product: GemStone/S 64 Bit

Versions: 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.2, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.3.1.6, 2.2.5.4

Platform All Platforms

Fixed In: 3.3

If you are debugging code in topaz that contains an interval #do: loop like the following:

```
. . .
(1 to: 3) do: [ ... loop code ... ].
... subsequent code ...
```

And step through the loop code using the topaz command "step thru", at the end of the loop code, rather than stepping into the subsequent code you will instead complete execution of the subsequent code without stopping.

Workaround:

Replace interval #do: loops like:

```
(1 to: 3) do: [ ... ]
```

with a #to:do: loop like:

```
1 to: 3 do: [ ... ]
```

Bug 45310 - performOnServer: handling of non-ASCII arguments

Product: GemStone/S 64 Bit

Versions: 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.3, 3.1.0.2, 3.1.0.1, 3.1, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.x, 2.3.1.6, 2.3.x, 2.2.5.4, 2.2.x, 2.x, 1.x

Fixed In: 3.3

System class >> performOnServer: accepts a string, sends it to the OS for execution, and passes the result back to GemStone Smalltalk. When arguments are outside the ASCII range, GemStone must perform operations on the string in order to get correct behavior.

As GemStone's handling of extended Character data has changed over recent releases, the behavior has changed. While versions before 3.3 do not accept arguments with Characters with codepoints over 255, in 2.x it was possible to send encodeAsUTF8 to the entire command, while for 3.x this does not work, and you would need to compose a kind of string containing the UTF8 encoded characters.

Workaround:

If you rely on server operations that may require non-ASCII characters to be used for arguments to performOnServer:, your uses of performOnSever: should be tested for each significant upgrade, to ensure that any special handling is added or removed.

Bug 45295 - In partial logging mode, not all tranlogs may be deleted

Product: GemStone/S 64 Bit

Versions: 3.6.3, 3.6.2, 3.6.1, 3.6, 3.5.7, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.x.0, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.x, 2.3.1.6, 2.3.x, 2.2.5.4, 2.2.x, earlier versions

In partial logging mode (STN_TRAN_FULL_LOGGING =FALSE), transactions smaller than STN_TRAN_LOG_LIMIT are written to the tranlog (larger transactions trigger a checkpoint). If the period starting from a begin_data until commit completes spans the point where a new tranlog is started, the previous tranlog cannot be deleted at the time the new log is started. The way tranlog deletion is handled does not allow later automatic deletion, so the old tranlog is remains until manually removed.

Workaround:

You will need to periodically check and manually delete leftover tranlogs.

Partial logging is not intended for production use, since it does not provide the ability to restore from backup and apply tranlogs. While partial logging mode allows you to recover from minor problems, if there is a serious problem such as repository corruption, the system may not be recoverable if using partial logging mode.

Bug 45268 - Stone or netldi names that match existing service names may cause confusion

Product: GemStone/S 64 Bit

Versions: 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.3, 3.1.0.2, 3.1.0.1, 3.1, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.4, 2.4.x, 2.3.1.6, 2.3.x, 2.2.5.4, 2.2.x, 2.x, 1.x

Fixed In: 3.3

Impact: Informational

Service names that require ports, such as gs64stone and gs64ldi, look for assigned port numbers in the OS services database. If a name is used that is already defined by the system, such as "auth", it will find this port, which may not be usable by the GemStone process.

Bug 45253 - GemConnect library load problems on Darwin

Product: GemConnect

Versions: 2.4, 2.3

Platform Darwin

Due to a bug in the 10.2.0.X versions of the Oracle libraries on Darwin, attempting to load the GemConnect liboraapi23-643.dylib library on GemStone/64 versions 3.2.13 and later will fail with a SIGSEGV.

Workaround:

Use 11.2 or later versions of the Oracle libraries when running GemConnect on Darwin.

Bug 45249 - topaz "shell" command does not work with lineeditor

Product: GemStone/S 64 Bit

Versions: 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.x

Fixed In: 3.3

The lineeditor feature in topaz, when enabled (as it is by default), makes the shell command window unresponsive.

Workaround:

If you use the SHELL command, disable lineeditor using:

```
topaz> OMIT LINEEDITOR
```

Bug 45227 - markGcCandidates does not scan pages correctly and should not be used

Product: GemStone/S 64 Bit

Versions: 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2

The internal phases in markGcCandidates were numbered incorrectly with respect to the object table sweep phase number, resulting in some phases sweeping pages that incorrectly include scavengeable pages.

Workaround:

Do not use markGcCandidates in affected versions; use markForCollection, which works correctly. markGcCandidates has been removed in v3.3.

Bug 45190 - findAllReferencePathsToObjects: and memory use

Product: GemStone/S 64 Bit

Versions: 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 2.4.8, 2.4.7, 2.4.6

Fixed In: 3.2.7

The Repository findAllReferencePathToObject: (and variants), return all paths to a given object or objects. For a highly connected object, this can result in a very large return set and a high demand for memory in internal computations, and Gem OutOfMemory errors. The memory demands may be higher than can be accommodated even with increased temporary object memory.

Workaround:

Increasing the Gem's GEM_TEMPOBJ_CACHE_SIZE may or may not allow your query to complete.

For highly connected objects, start by using findReferencePathToObject:. If you are trying to remove references to an object, disconnecting references close to the target object along the reference paths may significantly reduce the number of paths, depending on the graph.

You may be able to modify the limit set to reduce the number of paths.

Later versions include maxPaths: to allow control over the number of paths returned in one sweep.

Bug 45188 - quickObjectAuditWithLevel: reports of objects referencing invalid oops later reported nonexistent

Product: GemStone/S 64 Bit

Versions: 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.3, 3.1.0.2, 3.1.0.1, 3.1, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4, 2.4.3, 2.4.2, 2.4.1, 2.4, 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.5, 2.2.4, 2.2.3, 2.2

Platform All Platforms

Fixed In: 3.2

One of the short-cuts taken by quickObjectAuditWithlevel: that allow it to run faster than a normal objectAudit: has a side-effect that will cause objects reported as referencing invalids oops to also be reported as non-existent. For example, you might see something like this:

```
--------------------
object XXXX oops[N] has value ZZZZ which is not a valid oop
...
GemStone is now beginning a detailed re-scan of the Repository to obtain specific diagnoses of the above problems.
```

```
...
Object YYYY, of class C, at 1-based offset N, references nonexistent object XXXX
----------------------
```

In this case, you can safely ignore the report that object XXXX is nonexistent.

Bug 45168 - Number >> fromString: handling of exponent letters is inconsistent and incomplete

Product: GemStone/S 64 Bit

Versions: 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.3, 3.1.0.2, 3.1.0.1, 3.1, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.x, 2.3.1.6, 2.3.x, 2.2.5.4, 2.2.x, 2.x, 1.x

Fixed In: 3.2.7

Number and its subclasses inherit or implement fromString:, which creates instances of the appropriate subclass from the argument string. Not all legal floating point syntax was understood, and some not strictly legal syntax was accepted.

Legal syntax is provided in the GemStone BNF, in $GEMSTONE/doc/bnf.txt.

Bug 45158 - removedbf not usable for files on remote nodes

Product: GemStone/S 64 Bit

Versions: 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, earlier versions

Impact: Informational

The removedbf utility includes options that should allow it to work remotely, but these options are not correctly used, making it difficult for removedbf to work correctly for a non-local partition or disk file.

Workaround:

Run removedbf on the remote node. removedbf is no longer supported on non-local raw partitions or file systems disks.

Bug 45058 - Memory page cache statistics incorrect on AIX

Product: GemStone/S 64 Bit

Versions: 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.3, 3.1.0.2, 3.1.0.1, 3.1, 3.0.1, 3.0, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.4.3, 2.4.4, 2.4.3, 2.4.2, 2.4.1, 2.4, 2.3.1.6, 2.3.1, 2.3

Platform AIX

Fixed In: 3.2.5, 2.4.7

The following cache statistics are expressed on AIX as the number of 4KB pages, rather than KB, and so are understated by 4x in the statmonitor data.

```
  DataRSS
  TextRSS
  DataVmSize:q
```

Bug 45051 - System Cache Statistics for network on Linux have incorrect values

Product: GemStone/S 64 Bit

Versions: 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.3, 3.1.0.2, 3.1.0.1, 3.1, 3.0.1, 3.0, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.4.3, 2.4.4, 2.2.6, 2.x

Platform Linux

Fixed In: 3.2.5, 2.4.7

There was an error in the code that parsed the contents of /proc/net/dev, which provides the values for cache statistics collected on linux when statmonitor options to collect network stats are specified. This resulted in incorrect values reported for statistics such as PacketsOutboundErrors and PacketsReceivedErrors, which were too high, and others such as InputKBytes and OutputKBytes, which were zero.

Bug 45040 - Read authorization checks happen when object faulted into memory

Product: GemStone/S 64 Bit

Versions: 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.3, 3.1.0.2, 3.1.0.1, 3.1, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.3, 2.4.5.2, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.4.3, 2.4.4.2, 2.4.x, 2.4

Platform All Platforms

Fixed In: 3.2.5

In GS/S 32-bit, read authorization checks were made when an object was actually accessed, such as reading an instvar or element, or checking some attribute of the object, such as its size or class. With the re-introduction of object security in GS/64 2.0, read authorization checks now occur when an object is faulted into the VM memory. This can cause 2115 #authErrSegRead read authorization errors (SecurityError) to be generated earlier in code execution than in GS/S 32-bit.   This can in turn cause exception handlers to not be triggered properly because they do not cover the point of object faulting.

Workaround:

Refactor 2115 #authErrSegRead (SecurityError) exception handlers to cover the new location of error triggering.

Bug 45008 - Connection refused errors on NetLDI connect backlog

Product: GemStone/S 64 Bit

Versions: 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.3, 3.1.0.2, 3.1.0.1, 3.1, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.4.3, 2.4.4, 2.2.6, 2.x

Fixed In: 3.2.5

On login, all gems connect to the NetLDI on its listening socket, the configured port for the NetLDI. The backlog for this socket is set at 20. If there are a large number of simultaneous login requests, this backlog may be exceeded, and further logins will error with "Connection refused".

Workaround:

Retrying the login should succeed. Avoid initiating a large number of logins at the same time.

Bug 44961 - RcKeyValueDictionary at:ifAbsent and at:otherwise: returns an error on a nil key

Product: GemStone/S 64 Bit

Versions: 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.2, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.3.1.6, 2.2.5.4

Platform All Platforms

Fixed In: 3.2.7

The following two methods return the error #rtErrNilKey (error 2090) when the specified key is nil. Since these methods are intended to provide alternate handling of nil keys, this is incorrect behavior.

#at:ifAbsent:
#at:otherwise:

Workaround:

As SystemUser file-in and commit the following code.

```
category: 'Accessing'
method: RcKeyValueDictionary
at: aKey ifAbsent: aBlock

"Returns the value that corresponds to aKey.  If no such key/value pair exists,
 returns the result of evaluating the zero-argument block aBlock."

| hash |
aKey == nil ifTrue: [ ^ aBlock value ].

hash := self hashFunction: aKey.
^ (self _at: hash) at: aKey ifAbsent: aBlock
%

category: 'Accessing'
method: RcKeyValueDictionary
at: aKey otherwise: aValue

"Returns the value that corresponds to aKey.  If no such key/value pair exists,
 returns the given alternate value."

| hash |
aKey == nil ifTrue: [ ^ value ].

hash := self hashFunction: aKey.
^ (self _at: hash) at: aKey otherwise: aValue
%
```

Bug 44903 - System class >> standbyVmSessionId removed

Product: GemStone/S 64 Bit

Versions: 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2

Platform All Platforms

The method System class >> standbyVmSessionId was intended to be deprecated, but was removed from the product. In addition, the non-functional methods   startStandbySession and stopStandbySession were also removed.

Note that while the Smalltalk methods were removed, the underlying zero argument primitives called by them are still present. Customers are cautioned *not* to attempt to restore these commands by directly calling _zeroArgPrim: with either arguments 149 or 15 0. It is still safe to call primitive 151 directly to retrieve the standby session ID.

Workaround:

Use OS level shell scripts to execute appropriate startlogsender / startlogreceiver commands.

If you need to retrieve the session ID of the standby session, you can execute:

System _zeroArgPrim: 151

or alternatively, restore the method standbyVmSessionId by logging in as SystemUser and entering/committing the following code:

```
classmethod: System
standbyVmSessionId

"Return the session ID of the logsender or logreceiver session
 associated with this stone's role in a hotstandby pair."

^ self _zeroArgPrim: 151

%
```

Bug 44900 - Running out of disk space during fullBackupCompressedTo:MBytes: can trigger SIGSEGV

Product: GemStone/S 64 Bit

Versions: 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.2, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.3.1.6, 2.2.5.4

Platform All Platforms

Fixed In: 3.2.4

If you run out of disk space on the device being used for a compressed backup while executing "SystemRepository fullBackupCompressedTo: <file> MBytes: <size> you may trigger a SIGSEGV rather than the expected out of disk space error.

Workaround:

Re-run the compressed backup on a device with adequate disk space.

Bug 44871 - The gemsetup.sh and gemsetup.csh scripts append at end of path

Product: GemStone/S 64 Bit

Versions: 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.3, 3.1.0.2, 3.1.0.1, 3.1, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4x, 2.3.1.6, 2.3.1, 2.2.5.4, 2.2.x, 2.x, 2.4x, 1.x

Platform All

Fixed In: 3.3

Gemsetup.sh and gemsetup.csh are scripts that can be used to setup the environment for your version of GemStone. However, when you use these scripts, they add the GemStone version at the end of the current path. If you have an installation for a different version of GemStone already on your path, when you attempt to execute e.g. startstone, startnetldi, or topaz, you will get the old version instead of the new version.

Workaround:

Make sure there are no other GemStone installations in your path when you use gemsetup, or manually add the GemStone installation at the beginning of the path.

Bug 44862 - Pre 2.4.5 keyfiles on 2.4.5 and later will not allow remote gems

Product: GemStone/S 64 Bit

Versions: 2.4.8, 2.4.7, 2.4.6, 2.4.5.3, 2.4.5.2, 2.4.5.1, 2.4.5

Platform All Platforms

Fixed In: 3.0

A new field was added to the keyfile in 2.4.5 that checks for remote gem access to the stone. Older keyfiles can still be used in 2.4.5 and later versions (prior to 3.0), but since this field appears as FALSE on these older keyfiles, attempts to run a remote gem will trigger the following error during login:

GemStone: Error Fatal
Remote logins to this repository are disabled in the keyfile.

Workaround:

Contact GemTalk Technical Support for a new keyfile.

Bug 44786 - GsFile non-ASCII String filenames not correctly handled

Product: GemStone/S 64 Bit

Versions: 3.2.17, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.3, 3.1.0.2, 3.1.0.1, 3.1, 3.0.1, 3.0

Platform All Platforms

Fixed In: 3.3

GsFile operations involving filenames containing non-ASCII characters in a unix environment using a UTF8 Locale must be handled using UTF8 instances. If a String containing non-ASCII characters is used, the operation will either fail, or in the case of file creation, will create a file with an incorrectly formatted file name.

For example:

GsFile open: <StringWithNonAsciiCharacters> encodeAsUTF8

Bug 44783 - GEM_PGSVR_COMPRESS_PAGE_TRANSFERS does not compress Gem->PageServer

Product: GemStone/S 64 Bit

Versions: 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.3, 3.1.0.2, 3.1.0.1, 3.1, 3.0.1, 3.0

Fixed In: 3.3

The Gem configuration option GEM_PGSVR_COMPRESS_PAGE_TRANSFERS, when true, compresses page transfers from the Page Server to the Gem. However, it does not compress transfers in the other direction, from the Gem to the Page Server.

Bug 44685 - postConv only automatically converts sortBlocks that are SimpleBlocks

Product: GemStone/S 64 Bit

Versions: 3.6.3, 3.6.2, 3.6.1, 3.6, 3.5.7, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.x, 3.0.1, 3.0

Impact: Informational

Blocks in 2.x applications may be simple or complex. Complex blocks are created when the block code includes references to the block's context, or in the case of certain syntactical features. For such blocks, automatic recompliation may not be safe. postConv attempts to recompile the sort blocks of persistent SortedCollections in the application, but if any of these blocks are complex, they will not be converted, and will be reported as errors from portConv.

Note that you may have many instances of SortedCollection that reference the same sortBlock, such as when the sortBlock is in a method.

Workaround:

You will need to manually update the sortBlocks of persistent SortedCollections. For instances returned in the FailedSortedCollections bitmap, perform code of the form:

```
    aSortedCollection sortBlock: (recompiled sortBlock).
```

To determine the cause of specific sortBlock recompile failures, examine the source code for the SortedCollection's sortBlock. If there are reference to method context (such as self, or a variable), this may require defining a special method to provide that context.

Since sortBlocks are executed multiple times for each addition to the collection, it is recommended to keep the sortBlocks as simple and performant as possible is recommended; complex operations should be done outside of the sortBlock.

For further assistance, contact GemTalk Technical Support.

Bug 44664 - Linked logins that inadvertently appear to become RPC

Product: GemStone/S 64 Bit

Versions: 3.6.3, 3.6.2, 3.6.1, 3.6, 3.5.7, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.x, earlier versions

Impact: Informational

Topaz, and GemStone sessions in general, have an RPC interface supporting only multiple RPC sessions, and a linked interface that supports one linked and multiple RPC sessions.

To invoke the linked version, the topaz -l flag is used, which leaves the gemnetid field empty or set to a placeholder, this is used during login to indicate that the session should be linked. Subsequently, you can set the gemnetid (to gemnetobject or equivalent) to login an additional RPC session.

However, if you have a .topazini that sets gemnetid, the .topazini is executed as usual when topaz -l is started. This sets the gemnetid so the first login is RPC, not linked.

Workaround:

In v3.4 and later, the -L option instead of -l suppresses set gemnetid statements in topazini files. It is recommended to use this option when you need to start a linked topaz, to avoid any risk from topazini files.

Use caution in specifying a gemnetid in a topazini file.

Using separate topazini files and using -I to explicitly provide the filename, or -i to suppress the default topazini, can avoid problems.

If you are expecting a linked login, note a prompt such as

```
    topaz 2 >
```

which indicates an RPC login.

Bug 44657 - Mis-configured 6x to 2x conversion can fail without useful error messages

Product: GemStone/S 64 Bit

Versions: 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.4.3, 2.4.4.2, 2.4.4.1, 2.4.4, 2.4.3, 2.4.2.1, 2.4.2, 2.4.1, 2.4, 2.3, 2.2.3, 2.2.2, 2.2.1, 2.2, 2.1.1, 2.1, 2.0.1, 2.0

Platform All Platforms

Fixed In: 2.4.7

In 6x to 2x conversion, if you accidentally specifies the 64-bit extents of the 2.X stone as the source extents for the conv6xTo2x phase (by specifying the incorrect system.conf file in the command line, or by specifying the 64-bit extents in the system.conf rather than the 32-bit extents), then the conv6xTo2x step will fail with the report:

```
conv6xTo2x[ERROR]: conversion failure found. Check log <$upgradeLogDir>/convertGem-0.log conv6xTo2x[ERROR]: Errors found in conversion. Check logs listed above.
```

With the following entry in the stone log:

```
UTL_GUARANTEE failed, File /export/toronto2/users/buildgss/gss64-24x.1/src/stnlo op.c line XXXXX (line number may vary)
... [stack info] Error: premature logout/death of conversion session with processId XXXXX
```

But checking the indicated convertGem-0.log doesn't show any problems:

```
...
[Info]: Total pages needing conversion: XXXX
[Info]: Pages to be converted by this session: XXXX
[Info]: Starting PageId=X EndingPageId=XXXX
--- <timestamp> :
[Info]: Finished computing pages to convert.
[Info]: Ready to open source repository extent(s). GDBG> Leaving GEM debugger. Normal termination of stand-alone GEM
[Info]: Logging out...
```

Workaround:

Fix the configuration to properly separate the 32-bit extents from the 64-bit extents.

Bug 44641 - Inconsistent behavior in comparisons of Strings and Symbols with same contents

Product: GemStone/S 64 Bit

Versions: 3.6.3, 3.6.2, 3.6.1, 3.6, 3.5.7, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.x, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.4.3, 2.4.x, 2.4, 2.3.1.6, 2.3.x, 2.3, 2.2.5.4, 2.2.x, earlier versions

Impact: Informational

When a String and a Symbol have the same contents, equality and less/greater than return false, while less/greater than or equal returns true. Since intuitively, less/greater than or equal imples that either one or the other of less/greater than and equal is true, this is inconsistent.

E.g.:

```
'abc' = #abc --returns false
'abc' < #abc  --returns false
'abc' <= #abc  --returns true
```

VisualWorks and VA Smalltalk also have this behavior; to avoid perturbing existing code, GemTalk does not intend to change the behavior.

Workaround:

Avoid comparisons that mix Strings and Symbols. If you must perform comparisons that mix instances of different classes and perform comparisons and include elements with identical contents, carefully consider the operator you use to ensure that the results are what you intend.

Mixing Strings and Symbols with identical contents in a SortedCollection with a default SortBlock, which uses <=, will not be ordered deterministically; the ordering will depend on the order elements are added.

Bug 44625 - KeyvalueDictionary removeKey:* on nil doesn't always return an error

Product: GemStone/S 64 Bit

Versions: 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.3, 3.1.0.2, 3.1.0.1, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.x, 2.3.1.6, 2.3.x, 2.2.5.4, 2.x

Platform All Platforms

Fixed In: 3.2.3

Nil is not a valid value for a key in a KeyValueDictionary, and attempting to add it will trigger a #rtErrNilKey (2090) error. You *should* get this error when attempting to remove a nil key. But frequently the key-not-present behavior will happen instead:

For #removeKey:ifAbsent: The ifAbsent block will be executed. For #removeKey:otherwise: The otherwise value will be returned.

Bug 44617 - CR backlog during write-set union sweep

Product: GemStone/S 64 Bit

Versions: 3.6.3, 3.6.2, 3.6.1, 3.6, 3.5.7, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.x

Platform All

Impact: Informational

To make sure necessary information is preserved, the stone "pins" the commit record in effect at the start of the Admin Gem write-set union sweep during garbage collection (stone stat GcVoteState = 4) and prevents this commit record from being disposed until the sweep is completed. Normally the sweep completes quickly, but if the number of possible dead (stone stat PossibleDeadObjs) and the size of the write-set union (stone stat GcWsUnionSize) are high, then a CR backlog may develop.

Note that this is independent of the commit record actually held by the Admin Gem itself, so the GcGem may not be reported as holding the oldest commit record (stone stats OldestCrSession, OldestCrSessNotInTrans).

Workaround:

Reducing the time it takes to complete the write set union sweep, and the commit rate while it is running, will allow you to avoid problems.

Running markForCollection more frequently will result in smaller result sets that will allow faster write set union sweep. Time markForCollection to allow the sweep to run during periods of no or light application load. Configure the sweep to use as many sessions as your environment can support, using the following instructions:

You begin by specifying a maximum number of threads allowed by executing and committing as GcUser:

UserGlobals at: #sweepWsUnionMaxThreads put: X

Where X is the maximum allowed value between 1 and 8.

Next, you'll need to shutdown and restart the Admin GcGem so it picks up the new value:

System stopAdminGcSession. System startAdminGcSession.

Once the new Admin Gcgem is running, check the number of *actual* threads using:

SystemRepository mtThreadsLimit: <sessionId>

Where <sessionId> is the session ID of the Admin GcGem.

You can then adjust this dynamically up-and-down (to the max specified) by using:

SystemRepository mtThreadsLimit: <sessionid> setValue: X

WARNING: having additional threads running may adversely impact system performance -- monitor carefully and be prepared to reduce the thread count if the load becomes too high.

Bug 44594 - Equivalent Arrays and Intervals are not equal

Product: GemStone/S 64 Bit

Versions: 3.6.3, 3.6.2, 3.6.1, 3.6, 3.5.7, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.x, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.x, 2.x, 1.x

An instance of Array such as #(1 2 3 4 5) and an instance of Interval such as (1 to: 5) represent the same collection contents, and as such, should logically be equal. However, expressions such as #(1 2 3 4 5) = (1 to: 5) returned false.

Bug 44590 - nbstep command in topaz returns error

Product: GemStone/S 64 Bit

Versions: 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2

Platform All

Fixed In: 3.3

The nbstep command in topaz supports non-blocking command line debugging. Using this command encounters a message not understood error.

Workaround:

The step command allows blocking debugging.

Bug 44556 - filein older code can introduce comment class method and confuse class comment handling

Product: GemStone/S 64 Bit

Versions: 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.2, 3.1.0.1, 3.1

Fixed In: 3.3

Impact: Informational

An option for Class comments in versions earlier than 3.1 was a class method with the selector #comment. In some versions of GBS, such a method was explicitly created when the GBS tools were used to create a class comment.

This changed in v3.1, which stores class comments in an instance variable in the class. In v3.1 and higher, it is explicitly incorrect for classes to implement class methods named comment or comment:. The upgradeComments script cleans up such cases in the image, moving the text of these methods to the Class comment variable.

However, if you have code filed out from versions earlier than 3.1, and file this into 3.1 or later, you may introduce the comment class method. While this does not cause serious problems, it overrides normal class comment handling; edits to class comment handling via normal tools will never be visible.

Workaround:

Check for comment methods after filing in older code.

Bug 44545 - GsQuery>>reversedReadStream not implemented

Product: GemStone/S 64 Bit

Versions: 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2

Fixed In: 3.3

The method GsQuery>>reversedReadStream was added with 3.2, but the supporting code is not implemented. This method returns a does-not-understand error.

Bug 44478 - Blocks may return wrong values when nested variable is assigned in outer scope

Product: GemStone/S 64 Bit

Versions: 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.3, 3.1.0.2, 3.1.0.1, 3.1, 3.0.1, 3.0

Fixed In: 3.3

A block that is nested within another block can, in some cases, have the wrong value for a variable that is declared in an outer scope. An assignment to the variable performed in the outer scope is not always propagated to the inner block when it should be.

For example, the following returns 1, and should return 100:

```
[ | temp myBlock |
 temp := 1.
 myBlock := [ temp ].
 temp := 100.
 myBlock value ]
   value
```

Bug 44458 - searchlogs script requires OOP argument, fails on sessionid and client

Product: GemStone/S 64 Bit

Versions: 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2

Fixed In: 3.4

The searchlogs script allows you to search the tranlogs for transactions involving a particular OOP or OOPs, filtering using a number of criteria such as User ID and client host. This script fails (does not return results) if the OOP is not included in the command.

sessionId does not respect the OOP argument, and will return all results that match the filter regardless of any OOP/s specified.

For client search, it also requires the client be in IPV6, that is, ::ffff:N.N.N.N.

Workaround:

When searches do not require an OOP, the printlogs script will allow you to search tranlogs for records using filters.

Bug 44433 - Compiling method into an Array requires #OtherPassword privilege

Product: GemStone/S 64 Bit

Versions: 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.3, 3.1.0.2, 3.1.0.1, 3.1, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.x, 2.3.1.6, 2.3.x, 2.2.5.4, 2.2.x, 2.x

Fixed In: 3.3

When compiling a method (using compileMethod:dictionaries:category:), with the target dictionaries: argument being an instance of Array, the #OtherPassword privilege was required. This is not correct; this should require #CodeModification, but not #OtherPassword.

Bug 44413 - Session starting a mid-cache may hang on cache startup error

Product: GemStone/S 64 Bit

Versions: 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.3, 3.1.0.2, 3.1.0.1, 3.1.X, 3.0.1, 3.0, 3.0.X, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.X, 2.3.1.6, 2.3.X, 2.2.5.4, 2.2.X, 2.x

Fixed In: 3.2.2

In certain configurations, if the netldi is unable to create a mid-level cache, the gem that is attempting to use that mid-level cache may hang, rather than timeout and error.

Bug 44411 - GsSocket >> isConnected, peerAddress, and peerPort signal exceptions on error

Product: GemStone/S 64 Bit

Versions: 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.3, 3.1.0.2, 3.1.0.1, 3.1, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.4.3, 2.4.x, 2.3.1.6, 2.2.6, 2.2.5.4, 2.x

Fixed In: 3.2.2

The method comments for GsSocket instance methods #isConnected, #peerPort, and #peerAddress state that they return nil if an error occurs. These methods incorrectly signal an error if an error occurs, rather than returning nil.

In v3.2.2 and later, if an error occurs than #isConnected returns false, and #peerAddress and #peerPort, return nil.

Workaround:

Use an exception handler.

Bug 44375 - Exception handler #resume from primitive operation returns nil

Product: GemStone/S 64 Bit

Versions: 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2

Platform All Platforms

Fixed In: 3.3

If an exception handler is triggered from within low-level C code in a primitive, nil is incorrectly returned as the result of the Smalltalk method that called the primitive.

For example, if an exception handler is defined for AlmostOutOfMemory that exits using #resume, and that exception is triggered by the concatenate operator #, (which calls primitive 615) from the code:

result := string1 , string2.

The result will be nil.

Workaround:

Refactor code so that problematic operations are isolated in their own block and can be restarted from the beginning, so you can use #retry in the exception handler code that covers the block.

Bug 44298 - startcachewarmer not usable on remote caches

Product: GemStone/S 64 Bit

Versions: 3.2, 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.3, 3.1.0.2, 3.1.0.1, 3.1, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.4.3, 2.4.4, 2.4.3, 2.4.2, 2.4.1, 2.4, 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.x, 2.2, 2.1.4, 2.1.x, 2.1, 2.0.5, 2.0.4, 2.0.3, 2.0.x, 1.2.4, 1.2.x, 1.1.x, 1.x

Fixed In: 3.2.1

Impact: Informational

The startcachewarmer script loads pages into a shared page cache. This is limited to the Stone's shared page cache, and does not work with remote shared page caches.

In v3.2.1 and later, the startcachewarmer script include the -H option to specify the Stone's host, and other changes to allow use on remote caches.

Bug 44265 - Hot and warm standby subject to delay for large reclaim operations

Product: GemStone/S 64 Bit

Versions: 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.3, 3.1.0.2, 3.1.0.1, 3.1.X, 3.0.1, 3.0, 3.0.X, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.X, 2.3.1.6, 2.3.X, 2.2.5.4, 2.2.X, 2.x

Fixed In: 3.3

Replaying of transaction logs may be delayed if a large reclaim occurs and required OOPs are in late in the page ordering.

A hot or warm standby system replays transaction logs from a primary stone, in sequence, including operations such as reclaims. If there is a large reclaim in the primary stone, this reclaim must also be replayed in the standby, in order to reclaim the OOPs. If these OOPs are reused and referenced in later transaction logs, the reclaim of these objects on the standby system must complete. This may take some time, during which time the standby will not be up to date. While most such delays are short, for large reclaims on large systems this may cause a period in which the standby is not up to date.

Workaround:

Ensure that reclaims complete as quickly as possible. More frequent, smaller reclaims, such as by running MFC more frequently, will complete more quickly. If you have throttled reclaim on the primary, this will also cause reclaim to be throttled and slower on the standby system, since changes in GC configuration are transmitted to the standby along with application changes.

Bug 44202 - Change in behavior for = with Unicode string classes

Product: GemStone/S 64 Bit

Versions: 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.1

Impact: Informational

When comparing Unicode strings that for which all printing characters are the same, but in which there are difference in non-printing characters, previously the String would compare as false. In version 3.2 and later, such UnicodeStrings compare as equal.

For example, the following:

```
  (Unicode16 with: $a with: $b) = (Unicode16 with: $a with: $b with: (Character withValue: 0))
```

returns false in 3.1.x and returns true in 3.2.x

Bug 44199 - ClassOrganizer instances persisted through upgrade to 3.2 are missing symbol list

Product: GemStone/S 64 Bit

Versions: 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2

Platform All Platforms

Fixed In: 3.3

If you persist and re-use instances of ClassOrganizer in a pre-3.2 repository, and then upgrade the repository to 3.2, these instances will encounter an error when you attempt to use them.

In versions prior to 3.2, the instance variable #user was unused, and we retrieved the symbol list manually each time. Starting with v3.2, the #user instvar is initialized to a copy of "System myUserProfile symbolList" and that is used for the symbol list when needed. But for instances of ClassOrganizer created before v3.2 and persisted over upgrade, this instance variable will be nil, causing the error.

Workaround:

Manually initialize the #user instvar as follows:

```
myClassOrganizer _symbolList: System myUserProfile symbolList
```

Bug 44157 - Default IcuCollator must remain the same for reliable string collation

Product: GemStone/S 64 Bit

Versions: 3.6.3, 3.6.2, 3.6.1, 3.6, 3.5.7, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.x

Impact: Informational

Collation of Unicode strings (Unicode7, Unicode16, and Unicode32), depends on an instance of IcuCollator. Unless an IcuCollator is specifically supplied for the comparison, the default IcuCollator for the session is used. The default IcuCollator for the session is set during login based on the current OS locale.

This means that sessions that log in to the same repository from different locales may see differently collated Unicode strings. If using a SortedCollection rather than an IcuSortedCollection, these results may be persistent, and may result in lookup failures. Indexes on Unicode Strings (in v3.2 and later) persist the collator on index creation, and so avoid problems.

The default IcuCollator also be set explicitly by setting either the IcuCollator default: or the IcuLocale default:. This has the same risk as changing default IcuCollator based on OS locale.

Workaround:

Use IcuSortedCollection rather than SortedCollection. When sorting a particular collection according to customized rules, specify the collator for that sort, rather than changing the default collator.

If your gem will be in multiple locales with different collation rules, consider explicitly setting the default session collator for each gems on login.

Bug 44127 - Shared Cache size limited to 256GB on Linux

Product: GemStone/S 64 Bit

Versions: 3.6.3, 3.6.2, 3.6.1, 3.6, 3.5.7, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.x.X, 3.0.1, 3.0, 3.0.X, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.X, 2.3.1.6, 2.3.X, 2.2.5.4, 2.2.x, earlier versions

Platform Linux

Fixed In: 3.2, 2.4.7

Impact: Informational

Linux, by default, limits a single shared memory segment to 256GB. This effectively limits the size of the shared page cache.

Workaround:

Linux uses a memory page size of 4KB by default, but can be configured to use 2MB large memory pages. Large pages will allow a larger shared page cache. In v3.2 and later you can configure large pages on Linux by configuring the OS and using the configuration parameter SHR_PAGE_CACHE_LARGE_MEMORY_PAGE_POLICY. For details, see the Installation Guide for the specific version, such as Installation Guide for v3.4 for Linux.

Large pages can also be configured in v2.4.7, using a slightly different procedure. See the Installation Guide for v2.4.7 for Linux for details.

Bug 44087 - Unicode string comparision using equalsNoCase: may return false when = returns true

Product: GemStone/S 64 Bit

Versions: 3.6.3, 3.6.2, 3.6.1, 3.6, 3.5.7, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2

Unicode string comparison returns true for some cases where the actual characters in the string differ, such as when non-printing characters are present in one but the printing characters are the same. However, in these cases equalsNoCase: may return false.

For example, given the strings
Unicode16 with: $a with: $b
Unicode16 with: $a with: $b with: (Character withValue: 0).

These will return true for =, but false for equalsNoCase:.

Bug 44083 - Cannot execute = or other comparisons with mixed traditional/unicode strings

Product: GemStone/S 64 Bit

Versions: 3.6.3, 3.6.2, 3.6.1, 3.6, 3.5.7, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2

Impact: Informational

Unicode strings use the ICU libraries to perform equality operations, while traditional strings use traditional character-based equality. To avoid some conditions where the equality semantics are different, equality comparisons between a traditional and Unicode string using = have been disallowed in v3.2

This change impacts operations such as includes: that rely on = to identify objects, as well as ordering of objects in collections using <, <=, >, and >=. Identity-based comparisons using == or ~~ are unaffected.

Workaround:

You should avoid mixing traditional and Unicode strings in collections.

Using compareTo:collator: will allow any strings to be compared using Unicode rules.

If you enable Unicode comparison mode, it will redefine = for traditional strings and symbols to use the ICU libraries to perform equality and ordering operations; in which case compares will not longer be an issue.

Bug 44079 - GsQuery readstream with equality does not correctly handle cases where comparable objects are not equal

Product: GemStone/S 64 Bit

Versions: 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2

Fixed In: 3.3

Some kinds of objects can be ordered using comparison operators such as <=, etc., but are not equal; in other words, a <= b and a >= b does not mean a = b.

In particular, Strings and Symbols are comparable using <=, etc., and can be used as the indexed element within a single index. In this case, Strings and Symbols are mixed together in the comparison order within the internal btree, and index streaming will stream over a mixed collection.

Strings and Symbols with the same contents, however, are not equal using =. When performing a query using =, the indexed results should only include results of the appropriate class. This is handled correctly in query results returned as a collection, but indexed streams (using BtreeReadStream) could return both Strings and Symbols with the matching contents.

Bug 44060 - Stone may hang on AIX when swap space becomes very low

Product: GemStone/S 64 Bit

Versions: 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.3, 3.1.0.2, 3.1.0.1, 3.1.X, 3.0.1, 3.0, 3.0.X, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.X, 2.3.1.6, 2.3.X, 2.2.5.4, 2.2.X, 2.1, 2.0

Platform AIX only

Fixed In: 3.2, 2.4.8

Impact: Critical

When swap space on AIX becomes very low, AIX sends SIGDANGER. The Stone did not respond properly to this, and would hang.

Workaround:

In general, avoid low swap space conditions, but be particularly careful on AIX. If swap becomes low and the stone continues to hang after memory is recovered, you will need to restart the stone.

Bug 43983 - pageaudit does not accept -d option

Product: GemStone/S 64 Bit

Versions: 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.3, 3.1.0.2, 3.1.0.1, 3.1., 3.0.1, 3.0, 2.4.6, 2.4.5.1, 2.4.5

Fixed In: 3.2, 2.4.7

The pageaudit help includes the -d option to disable audit of data pages. This option was not accepted by the pageaudit executable.

Bug 43801 - Removing an entry from an index when collision buckets at full size may result in corruption

Product: GemStone/S 64 Bit

Versions: 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.3, 3.1.0.2, 3.1.0.1, 3.1, 3.0.1, 3.0, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.4.3, 2.4.4, 2.4.x, 2.3.x, 2.2.5.4, 2.2.5.x, 2.x, 1.x

Fixed In: 3.2, 2.4.7

GemStone identity indexes and most equality indexes use an internal structure, an instance of RcIndexDictionary, which contains instances of RcIndexBucketWithCache. When these collision buckets are fully packed (1380 entries), removing an element may corrupt the RcIndexBucketWithCache, and thus the index.

The index remove operations returns an error in this case, which in 2.x may be similar to:
"The object with object ID 20 is corrupt. Reason: 'FetchSmallInt_ bad value'"

Bug 43785 - Failed commit does not ensure stale object views are updated

Product: GemStone/S 64 Bit

Versions: 3.6.3, 3.6.2, 3.6.1, 3.6, 3.5.7, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.x, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.4.x, 2.4.x, 2.3.1, 2.3, 2.2.5.4, 2.2.x, earlier versions

A failed commit should retain all changes that were made to objects within the current transaction, and update any objects for which updates were committed in other sessions and were not changed in this session. However, there are cases when this second part of the behavior is not reliable.

If an object has been read into memory, then it is not marked invalid by the failed commit, so the old value remains visible as long as the object remains in memory. If the object happens to be removed by the in-memory GC, then when it is read in again from SPC/disk it will have the new value.

Workaround:

To get the latest values of objects after a failed commit, perform a selective abort.

Bug 43772 - MinusQuietNaN and MinusSignalingNaN are printed incorrectly

Product: GemStone/S 64 Bit

Versions: 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.3, 3.1.0.2, 3.1.0.1, 3.1, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.4.3, 2.4.4, 2.4.3, 2.4.2.1, 2.4.2, 2.4.1, 2.4, 2.4, 2.3.1.6, 2.3.1, 2.3, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.x, 2.x, 1.x

Platform All

Fixed In: 3.2

When inspecting or printing MinusQuietNaN and MinusSignalingNaN they are displayed as the Plus... counterpart, although the bits of the values are correct.

Bug 43764 - Multiple set-valued/enumerated predicates disallowed with &

Product: GemStone/S 64 Bit

Versions: 3.6.3, 3.6.2, 3.6.1, 3.6, 3.5.7, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2

Impact: Informational

Due to the design of the query code, multiple predicates, conjoined with &, that query on the same collection path (either set-valued or enumerated), would produce incorrect results. Queries structured in this way have been disallowed, as a result.

Note that queries that can be restructured by optimization to remove the &, or reduce the predicates to one for the collection path, will not have this problem and are allowed.

Bug 43758 - Interrupting reclaimAll/objectAudit can leave CR backlog setting / GC config options incorrect

Product: GemStone/S 64 Bit

Versions: 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.3, 3.1.0.2, 3.1.0.1, 3.1, 3.0.1, 3.0, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.3.1.6, 2.2.5.4

Platform All Platforms

Fixed In: 3.2

Repository>>reclaimAll reconfigures the stone configuration option StnSignalAbortCrBacklog to 3 and restores it to its original value when finished. But if reclaimAll is interrupted (the session is killed or an unexpected signal/error occurs and it is not allowed to complete) StnSignalAbortCrBacklog will be left at 3, causing major changes in the frequency of sigAbort/lostOT signals on the system.

In addition, reclaimAll also sets the following GcUser configuration options:

reclaimMinPages = 1

reclaimSleepTime = 1

sleepTimeBetweenReclaimMs = 0

These *must* be reset by executing Repository>>postReclaimAll: with the arguments returned by the reclaimAll:. If reclaimAll is interrupted and postReclaimAll: is not executed, these GC configuration options will be left at these settings. This may cause changes in the behavior of reclaim activity on the system.

Note that Repository>>objectAudit / Repository>>auditWithLimit: internally calls reclaimAll, and will therefore be subject to this bug as well (fixed in 3.0 during a redesign of objectAudit).

Workaround:

Manually reset these values:

As a user with SystemControl and GarbageCollection privileges:

```
run

System stoneConfigurationAt: #StnSignalAbortCrBacklog put: <original-value>.

SystemRepository _setGcConfigAt: #reclaimMinPages put: <original-value>
.

SystemRepository _setGcConfigAt: #reclaimSleepTime put: <original-value>.

SystemRepository _setGcConfigAt: #sleepTimeBetweenReclaimMs put: <original-value>.

%
```

Bug 43703 - Indexes on temporary collections with persistent elements cause problems on commit/abort

Product: GemStone/S 64 Bit

Versions: 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.3, 3.1.0.2, 3.1.0.1, 3.1, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.4.3, 2.4.4, 2.4.3, 2.4.2, 2.4.1, 2.4, 2.3.1.6, 2.3.x, 2.2.x, 2.1.x, 2.0.x

Fixed In: 3.2

Creating indexes requires that internal structures be created that reference both the collection and the elements in the collection. If the collection is temporary, this creates problems on commit or abort, since there is no reasonable correct state for the indexing structures. The operation may succeed, but subsequent audits may fail or other errors occur.

In v3.2, indexing structures are removed from temporary collections on abort.

Workaround:

Do not create indexes on temporary collections that reference persistent objects.

Bug 43656 - killing gem/topaz while doing index removal can crash reclaim GcGems

Product: GemStone/S 64 Bit

Versions: 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.3, 3.1.0.2, 3.1.0.1, 3.1, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.4.3, 2.4.4.1, 2.4.4, 2.4.x

Platform All Platforms

Killing a gem/topaz session while it is doing an index removal operation includes a possible risk of a reclaim GcGem crashing with a page cache fault.

Index removal operations include:
#removeAllIndexes
#removeEqualityIndexOn:
#removeIdentityIndexOn:

The crashed reclaim GcGem logs will contain entries like:

```
*************** Page Cache Fault ***************
  occurred at /some-file-directory-structure/pagecache.hc: NN
  message is: Page header fault on page NN, expected kind 7, found kind NN
```

(Where NN indicates numbers)

Bug 43655 - Use of SIGUSR1 can hang gem/topaz process

Product: GemStone/S 64 Bit

Versions: 3.6.3, 3.6.2, 3.6.1, 3.6, 3.5.7, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.x, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.4.3, 2.4.4.2, 2.4.4.1, 2.4.4, 2.4.x, 2.4, 2.3.1.6, 2.3.x

Platform All Platforms

When sending a SIGUSR1 signal to a gem or linked topaz process to generate a stack trace, there is a rare possibility of hanging the process, due to a deadlock between malloc / free operations during the handling of the signal.

The gem/topaz process can only be terminated using "kill -9". Attempts to use stopSession: will only cause the OOB thread on the process to also deadlock on a malloc operation.

Workaround:

No workaround. Use "kill -9" to terminate the process.

Bug 43585 - Positive exponent literals interpreted differently between GemStone and VisualWorks

Product: GemStone/S 64 Bit

Versions: 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.3, 3.1.0.2, 3.1.0.1, 3.1, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.4.3, 2.4.4, 2.4.3, 2.4.2.1, 2.4.2, 2.4.1, 2.4, 2.4, 2.3.1.6, 2.3.1, 2.3, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.x, 2.x, 1.x

Fixed In: 3.2

Impact: Informational

GemStone historically supports + in exponent notation, e.g. 1E+3. This
does not follow the ANSI standard nor the VW behavior, which interprets

that + as an addition operator.

Starting in v3.2, the + operator is disallowed when creating literal Doubles
in exponent notation; to avoid the risk of different results,

For example:

- In GemStone versions before 3.2: 1E+3 = 1E3 = 1000
- In VisualWorks: 1E+3 = 1E + 3 = 4

In GemStone 3.2 and later: 1E+3 creates a compiler error.
- to get results similar to VisualWorks, use 1E0+3.
- to get results similar to older versions of GemStone, use 1E3

Bug 43541 - Pragmas not currently functional in base product without Seaside

Product: GemStone/S 64 Bit

Versions: 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.3, 3.1.0.2, 3.1.0.1, 3.1, 3.0.1, 3.0

Platform All Platforms

Fixed In: 3.4

While pragmas are compiled properly, they require additional support provided
by the Seaside framework to be fully functional. Attempts to retrieve
pragmas using only the base product will always return nil.

Workaround:

The following code provides basic support for pragmas. Read the comments
for details. File-in as SystemUser and commit.

```
! Special code for compiling and accessing Pragmas
! until feature request 43541 is implemented.
!
! Note that this establishes an IdentityDictionary in Globals
! for holding the pragmas.  The eventual implementation of pragmas
! in the product is likely to use a different mechanism requiring
! that all code with pragmas be recompiled.
!

! Install the PragmaDictionary.
!
! This dictionary contains classes as keys and IdentityDictionaries
! as values.  The secondary IdentityDictionaries contain method selector
! symbols as keys and the associated pragmas as values.
!
run
Globals at: #PragmaDictionary put: IdentityDictionary new.
%


category: 'Updating the Method Dictionary'
method: Behavior
compileMethodWithPragmas: sourceString
category: categoryString
```

```
"Compiles sourceString and records any pragmas in PragmaDictionary."

  | pragmas meth pclass |
  pragmas := Array new.
  meth :=
     self compileMethod: sourceString
          dictionaries: System myUserProfile symbolList
          category: categoryString asSymbol
          intoMethodDict: nil
          intoCategories: nil
          intoPragmas: pragmas
          environmentId: 0.
  (pragmas size) > 0 ifTrue: [
    pclass := PragmaDictionary at: self ifAbsent: [ nil ].
    pclass ifNil: [
      pclass := IdentityDictionary new.
      PragmaDictionary at: self put: pclass ].
    pclass at: meth selector put: pragmas ].
  ^ meth
%

category: 'Pragmas'
method: GsNMethod
myPragmas

  "Returns the pragmas associated with this method."

  | pclass |
  pclass := PragmaDictionary at: self inClass ifAbsent: [ ^nil ].
  ^pclass at: self selector ifAbsent: [ nil ].
%

category: 'Pragmas'
method: GsMethod
myPragmas

  "Returns the pragmas associated with this method."

  | pclass |
  pclass := PragmaDictionary at: self inClass ifAbsent: [ ^nil ].
  ^pclass at: self selector ifAbsent: [ nil ].
%
```

Bug 43521 - Upgrade resets #sleepTimeBetweenReclaimMs to 0

Product: GemStone/S 64 Bit

Versions: 3.1.0.5, 3.1.0.4, 3.1.0.3, 3.1.0.2, 3.1.0.1, 3.1, 3.0.1, 3.0, 2.4.5.1, 2.4.5, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.4.3, 2.4.4.2, 2.4.4.1, 2.4.4, 2.4.x, 2.3.1.6, 2.3.x, 2.2.5.4, 2.2.5.2, 2.2.4, 2.2.x

Fixed In: 3.2, 3.1.0.6, 2.4.6

The upgrade process updates GcUser's configuration parameters. Any configured value for #sleepTimeBetweenReclaimMs was not preserved by upgrade.

On upgrade from version 2.2.4 or earlier, it would be set based on the previous value of #sleepTimeBetweenReclaim; upgrades from later versions would always result in a setting of 0.

Workaround:

Reset the configuration parameter manually after upgrade.

Bug 43515 - Corruption problem on large IdentityDictionaries

Product: GemStone/S 64 Bit

Versions: 3.1.0.5, 3.1.0.4, 3.1.0.3, 3.1.0.2, 3.1.0.1, 3.1, 3.0.1, 3.0, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.3.1.6, 2.2.5.4

Platform All Platforms

Fixed In: 3.2, 3.1.0.6, 2.4.6

Impact: Critical

When working with IdentityDictionaries with over 20K entries, there is a risk that the value assigned to a key will be the association containing the key and value rather than the expected value. For example:

After performing:

anIdentityDict at: #key put: #value.
^ anIdentityDict at: #key

Rather than returning #value as expected, the code could return the Association #key -> #value.

Workaround:

Login as SystemUser and file-in / commit the code below.

```
! Patch for bug 43515 on GS/64 2.X - 3.1.0.5
!
category: 'Private'
method: KeyValueDictionary
_rebuildAt: aKey put: aValue

  self _rebuild ; at: aKey put: aValue
%

category: 'Private'
method: IdentityCollisionBucket
at: aKey put: aValue keyValDict_coll: aKeyValDict

"Private.  Adds the key-value pair (aKey, aValue) to the collision bucket.
 Returns 1 if this at:put: added a new key, 0 if this at:put:
 replaced the value for an existing  key .  "

| marker |

aKey == nil ifTrue:[ ^ self _error: #rtErrNilKey ] .

marker := self binarySearchForInsertKey: aKey.
marker < 0 ifTrue:[ "replace existing key"
  self at: marker negated putValue: aValue.
  aKeyValDict _markDirty .
  ^ 0
  ].
(numElements >= 974) ifTrue:[
  aKeyValDict _rebuildAt: aKey put: aValue .
  ^ 0
  ].
self insertAt: marker key: aKey value: aValue.
^ 1
%

category: 'Private'
method: IdentityDictionary
_rebuildAt: aKey put: aValue

  ^ self _rebuild ; _at: aKey put: aValue
%
```

Bug 43454 - Character and String isUppercase, asUppercase, etc. unreliable for Characters over 256

Product: GemStone/S 64 Bit

Versions: 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.3, 3.1.0.2, 3.1.0.1, 3.1.X, 3.0.1, 3.0, 3.0.X, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.X, 2.3.1.6, 2.3.X, 2.2.5.4, 2.2.X, 2.x, 1.x

Fixed In: 3.2

Case testing and conversions, such as isUppercase/isLowercase and asUppercase/asLowercase, use the internal character data tables to determine the relationship between an upper and lower case versions of a Character. By default, these tables only include Characters with codePoings up to 256; for Characters with larger codePoints, no conversion is done.

If you have installed larger Character Data Tables, then these will be used to perform correct case testing and conversion. Installing Character Data Tables is deprecated in v3.1 and later.

Workaround:

in v3.1 and later, use Unicode strings rather than traditional String/DoubleByteString/QuadByteString. In earlier versions, install Character Data Tables will resolve the problem, but has application impacts.

Bug 43424 - Modifying method categories of a class without authorization silently fails

Product: GemStone/S 64 Bit

Versions: 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.3, 3.1.0.2, 3.1.0.1, 3.1.X, 3.0.1, 3.0, 3.0.X, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.X, 2.3.1.6, 2.3.X, 2.2.5.4, 2.2.X

Fixed In: 3.2

Attempting modify method categories on a class when to you do not have authorization to modify that class does not succeed, but does not report an error. This includes adding or removing a method category, and moving or renaming, and renameOrMerge.

Workaround:

Ensure you have authorization before modifying categories.

Bug 43418 - ProfMonitor results incomplete

Product: GemStone/S 64 Bit

Versions: 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.3, 3.1.0.2, 3.1.0.1, 3.1.X, 3.0.1, 3.0, 3.0.X, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.X, 2.3.1.6, 2.3.X, 2.2.5.4, 2.2.X, 2.1, 2.0

Platform All Platforms

Fixed In: 3.2

Due to a logic error introduced in GS/64 2.0, samples of the call stack taken by ProfMonitor only covered the top half of the stack. The resulting statistical summaries would therefore not account for method calls in the lower half of the stack.

Workaround:

In the method ProfMonitor>>gatherResults, replace the line:

```
1 to: depth -1 by: 2 do: [:m |
```

with:

```
1 to: (depth * 2) - 1 by: 2 do: [:m |
```

Bug 43414 - Bag>>= false positive when bags differ in count of objects

Product: GemStone/S 64 Bit

Versions: 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.3, 3.1.0.2, 3.1.0.1, 3.1, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.4.3, 2.4.4.2, 2.4.4.1, 2.4.4, 2.4.x, 2.3.1.6, 2.3.x, 2.2.5.4, 2.2.5.2, 2.2.x, 2.x, 1.x

Fixed In: 3.2

Bag >> = checks for the total count and that each object is present in the argument, but does not check that the numbers for each object in each Bag are the same. This leads to some bags reported as equal that are not.

Bug 43400 - GsFile class>>contentsAndTypesOfDirectory:onClient: reports files as directories

Product: GemStone/S 64 Bit

Versions: 3.6.3, 3.6.2, 3.6.1, 3.6, 3.5.7, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.x, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.x, 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.x, earlier versions

Platform All

The return value of GsFile class>>contentsAndTypesOfDirectory:onClient: is an array of values with the contents of the directory, and, for each, a boolean indicating if this item is a file. Files in the directory that have restricted permissions are reported false, which incorrectly implies that they are directories.

Bug 43391 - to:by: does not disallow stepvalue of 0

Product: GemStone/S 64 Bit

Versions: 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.3, 3.1.0.2, 3.1.0.1, 3.1, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.x, 2.4.x, 2.3.1.7, 2.3.1.6, 2.3.x, 2.2.5.4, 2.2.5.x, older versions

Fixed In: 3.2

The method Number >> to:by: creates an Interval. The step argument must be a positive or negative integer; 0 was incorrectly allowed. This created an Interval that was meaningless and not usable.

Bug 43388 - DateAndTime timeZoneName and timeZoneAbbreviation unsupported for most offsets

Product: GemStone/S 64 Bit

Versions: 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.3, 3.1.0.2, 3.1.0.1, 3.1, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.x, 2.4.x, 2.3.1.7, 2.3.1.6, 2.3.x, 2.2.5.4, 2.2.5.x, older versions

Fixed In: 3.2

ANSI specifies that DateAndTime should implement methods timeZoneName and timeZoneAbbreviation. These methods are only supported for DateAndTime instances in UTC (GMT), for which 'UTC' was returned.

An instance of DateAndTime has an offset, which specifies the difference from UCT. This is the only information available to provide the information for these methods. Since it is not possible to determine precisely which of the many named TimeZones correspond to a particular offset, in later versions these methods return a string containing the numerical offset.

Bug 43294 - String sorting methods using collation table are unreliable

Product: GemStone/S 64 Bit

Versions: 3.6.3, 3.6.2, 3.6.1, 3.6, 3.5.7, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.x, 3.0.1, 3.0

String implements a set of methods such as lessThanOrEqual:collatingTable:. These methods do not work reliably for instances with DoubleByteString or QuadByteString, and are likely to encounter errors, and not sort according to a customized table.

Methods that rely on these, such as asciiLessThan:, etc. also may have issues.

Workaround:

These methods have been deprecated in v.3.2. With Unicode string sorting, this collation table sort is not longer needed.

Bug 43226 - Non-string arguments to oldPassword:newPassword: crashes when disallowUsedPasswords is true

Product: GemStone/S 64 Bit

Versions: 3.1.0.4, 3.1.0.3, 3.1.0.2, 3.1.0.1, 3.1, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.x, 2.4, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2, older versions

Platform All

Fixed In: 3.1.0.5

In a repository configured with disallowUsedPasswords to true, attempting to change your password using oldPassword:newPassword: to a value that is not a string causes a SEGV.

Bug 43181 - deepCopy broken for Bag, Set, RcQueue

Product: GemStone/S 64 Bit

Versions: 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.3, 3.1.0.2, 3.1.0.1, 3.1, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.4.3, 2.4.4, 2.4.3, 2.4.2, 2.4.1, 2.4

Platform All

Fixed In: 3.2

deepCopy of instances of non-empty instances of Bag, Set, and RcQueue are empty.

Bug 43128 - Fileout requires CodeModification privilege

Product: GemStone/S 64 Bit

Versions: 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.3, 3.1.0.2, 3.1.0.1, 3.1, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.x, 2.4, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2, older versions

Platform All

Fixed In: 3.2

Attempts to file out code fails if logged in as a user without CodeModification privilege

Bug 42913 - Statmonitor started with -f will overwrite existing file with that name, including active file

Product: GemStone/S 64 Bit

Versions: 3.1.0.2, 3.1.0.1, 3.1, 3.0.1, 3.0, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.4.3, 2.4.4, 2.4.x, 2.3.x, 2.2.5.4, 2.2.5.x, 2.x, 1.x

Platform All

Fixed In: 3.1.0.3, 2.4.6

Starting statmonitor with the -f option allows you to specify a filename. Using this option will overwrite an existing file with that name. If two statmonitors are started specifying the same file name, both will write to this same filename, which corrupts the file.

Workaround:

Ensure that if you start up multiple instance of statmonitor using the -f option, that different file names are specified.

Bug 42760 - Dictionary copy creates copy with shared associations

Product: GemStone/S 64 Bit

Versions: 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.3, 3.1.0.2, 3.1.0.1, 3.1, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.4.3, 2.4.4, 2.4.3, 2.4.2.1, 2.4.2, 2.4.1, 2.4.x, 2.3.x, 2.2.6, 2.2.x

Fixed In: 3.2

When an object is copied, changes to that object should not have the side effect of modifying the original object, although copy does not make copies of the instance variable values themselves. For Dictionaries that are implemented using associations, including Dictionary and IdentityDictionary, this resulted in incorrect behavior.

Bug 42719 - RcQueue>>changeMaxSessionId: may cause problems

Product: GemStone/S 64 Bit

Versions: 3.1.0.5, 3.1.0.4, 3.1.0.3, 3.1.0.2, 3.1.0.1, 3.1, 3.0.1, 3.0, 2.4.5.1, 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.x, 2.4, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.x, older versions

Fixed In: 3.2, 2.4.6

The code implementing changeMaxSessionId: did not correctly initialize
the new sections of the RcQueue, which could result in later errors.

Bug 42702 - NotTranloggedGlobals not handled correctly in restore from logs

Product: GemStone/S 64 Bit

Versions: 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.3, 3.1.0.2, 3.1.0.1, 3.1, 3.0.1, 3.0

Fixed In: 3.3

Creation of non tranlogged objects are not recorded in the tranlogs. This can cause problems if these tranlogs are used to restore, and garbage collection has occurred, depending on the timing of the operations and log creation.

These problems can result in object audit failures.

Workaround:

Do not use NotTranloggedGlobals. NotTranloggedGlobals is disabled in v3.1.0.3 and later 3.1.x releases; while the bug was thought fixed in v3.2, more complex restore scenarios exposed variations of this problem.

Bug 42690 - CommitRecordBacklog created by terminated session

Product: GemStone/S 64 Bit

Versions: 3.1.0.2, 3.1.0.1, 3.1, 3.0.1, 3.0, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.4.3, 2.4.4, 2.4.3, 2.4.2.1, 2.4.2, 2.4.1, 2.4, 2.4, 2.3.1.6, 2.3.1, 2.3, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5

Fixed In: 3.1.0.3, 2.4.6, 2.2.6.2

It was possible for a session that is no longer logged in to hold the oldest commit record, causing a commit record backlog that could not be recovered normally (by terminating the session). The specific scenario involved a session killed by idle session timeout and exiting with a fatal error in such a way that the OOB socket was not properly terminated.

This bug is the same underlying problem as bug 41504.

Workaround:

For version 2.3 and earlier, you will need to shutdown and restart the stone to clear the condition. For version 2.3.1 and later, login as SystemUser and execute:

```
   System _cleanupZombieSession: <sessionId>
```

Warning: make sure that this session really is a zombie. Use of this command on a session that is still alive will cause that session's current transaction to fail.

Bug 42599 - PassiveObject operations fails with positionA methods installed

Product: GemStone/S 64 Bit

Versions: 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.4.3, 2.4.4, 2.4.x, 2.4

Platform All

Fixed In: 3.0

GemStone's PositionableStream classes are not ANSI-compliant; they use
a 1-based offset for position, while ANSI specifies an 0-based offset.
Version 2.4 and above include an option to installed ANSI-compliant position
methods (positionA, positionA:) on PositionableStream classes.

With these position methods installed, operations using PassiveObjects
will fail, including silently returning empty results.

Workaround:

Do not use both ANSI-compliant stream position methods and PassiveObject.
If this is required, contact GemStone Techinical Support for method changes
required in PassiveObject.

While this bug is fixed in v3.0, see bug 42596 for a related issue.

Bug 42593 - Inconsistent behavior for sortAscending for Dictionaries

Product: GemStone/S 64 Bit

Versions: 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.3, 3.1.0.2, 3.1.0.1, 3.1, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.4.3, 2.4.4, 2.4.3, 2.4.2, 2.4.1, 2.4, 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.x, 2.2.5, 2.2.4, 2.2.3

Platform All

Fixed In: 3.2

The methods sortAscending, sortAscending:, sortDescending, sortWith: and
so on, are legacy GemStone protocol that historically operated on the elements
of Dictionaries as the associations, and return the associations sorted
by each association's key. ANSI considers the "elements" of a Dictionary
to be the values, not associations, so results would be the values, sorted
by value.

The behavior of these methods was updated to adhere closer to the ANSI
concept of Dictionary. This is a difference both in the type of object
in the returned collection, and the order of the results.

However, the methods sortAscending:, sortDescending, and sortWith: were
not updated and still return Associations sorted by key.

Starting with v3.2, all these methods follow the ANSI behavior.

Bug 42518 - GciInterface traversal code is broken

Product: GemStone/S 64 Bit

Versions: 3.1.0.1, 3.1, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.4.3, 2.4.x, 2.4, 2.3.1.6, 2.3.1, 2.3, 2.2.x, 2.1.x, 2.0.x

Platform All

Fixed In: 3.1.0.2

The GciInterface class defines a number of methods that support object traversal. While the underlying object traversal code, which is used by GBS via the GCI C-level inteface, is reliable, there are a number of problems with the interface from the image. This includes protection on TraversalBuffer creation, failures to handle double byte strings, and primitive errors from TraversalBuffer operations.

GciInterface methods, other than the traversal buffer related operations, in general are reliable.

Workaround:

Create user actions and use the C-level GCI interface traversal functions.

Bug 42509 - remoteExecuteTrav: cannot be used with DoubleByteStrings

Product: GemStone/S 64 Bit

Versions: 3.6.3, 3.6.2, 3.6.1, 3.6, 3.5.7, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.2, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.5, 2.4.4.x, 2.4.x, 2.3.1.6, 2.3.x, 2.2.5.4, 2.2.x, earlier versions

The method GciInterface >> remoteExecuteTrav: relies on GCI interface methods that are limited to 8 bit characters and that are valid C strings. Strings passed to this method, and results returned, must not contain Characters with codePoints outside the range 1...254.

Bug 42470 - Reclaim GC gem aborting while reclaiming dead can corrupt repository

Product: GemStone/S 64 Bit

Versions: 3.1.0.1, 3.1, 3.0.1, 3.0, 2.4.5.3, 2.4.5.2, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.4.3, 2.4.4.2, 2.4.4.1, 2.4.4, 2.4.3, 2.4.2.1, 2.4.2, 2.4.1, 2.4, 2.2.6, 2.2.5.4.2.2, 2.2.5.4.1, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.4, 2.2.3, 2.2.2, 2.2.1, 2.2

Platform All Platforms

Fixed In: 3.2, 3.1.0.2, 2.4.6, 2.2.6.1

Impact: Critical

There is a small risk that if a reclaim GC gem is reclaiming dead objects and does an abort in response to a sigAbort, internal tables tracking the dead objects will become corrupted. This can cause a wide range of various types of page and object level corruption, including page cache errors, object does not exist errors, and corrupted application data.

Workaround:

We *highly* recommend that customers upgrade to a version containing the fix ASAP to avoid this bug (3.2, 3.1.0.2, 2.4.6, 2.2.6.1, or later).

You can reduce the likelihood of sending a reclaim GC gem a sigAbort by avoiding the following:

1. Generating a CR backlog that exceeds the setting of the STN_CR_BACKLOG_THRESHOLD. This is difficult to do in practice and the main reason why we encourage you to upgrade ASAP. In the interim, you should develop a monitoring task that detects when the CR backlog threshold is exceeded and shuts down the reclaim GC gems. See the workaround code for an example.

2. Running any garbage collection operation that reports new dead objects, such as MFC, markGcCandidates, and Epoch GC. At the completion of these operations, a vote is taken on the possible dead, and sessions can be sent sigAborts to get their attention.

To safely run any garbage collection operations, you should either shutdown the reclaim GC gems or reconfigure them to not process dead objects. They should remain in this state until the vote is completed and statmon stone stat GcVoteState returns to zero.

You can shut them down by doing:

1. Run: System stopAllReclaimGcSessions.

Restore by doing:

1. Run: System startAllReclaimGcSessions.

If you would like to keep the reclaim GC gems running to perform shadowed page reclaim but not process dead objects, do the following:

1. Run: System stopAllReclaimGcSessions.
2. As GcUser: set UserGlobals at: #reclaimDeadEnabled to false.
3. Run: System startAllReclaimGcSessions.

Restore by doing:

1. Run: System stopAllReclaimGcSessions.
2. As GcUser: set UserGlobals at: #reclaimDeadEnabled to true.
3. Run: System startAllReclaimGcSessions.

```
! BUG 42470 Support Code
!
! Example code for stopping Reclaim GC gems when CR backlog threshold
! is exceeded, and restarting them when it falls back below.
!
! Useful as a temporary workaround for part of bug 42470.
! Note that you also need to disable Epoch GC and disable
! reclaim GC gems when performing MFC (see bugnote for details).
!
! Note that depending on settings of sleepTime, shutdownOffset,
! and restartOffset and the behavior of your system,  you *still* could
! have reclaim GC gems running when the CR backlog threshold is crossed
! and risk hitting bug 42470.
!
run
| crb crbThreshold sleepTime reclaimRunning shutdownOffset restartOffset |

"Configure as appropriate for your site"
sleepTime := 1.  "in seconds"
shutdownOffset := 10. "to allow for lag time before reclaim gems can shutdown"
restartOffset :=  20.
reclaimRunning := true.

"Make us transactionless so we respond to sigAborts (if needed)"
System transactionMode: #transactionless.

"Get the stone configuration STN_CR_BACKLOG_THRESHOLD"
crbThreshold := System configurationAt: #StnCrBacklogThreshold.

[ true ] whileTrue: [
    System abortTransaction.
    crb := (System cacheStatistics: 1) at: 160.
    (reclaimRunning and: [crb > (crbThreshold - shutdownOffset)])
    ifTrue: [
        GsFile stdout log:
            (DateTime now asString) , ': ' ,
            'CR backlog exceeded: shutting down reclaim gems'.
        System stopAllReclaimGcSessions.
        reclaimRunning := false ].
    ((reclaimRunning not) and: [crb < (crbThreshold - restartOffset)])
    ifTrue: [
        GsFile stdout log:
            (DateTime now asString) , ': ' ,
            'CR backlog cleared: restarting reclaim gems'.
        System startAllReclaimGcSessions.
        reclaimRunning := true ].
    System sleep: sleepTime ].
%
```

Bug 42463 - Memory leak in Stone with use of notifiers

Product: GemStone/S 64 Bit

Versions: 3.1.0.1, 3.1, 3.0.1, 3.0, 2.4.5.1, 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.x, 2.4, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.x, 2.2

Fixed In: 3.1.0.2, 2.4.6

Gem signaling allocates memory on the C heap of the Stone's process. When notifiers are used, in some cases this memory was not freed, which resulted in growth of the Stone's memory footprint. This ultimately could cause the Stone to run out of memory and crash with a malloc error.

Bug 42461 - Reclaiming dead while backup is running can corrupt repository

Product: GemStone/S 64 Bit

Versions: 3.1.0.1, 3.1, 3.0.1, 3.0, 2.4.5.3, 2.4.5.2, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.4.3, 2.4.4.2, 2.4.4.1, 2.4.4, 2.4.3, 2.4.2.1, 2.4.2, 2.4.1, 2.4, 2.2.6, 2.2.5.4.2.2, 2.2.5.4.1, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.4, 2.2.3, 2.2.2, 2.2.1, 2.2

Platform All Platforms

Fixed In: 3.2, 3.1.0.2, 2.4.6, 2.2.6.1

Impact: Critical

There is a small risk that if reclaim GC Gems are reclaiming dead objects while a full backup is running, internal tables tracking the dead objects will become corrupted. This can cause a wide range of various types of page and object level corruption, including page cache errors, object does not exist errors, and corrupted application data.

Workaround:

If backups do not take a long time to run and you don't mind disabling shadowed page reclaim as well as dead reclaim:

1. Run: System stopAllReclaimGcSessions.
2. Perform the backup.
3. Run: System startAllReclaimGcSessions.

If backups take a long time to run and you would like to keep shadowed page reclaim running:

1. Run: System stopAllReclaimGcSessions.

2. As GcUser: set UserGlobals at: #reclaimDeadEnabled to false.
3. Run: System startAllReclaimGcSessions.
4. Perform the backup.
5. Run: System stopAllReclaimGcSessions.
6. As GcUser: set UserGlobals at: #reclaimDeadEnabled to true.
7. Run: System startAllReclaimGcSessions.

Bug 42441 - Checkpoint error on startup after clean shutdown

Product: GemStone/S 64 Bit

Versions: 3.1.0.1, 3.1, 2.4.5.1

Platform All, using raw partitions

Fixed In: 3.1.0.2, 2.4.6

When tranlogs are on raw partitions, if a clean shutdown is done shortly
after starting a new transaction log (so there are very few records in
the final tranlog before shutdown), the subsequent startstone will incorrectly
report a checkpoint mismatch.

This error message is incorrect; there is actually no problem with the
extents, all data was written correctly. The startup checks include a
check for filesize, as well as the block id checks; both checks use the
same error message. The filesize check is failing due to filesize not
being recorded correctly for very small raw partition tranlogs.

Workaround:

(1) Verify that the stone was cleanly shutdown by checking the stone log.
A clean shutdown includes via stopstone, System shutDown, or kill -TERM.
The stone log will show messages of either of these forms:
   SHUTDOWN command was received from user ...
   ... < other messages >
   Now stopping GemStone. or
   SHUTDOWN command was generated by SIGTERM handler.
   ... < other messages >
   Now stopping GemStone.

(2) Archive and remove the current tranlogs

(3) Restart stone with -N -R

(2) Execute SystemRepository commitRestore. This will warn of restoring
without previously execution restoreFromCurrentLogs, but you can ignore
this; all of the data was checkpointed as part of the shutdown.

Bug 42317 - 2.x fileout may version classes in 3.x filein

Product: GemStone/S 64 Bit

Versions: 3.6.3, 3.6.2, 3.6.1, 3.6, 3.5.7, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.x, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.4.3, 2.4.4, 2.2.6, 2.x

Platform All

Fileout of the contents of a SymbolDictionary, using GBS or directly by invoking
  ClassOrganizer >> fileOutClassesAndMethodsInDictionary:on:
creates a fileout that is poorly structured. When filed in again, this will cause the classes to be versioned, circumventing recent changes that avoid unneccessarily versioning classes.

This is a particular problem when performing file-in as part of a GemStone/S 64 Bit upgrade to 3.0 or later. If the filein versions the classes, the unconverted classes and methods will remain in the image.

Fileouts created in v3.2 or later are correct, so this bug is fixed in this version; but upgrade from versions that include this bug to v3.2 or later, will still be impacted.

Fileouts of individual classes are not affected.

Workaround:

You can recompile methods directly after conversion, rather than filing in code. Filing out the 2.x code as classes avoids the problem.

Bug 42292 - topaz problem with paste of long lines with tabs

Product: GemStone/S 64 Bit

Versions: 3.6.3, 3.6.2, 3.6.1, 3.6, 3.5.7, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.x, 3.0.1, 3.0

When pasting text from an editor into topaz, if the text being pasted does not have regular line breaks and include tab characters, the display may include multiple copies of lines in the text. Only the echo to terminal is affected; code executes correctly and output push results are correct.

Workaround:

Topaz executing using input is not affected.

Turning off the lineditor avoids this problem. To turn off, execute:
  OMIT LINEEDITOR

Bug 42281 - Infinite loop comparing SymbolDictionaries with same name and contents

Product: GemStone/S 64 Bit

Versions: 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.3, 3.1.0.2, 3.1.0.1, 3.1, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.4.3, 2.4.4, 2.4.3, 2.4.2, 2.4.1, 2.4, 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.x, 2.2, 2.1.4, 2.1.x, 2.1, 2.0.5, 2.0.4, 2.0.x, 2.0, 1.x

Fixed In: 3.2

If two instance of SymbolDictionary are compared, which have the same contents and the same name, the comparison will get stuck in an infinite loop.

Workaround:

Avoid comparing exact duplicate SymbolDictionaries.

Bug 42185 - GsFile write of large character silently fails, causing later issues

Product: GemStone/S 64 Bit

Versions: 3.0.1, 3.0, 2.4.5.1, 2.4.5, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.4.3, 2.4.4.2, 2.4.4.1, 2.4.4, 2.4.3, 2.4.2, 2.4.1, 2.4, 2.3.1.7, 2.3.1.6, 2.3.1.5, 2.3.1.4, 2.3.1.3, 2.3.1.2, 2.3.1.1, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.4, 2.2.x, 2.1.x, 2.0.x, 1.x

Fixed In: 3.1, 2.4.6

GsFile is limited to writing single-byte characters, and does not correctly handle writes of characters outside this range nor DoubleByteStrings. Attempting to write a large character, however, did not always return an error. In this case, the error status was set but not detected, and a later unrelated operations could return the character out of range error.

Since GsFile is implemented a user actions and used the same error handling, the later unrelated operations that failed could be related to a user action.

Bug 42153 - Linux process stats report values 16 times smaller than actual

Product: GemStone/S 64 Bit

Versions: 3.0.1, 3.0, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.4.3, 2.4.4.2, 2.4.4.1, 2.4.4, 2.4.3, 2.4.2, 2.4.1, 2.4

Platform Linux

Fixed In: 3.1, 2.4.5.1

The process stats values reported on linux are 16x smaller than the actual values.

Bug 42147 - Failure in pregrow of dynamically added extent causes extent corruption

Product: GemStone/S 64 Bit

Versions: 3.0.1, 3.0, 2.4.5, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.4.3, 2.4.4, 2.4.3, 2.4.2, 2.4.1, 2.4, 2.3.1, 2.3, 2.2.5.4, 2.2.x, 2.x, 1.x

Platform All

Fixed In: 3.1, 2.4.5.1, 2.4.4.8, 2.2.6

Impact: Critical

When an extent is added dynamically using methods such as createExtent:withMaxSize:, the new extent is pregrown if DBF_PRE_GROW is true.

If the operation fails, for example if the specified size is larger than available disk, the system does not clean up properly. This can result in the wrong information written to the root page and the inability to restart due to corrupt root page.

Bug 42141 - Memory leak in tranlog restore

Product: GemStone/S 64 Bit

Versions: 3.0.1, 3.0, 2.4.5, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.4.3, 2.4.4.2, 2.4.4.1, 2.4.4, 2.4.3, 2.4.2, 2.4.1, 2.4

Platform All

Fixed In: 3.1, 2.4.5.1, 2.4.4.8

In restoring a tranlog, there is a memory leak. While this can be a 3M leak per tranlog restore operations, depending on other factors it can be more than 500M per tranlog.

This is primarily an issue for warm standbys, which continuously restore tranlog one by one. Eventually the stone will run out of memory and shut down.

Workaround:

Restarting the stone frees up the memory. Restarting the stone does not affect restore state, so it can be done between tranlog restores in a warm standby.

Bug 42125 - Commit record backlog during tranlog restore

Product: GemStone/S 64 Bit

Versions: 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.4.3, 2.4.4.2, 2.4.4.1, 2.4.4, 2.4.3, 2.4.2, 2.4.1, 2.4

Platform All

Fixed In: 3.1, 2.4.5.1, 2.4.4.8

When restoring transaction logs containing a large amount of reclaim, it is possible for a commit record backlog to develop as the recovery thread waits for reclaim to complete. Under worst case conditions, this can use up all available free space and cause the restore to fail.

Bug 42114 - Tranlog replay may fail with unhandled recordKind

Product: GemStone/S 64 Bit

Versions: 3.0.1, 3.0, 2.4.5, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.4.3, 2.4.4, 2.4.3, 2.4.2, 2.4.1, 2.4

Platform All

Fixed In: 3.1, 2.4.5.1, 2.4.4.8

Impact: Critical

If a repository has a non-zero STN_GEM_TIMEOUT, it is possible for a timing condition when a session is terminated to cause a bad transaction log record to be written to the tranlog. The bad transaction log record has an invalid record type number, which exhibits as a fatal error on tranlog replay:

```
  readRecord: unhandled recordKind NNN
```

Valid record types are 1 through about 53 (depending on the version); this bug has been observed with invalid record types of 103 and 7968.

In addition to gem termination due to non-zero STN_GEM_TIMEOUT, a failure to update the config file after dynamically adding an extent could also result in this problem.

Repository recovery after crash, restoring tranlogs after restoring a backup, and warm standbys are all affected by this bug. In systems that are subject to this bug, there is a chance that recovery after crash or restore from backup may fail.

The actual risk of this bug causing bad tranlog records is unknown. This bug is related to code changes introduced in v2.4, and has only been encountered once. However, since the tranlogs appear normal and would only report problems if replayed, it is possible there are silent occurances of this bug.

Workaround:

Running with STN_GEM_TIMEOUT set to 0 avoids the conditions that cause this bug.

Reducing the changes that idle sessions are terminated would reduce the risk.

We recommend customers upgrade to a version in which this bug is fixed as soon as this is available.

If you do encounter this bug, contact GemStone Technical Support. Engineering may be able to assist by editing the tranlog so that only one tranlog record or transaction might be lost.

Bug 42089 - Problems with AIX on POWER7

Product: GemStone/S 64 Bit

Versions: 3.0.1, 3.0, 2.4.5, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.4.3, 2.4.4, 2.4.3, 2.4.2, 2.4.1, 2.4, 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5.x, 2.x

Platform AIX on POWER7

Fixed In: 3.1, 2.4.5.1, 2.2.6

Impact: Informational

The POWER7 chip provides more aggressive prefetch of data and instructions. This requires additional memory barriers on atomic updates within GemStone, to avoid the risk of coherency and audit problems.

These problems are exposed under heavy load on GemStone. It is unlikely that this bug would cause problems in non-production use.

POWER7 is not a certified platform with currently released GemStone versions, but documentation for recent releases does not always clearly indicate problems on POWER7.

Workaround:

Do not upgrade production system to a machine running POWER7 when running a version of GemStone/S 64 Bit that is subject to this bug.

Bug 42073 - to:by:do: does not disallow stepvalue of 0

Product: GemStone/S 64 Bit

Versions: 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.x, 2.4.x, 2.3.1.7, 2.3.1.6, 2.3.x, 2.2.5.4, 2.2.5.x, older versions

Fixed In: 3.1

The method Number >> to:by:do: iterates a block as specified by the receiver, argument, and the step value. A step value of 0 is not disallowed, so if the step value is 0, the block is evaluated infinitely with the initial argument.

Bug 41992 - SmallDouble literals and conversion from String slight rounding inaccuracy on Linux

Product: GemStone/S 64 Bit

Versions: 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.3, 3.1.0.2, 3.1.0.1, 3.1, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4x, 2.3.1.6, 2.3.1, 2.2.5.4, 2.2.x, earlier versions

Platform Linux only

Rounding of floating point values, per IEEE spec 754, rounds toward even.

The value exactly between 1.0 and its SmallDouble successor is precisely
h = 1.00000000000000011102230246251565404236316680908203125

This value rounds to 1.0.

A value slightly larger than h should round to 1.0000000000000002.

However, given a value slightly larger than h, such as
s = 1.000000000000000111023

On Linux versions with glibc earlier than 2.17, the SmallDouble created from s is incorrectly 1.0, rather than the correct result which is 1.0000000000000002.

The GemStone internal code that creates SmallDoubles from literals or String conversion uses the OS function strtod(). Linux kernels have a bug in this code: https://sourceware.org/bugzilla/show_bug.cgi?id=3479.

Workaround:

This Linux bug is fixed in glibc v2.17.

Bug 41950 - Corruption using KeyValueDictionary>>removeKey:otherwise:

Product: GemStone/S 64 Bit

Versions: 3.0.1, 3.0, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.4.3, 2.4.4, 2.4.3, 2.4.2, 2.4.1, 2.4, 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.4, 2.2.3, 2.2.2

Platform All platforms

Fixed In: 3.1, 2.4.5

Impact: Critical

Using KeyValueDictionary>>removeKey:otherwise: with a key that does not occur within the dictionary can sometimes result in decrementing the numElements and numCollisions fields in the dictionary, even though nothing was actually removed. This won't be detectable until additional entries are removed from the dictionary, causing the numElements field to become negative and triggering an #objErrCorruptObj error (2261).

This bug effects KeyValueDictionary and all subclasses that don't override the #removeKey:otherwise: method, including:

IntegerKeyValueDictionary
StringKeyValueDictionary
CanonicalStringDictionary
IdentityKeyValueDictionary

IdentityDictionary
SymbolDictionary
LanguageDictionary
SymbolKeyValueDictionary

Workaround:

Use other remove methods, or check that the key actually does exist in the dictionary before using #removeKey:otherwise:.

If you have dictionaries suffering from this problem, contact GemStone Technical Support for code that will repair the dictionaries.

Bug 41904 - commitRestore may reset tranlog sequence to 0

Product: GemStone/S 64 Bit

Versions: 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.4.3, 2.4.4, 2.4.3, 2.4.2.1, 2.4.2, 2.4.1, 2.4, 2.4

Platform All

Fixed In: 3.0, 2.4.5, 2.4.4.8

When commitRestore is performed immediately after startstone -N, it may reset the stone's tranlog Id to 0. While this should not cause a problem with the new sequence, it may cause a conflict with existing tranlogs.

Bug 41882 - Recovery problems with free space

Product: GemStone/S 64 Bit

Versions: 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.3, 3.1.0.2, 3.1.0.1, 3.1, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.4.3, 2.4.4, 2.4.3, 2.4.2, 2.4.1, 2.4, 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.x, 2.2, 2.1.x, 2.0.x, 1.2.4, 1.2.x, 1.x

Platform All

Fixed In: n/a

Impact: Informational

Under some circumstances, on recovery after unexpected shutdown, page demands during recovery will be more than the space in the repository, and recovery will fail.

The most likely scenario for this problem is when there is a large amount of reclaim, and possibly a commit record backlog, and the system is not able to write a checkpoint prior to shudown. If there is a large amount of reclaim and other work done during the period of the commit record backlog, all these records must be replayed from the tranlogs during recovery. During the special conditions of recovery, used space cannot be freed for reuse.

Workaround:

Adding sufficient extent space will allow recovery to complete.

During tranlog restore, unlike during recovery, space can be reclaimed, so restoring from backup and tranlogs will succeed.

To avoid this problem, the best way is to avoid the commit record backlog and resulting shutdown in the first place. Be especially careful about commit record backlogs in the period following an MFC or other operation that will cause a large amount of reclaim.

The default for STN_FREE_SPACE_THRESHOLD in earlier version is 1MB, which is much too small; we suggest configuring this to be 1/1000 of the repository size, which is the default in later versions. Depending on the rate of page use, this larger value would be likely to allow the session holding the CRB to be terminated, avoiding a shutdown, and increases the chance that the checkpoint can complete.

However, note that if the repository is large, with a large amount of changes that need to be written out as part of the checkpoint, it may take significant time for a checkpoint to complete. See also bug #41809.

Bug 41809 - Slow checkpoints if much more AIO page servers than Free Frame page servers

Product: GemStone/S 64 Bit

Versions: 3.6.3, 3.6.2, 3.6.1, 3.6, 3.5.7, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.x, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.x, 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.x, earlier versions

Platform All

Fixed In: n/a

Impact: Informational

If you increase the number of AIO page servers, but leave the number of free frame page servers at 1, under some conditions of heavy load the checkpoints may take a very long time to complete, an hour or more.

The problem is that the page servers operate on separate regions within the cache, dividing the cache amoung the total number of page servers. If there is only one free frame page server, it operates on the whole cache, so it will return pages to one region at a time as it progresses through the cache, and these free pages may be used for new work. On checkpoint, when the AIO page servers write out all modified pages, most of these pages will be in one region, so the remaining page servers will sit idle waiting for one page server to complete writing.

Workaround:

Ensure that the SHR_NUM_FREE_FRAME_SERVERS is the same as STN_NUM_LOCAL_AIO_SERVERS.

In later versions this is the default.

Bug 41780 - Restore of compressed backups is very slow

Product: GemStone/S 64 Bit

Versions: 3.0, 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.4.3, 2.4.4, 2.4.3, 2.4.2, 2.4.1, 2.4, 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.x, 2.2, 2.1.4, 2.1.x, 2.1, 2.0.5, 2.0.4, 2.0.3, 2.0.x, 1.2.4, 1.2.x, 1.1.x, 1.x

Fixed In: 3.0.1, 2.4.5

The way compressed backups are read during restore is very inefficient, due to buffer sizes and seeks; this results in restore from a compressed backup taking an unreasonably long time.

Workaround:

gunzip the compressed backup prior to restoring.

Bug 41764 - findReferencePathToObject: not intended for use with Classes/Metaclasses

Product: GemStone/S 64 Bit

Versions: 3.6.3, 3.6.2, 3.6.1, 3.6, 3.5.7, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.x, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.x, 2.3.1.7, 2.3.x, 2.3, 2.2.5.4, 2.2.x, earlier versions

Platform all

Impact: Informational

The method Repository >> findReferencePathToObject: can be used to find references to objects in the repository, to determine why these instance are not garbage collected. Due to the design of the code, this method does not work to find references to instances of Class or MetaClass.

Workaround:

To find immediate references to instances of Class or MetaClass, use listReferences:. If this is not enough information, you can use findReferencePathToObject: to find a reference path each of the objects returned by listReferences:.

Bug 41763 - Versioning class may have left reference to original class

Product: GemStone/S 64 Bit

Versions: 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.4.3, 2.4.4.2, 2.4.4.1, 2.4.4, 2.4.3, 2.4.2, 2.4.1, 2.4, 2.3.1.7, 2.3.1.6, 2.3.1.5, 2.3.1.4, 2.3.1.3, 2.3.1.2, 2.3.1.1, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.4, 2.2.x, 2.1.x, 2.0.x, 1.x

Fixed In: 3.1, 2.4.5, 2.4.4.7

Creating a version of a class copies the contents of the #description field to the new class. If the description contains an instance of GsClassDocumentation (i.e. if the class comment is set, for example by using GBS browsers), this instance of GsClassDocumentation is referenced by both class versions, but it itself continues to reference the original version of the class.

If the original class was otherwise dereferenced, this reference would prevent garbage collection.

Bug 41748 - Topaz commands that accept file paths have limited flexibility in specification

Product: GemStone/S 64 Bit

Versions: 3.6.3, 3.6.2, 3.6.1, 3.6, 3.5.7, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.x.X, 3.0.1, 3.0, 3.0.X, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.X, 2.3.1.6, 2.3.X, 2.2.5.4, 2.2.x, 1.x, earlier versions

Platform All

topaz commands that take a file path, such as output push and fileout tofile:, do not provide full host filename expansion capability.

While environment variables that specified a directory within the path are expanded, embedded environment variables are not expanded. The tilde
~ also cannot be used in versions 3.2.x and earlier.

Workaround:

Use fully-specified filenames.

Bug 41550 - Native code may be disabled with large gem TOC on Darwin

Product: GemStone/S 64 Bit

Versions: 3.6.3, 3.6.2, 3.6.1, 3.6, 3.5.7, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.x, 3.0.1, 3.0

Platform Darwin/Macintosh

Impact: Informational

With a very large temporary object cache, the memory offset may exceed 32bits on Darwin. In this case, native code will be disabled and the session will run without native code (as in GS64 2.x). A message is written to stdout (either the gem log or the linked session output).

Workaround:

We do not recommend running large Gem caches on the Macintosh.

Bug 41504 - Zombie gem can hold commit record, cause CR backlog

Product: GemStone/S 64 Bit

Versions: 3.1.0.2, 3.1.0.1, 3.1, 3.0.1, 3.0, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.4.3, 2.4.4, 2.4.3, 2.4.2.1, 2.4.2, 2.4.1, 2.4, 2.4, 2.3.1.6, 2.3.1, 2.3, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5

Platform All platforms

Fixed In: 3.1.0.3, 2.4.6, 2.2.6.2

When a session has logged out or been terminated, the stone normally waits
until it receives confirmation that the associated process is gone before
clearing the session's reference to a commit record. Under rare conditions
it is possible for the stone to miss this confirmation, causing the
"zombie" gem to continue holding the commit record and causing a commit
record backlog to develop that cannot be cleared.

You can confirm you have a zombie session causing a commit record backlog
when "System sessionsReferencingOldestCr" returns an array containing
a session ID that does not appear in "System currentSessions". Using System>>descriptionOfSession:
on this ID returns an array of nils.

Workaround:

For version 2.3 and earlier, you will need to shutdown and restart the
stone to clear the condition. For version 2.3.1 and later, login as SystemUser
and execute:

```
  System _cleanupZombieSession: <sessionId>
```

Warning: make sure that this session really is a zombie. Use of this
command on a session that is still alive will cause that session's current
transaction to fail.

Bug 41501 - Cannot start stone, IPC identifier already exists

Product: GemStone/S 64 Bit

Versions: 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.x, 2.3.1.6, 2.3.x, 2.2.5.4, 2.2.x, 2.x, 1.x

Fixed In: 3.0

When you have multiple stones running on the same machine, particular if you are using multiple definitions for GEMSTONE_GLOBAL_DIR to define multiple separate lock file areas and gemstone-logical-spaces on the same machine, there is a risk that the same shared memory key will be computed by the OS. This results in an incorrect error that the IPC identifier already exists.

This occurs since the ftok() command is documented to have the risk of returning the same key for different files, since it only uses a set number of bits of the inode (plus device, plus project id).

Workaround:

Upgrading is recommended. As an immediate workaround, creating files to order to use inodes would allow a different lock file inode to be allocated to avoid the conflict.

Bug 41476 - Restore on platform with different byte order corrupts indexes

Product: GemStone/S 64 Bit

Versions: 2.4.4.4, 2.4.4.3, 2.4.4, 2.4.3, 2.4.2, 2.4.1, 2.4, 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.x, 2.2, 2.1.4, 2.1.3, 2.1.2, 2.1.1, 2.1

Fixed In: 2.4.4.5

Impact: Critical

If a backup of a repository containing indexed collections is restored on a platform with a different native byte order than the one it was created on, indexes in the restored repository are corrupted.

For example, a backup created on Solaris on SPARC and restored to a repository on Linux on x86, or vice versa, will have this problem.

auditIndexes will report missing DependencyList for all indexed elements. Updating the indexed data will result in additional corruption.

Workaround:

Remove all indexes and rebuild after the restore.

Bug 41469 - waitstone may report stone ready during restart after crash

Product: GemStone/S 64 Bit

Versions: 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.4.3, 2.4.4.2, 2.4.4.1, 2.4.4, 2.4.3, 2.4.2, 2.4.1, 2.4

Platform All

Fixed In: 3.0

Impact: Critical

When the stone is restarted following a crash, it performs recovery from tranlogs. During this period, waitstone may incorrectly report that the stone is ready for logins.

This is most dangerous in the presence of bug #41462, in which logins are allowed during recovery; any commits during the recovery process can permanently corrupt the repository.

Workaround:

Do not use waitstone as a test for login. Waiting until startstone returns is a safe test that the stone is ready for login. startstone will not return to the prompt until recovery is complete.

Bug 41449 - Turning on stale account aging may disable accounts

Product: GemStone/S 64 Bit

Versions: 3.6.3, 3.6.2, 3.6.1, 3.6, 3.5.7, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.x, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.x, 2.3.1.7, 2.3.x, 2.3, 2.2.5.4, 2.2.x, earlier versions

Platform All

The lastLoginTime of an account is only updated if the repository has stale account aging or password aging enabled. The update of the lastLoginTime on login requires a commit, which is not always desireable.

As a result, if a stale account age limit is set in a repository that did not previously have either check, the lastLoginTime of accounts that log in frequently may be still set to a date well in the past, which results in the account being disabled immediately.

Accounts created in earlier versions of GemStone may have the lastLoginTime set to nil, which avoids the checks and reduces the risk in cases of repositories that have never had either check enabled. Accounts created in 2.4 or later have the lastLoginTime set to the time the account was created.

Workaround:

Decisions to enable or disabled account and password aging should be done with forethought, in any case.

To turn on account aging safely, an initial period with the stale account age limit set to a large value, or with password age check enabled but not account age limits, will allow accounts time to login in with updates to the lastLoginTime.

Version 2.4.4.7 and 3.0 and later have a method to explicity set the lastLoginTime, which should be used on existing accounts.

Bug 41404 - Write on closed socket may cause hang on AIX

Product: GemStone/S 64 Bit

Versions: 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.4, 2.4.x, 2.3.1.6, 2.3.x, 2.2.5.4, 2.2.x, 2.x

Platform All

Fixed In: 3.0, 2.4.8

Impact: Critical

If write is attempted on a closed socket, a SIGPIPE gets sent with the SA_RESTART flag set. However, on AIX, this does not restart the send(), which results in a infinite loop in the Stone or other calling process.

Bug 41341 - Encoded size ByteArray comparison method primitive failures

Product: GemStone/S 64 Bit

Versions: 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.4.3, 2.4.4, 2.4.3, 2.4.2, 2.4.1, 2.4, 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.x, 2.2, 2.1.4

Platform All platform

Fixed In: 3.0

When using one of the two encoded size ByteArray comparison methods with indexing arguments other than one, they will occationally fail with #rtErrPrimFailed (2258).

The problem methods are:

ByteArray>>compareStringAt:to:startingAt:sizeBytes:useCase:
ByteArray>>shortStringAt:compareWith:startingAt:opCode:

Using a value other than 1 as an argument to *StringAt: or startingAt: will expose the bug.

Workaround:

No workaround.

Bug 41283 - ProfMonitor over user action code can be biased

Product: GemStone/S 64 Bit

Versions: 3.6.3, 3.6.2, 3.6.1, 3.6, 3.5.7, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.x, 3.2, 3.1.0.6, 3.1.0.5, 3.1.x, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.x, 2.4.x, 2.4, 2.3.x, 2.2.5.4, 2.2.x

Platform All platforms

When the ProfMonitor sample timer fires while user action C code is executing, the sample is not actually taken until control returns back to smalltalk. This is usually when the user action code finishes and control has returned back to the calling smalltalk method. But if the user action code makes a smalltalk callback, the sample will be taken in that smalltalk method. In both cases this can result in a sampling bias toward these methods, leading to higher than expected tally counts and percentages for these methods.

Workaround:

Versions 3.3 and later allow real-time profiling as an alternative to elapsed time. Include #real in the array passed to the ProfMonitor method with the options keyword.

Bug 41267 - Cannot setup netldi to run as root without captive account on Macintosh

Product: GemStone/S 64 Bit

Versions: 3.6.3, 3.6.2, 3.6.1, 3.6, 3.5.7, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.x, 3.0.1, 3.0

Platform Darwin/Macintosh

Impact: Informational

When running netldi as root, on Darwin platform only, you cannot also use the captive account option -a.

Workaround:

On Darwin, avoid running netldi as root, instead run netldi in guest mode with a captive account

Bug 41261 - ProfMonitorTree MNU on profiling code that includes a user action

Product: GemStone/S 64 Bit

Versions: 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.4.3, 2.4.4.2, 2.4.4.1, 2.4.4, 2.4.3, 2.4.2, 2.4.1, 2.4

Platform All

Fixed In: 2.4.5, 2.4.4.8

When profiling code that contains a call to a user action, if samples are
taken within the user action code, the profile report will error with a
message not understood error on #'GCI' #'_classAndSelectorNameWidth:'

Workaround:

File in the following as SystemUser and commit

```
category 'Private'
method: ProfMonitorTree
printPME: aPME on: str total: theTotal

  str add: (self printablePercentFor: (aPME tally / theTotal)); space.
  str add: '('; add: aPME tally asString; add: ')'; space.

  aPME cmethod class == GsMethod
    ifTrue: [
      str
        add: (aPME cmethod _classAndSelectorNameWidth: 20);
        add: ' ['; add: aPME rcvrClass name; add: ']'; lf
      ]
      ifFalse: [      "for user actions, cmethod is #GCI"
        str addAll: (aPME cmethod asString width: 20); lf
      ]
%
```

Bug 41178 - Multi-threaded operations run single threaded on remote machines

Product: GemStone/S 64 Bit

Versions: 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.3, 3.1.0.2, 3.1.0.1, 3.1, 3.0.1, 3.0

Platform All

Fixed In: 3.3

Impact: Informational

If a multi-threaded operation is started from a gem running on a machine
that is remote from the stone, the operation will be performed by one thread,
regardless of the number of threads specified. This is due to limitations
in the pageserver repository I/O.

Workaround:

If you wish to use multi-threaded operations for performance, ensure that you are running on a machine local to the stone. You can check this by executing System sessionIsOnStoneHost

Bug 41116 - objErrAlreadyExists during dead object reclaim

Product: GemStone/S 64 Bit

Versions: 2.4.4.3, 2.4.4.2, 2.4.4.1, 2.4.4, 2.4.3, 2.4.2, 2.4.1, 2.4, 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.4, 2.2.3, 2.2.2, 2.2.1, 2.2

Platform All platforms

Fixed In: 2.4.4.4

Impact: Critical

During the time period when a GcGem is reclaiming dead objects, a session using GBS to create new objects containing circular references may inadvertently fail with error 2105 (#objErrAlreadyExists).

Workaround:

Commit/Abort and repeat the operation. The error is caused when an oop from a recently reclaimed dead object is given to the gem for reuse, the gem has an older view of the repository where the oop is still in use, and the new object is part of a circular reference loop (for example, object A references object B which references object A).

Committing/Aborting will refresh the view, while repeating the operation will likely use different oops for the new objects.

Bug 40939 - User creation fails if no #Published in DataCurator's SymbolList

Product: GemStone/S 64 Bit

Versions: 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.4.3, 2.4.4, 2.4.3, 2.4.2, 2.4.1, 2.4, 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.4, 2.2.x, 2.2, 2.1.5, 2.1.4, 2.1.x, 2.0.x, 1.2.6, 1.2.5, 1.x

Platform All

Fixed In: 3.0, 2.4.5

The code that sets up the account for a new user expects to find the SymbolDictionary #Published in DataCurator's SymbolList.

If you remove the #Published SymbolDictionary from DataCurator's SymbolList, new user creation will result in an error.

Workaround:

Restore the Published SymbolDictionary to DataCurator's SymbolList.

Bug 40880 - statmonitor with -r option problem behavior on stone restart

Product: GemStone/S 64 Bit

Versions: 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.4.3, 2.4.4, 2.4.3, 2.4.2, 2.4.1, 2.4, 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.x, 2.x

Platform All

Fixed In: 3.0, 2.4.5

If the stone is restarted while statmonitor is running with the -r option,
it will usually correctly start monitoring the restarted stone on a new
statmonitor data file.

However, the timing of this is not reliable, so under some circumstances,
depending on platform and how fast the system is, statmonitor may SEGV,
or the new statmonitor output file may be corrupt when old data is written
to the new file.

Workaround:

Shut down statmonitor if the stone is stopped and restarted.

The fix for this bug is to not allow statmonitor to continue to run after
the stone is shut down; when the stone shuts down, statmonitor will exit
also and needs to be manually restarted.

Bug 40859 - Newly created extents are not reclaimed

Product: GemStone/S 64 Bit

Versions: 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.4.3, 2.4.4, 2.4.3, 2.4.2, 2.4.1, 2.4, 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.x, 2.2, 2.1.4, 2.1.x, 2.1, 2.0.x, 2.0, 1.2.5, 1.2.4, 1.2.x, 1.1.14, 1.1.x

Platform All

Fixed In: 3.0, 2.4.5

Reclaim GcGems are assigned to specific extents when the GcGems are started.
If an extent is added programmatically, the new extent will not have a
Reclaim GcGem, and no pages on the new extent will be reclaimed until the
Reclaim GcGems are restared by an administrator. This can result in a
backlog of shadow pages.

Workaround:

Restart the Reclaim GcGems when an extents is added programmatically.

Bug 40837 - Parallel index creation does not update progress

Product: GemStone/S 64 Bit

Versions: 3.6.3, 3.6.2, 3.6.1, 3.6, 3.5.7, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.x, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.x, 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.x, earlier versions

Platform All

Parallel index creation allows creation of multiple indexes silmoutaneously.
Under some conditions, this may be faster than creating them individually,
although in most cases there is no improvement.

Due to the way parallel index creation is done, UnorderedCollection >>
progressOfIndexCreation does not return useful information when indexes
are created using parallel index creation.

Bug 40822 - Inherited class instance variables not copied on class versioning

Product: GemStone/S 64 Bit

Versions: 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.4.3, 2.4.4.2, 2.4.4.1, 2.4.4, 2.4.3, 2.4.2, 2.4.1

Platform All

Fixed In: 3.0, 2.4.5

On class versioning, in versions 2.4.4 and later the class instance variables are also copied to the new version. However, inherited class instance variables are not copied.

Bug 40815 - ExceptionSets break exception handling

Product: GemStone/S 64 Bit

Versions: 2.4.4, 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.4, 2.2.3, 2.2.2, 2.2.1, 2.2

Platform All

Fixed In: 2.4.4.2

Passing an ExceptionSet, rather than an Exception, to an on:do: or try:on:do: message, results in the exception not getting caught.

ExceptionSets are created when multiple exceptions are passed, for example:

```
[1 / 0 ]
  on: ZeroDivide, MessageNotUnderstood
  do: [ :ex | ... ].
```

Workaround:

file in the following:

```
category: 'ANSI support'
method: ExceptionSet
handlesCategory: anObject number: anInteger
    "This method may be overridden for ExceptionSelectors that are more discriminating.
    For example, one might want to trap all GemStoneError numbers in a certain
range."

    ^selectors anySatisfy: [:each | each handlesCategory: anObject number:
anInteger].
%
category: 'Exceptions'
method: ExceptionHandler
try: aBlock on: anExceptionSelector do: anotherBlock
    "anExceptionSelector is generally a subclass of ExceptionA or
    an ExceptionSet containing subclasses of ExceptionA"

    | protectedBlock |
    protectedBlock := aBlock.
    exceptionSelector := anExceptionSelector.
    returnBlock := [:value | ^value].
    gsException := Exception
        category: exceptionSelector errorCategoryToCatch
        number: exceptionSelector errorNumberToCatch
        do: [:ex :cat :num :args |
            "The following code can return from the #'on:do:' message by using the
returnBlock"
            self caughtEx: ex number: num cat: cat args: args handler: anotherBlock.
            "If we fall through, the result of the above expression will be returned
by the #'signal' message."
        ].
    gsException handler: self.
    [true] whileTrue: [
        | newBlock |
        (newBlock := self doTryBlock: protectedBlock) notNil ifTrue: [protectedBlock
:= newBlock].
    ].
%
category: 'Block Evaluation'
method: ExecutableBlock
on: selector do: action
    "Try to evaluate the receiver, and should an exception occur which is
matched
    by selector (normally a class object which is a subclass of ExceptionA
but can
    also be an ExceptionSet with subclasses of ExceptionA), evaluate the
    <monadicBlock>, action, passing it the exception instance as its argument.

    This method is replaced by AnsiMisc.gs"

    ^(Globals at: #'ExceptionHandler') new
        try: self
        on: selector
        do: action.
%
category: 'ANSI support'
classmethod: MessageNotUnderstood
handlesCategory: anObject number: anInteger

    ^anObject == GemStoneError and: [anInteger == 2010].
%
category: 'ANSI support'
classmethod: ZeroDivide
handlesCategory: anObject number: anInteger

    ^anObject == GemStoneError and: [anInteger == 2026].
%
category: 'Block Evaluation'
method: ExecutableBlock
on: selector do: action
    "Try to evaluate the receiver, and should an exception occur which is
matched
    by selector (normally a class object which is a subclass of ExceptionA
but can
    also be an ExceptionSet instance with subclasses of ExceptionA), evaluate
the
    <monadicBlock>, action, passing it the exception instance as its argument."
    ^ExceptionHandler new
        try: self
        on: selector
        do: action.
%
```

Bug 40814 - Removing a user may leave segments owned by that user

Product: GemStone/S 64 Bit

Versions: 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.4.3, 2.4.4, 2.4.3, 2.4.2, 2.4.1, 2.4, 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.4, 2.2.x, 2.2, 2.1.5, 2.1.4, 2.1.x, 2.0.x, 1.2.6, 1.2.5, 1.x

Platform All

Fixed In: 3.0

When you create a new user, depending on which protocol is used to create it, it will be assigned a newly created Segment as its default segment, and ownership of this segment is set to the new user.

If you remove a user from AllUsers, the segment/s owned by this user are not modified, and continue to reference the removed user. This will prevent this user and objects created by this user from being garbage collected.

Workaround:

The User Administration tools in GBS transfer ownership of Segments to the user that is performing the operation.

Bug 40796 - DateAndTimeANSI >> dayOfWeek results are GMT not local

Product: GemStone/S 64 Bit

Versions: 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.4.3, 2.4.4, 2.4.3, 2.4.2, 2.4.1, 2.4, 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.4, 2.2.3, 2.2.2, 2.2.x, 2.1.4, 2.1.x, 2.0.x

Platform All

Fixed In: 3.0, 2.4.5

The results returned by DateAndTimeANSI >> dayOfWeek are in GMT, not local time.

Bug 40792 - Query performance degradation on Date due to incompletely cached value

Product: GemStone/S 64 Bit

Versions: 2.4.4, 2.4.3, 2.4.2, 2.4.1, 2.4, 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.x, 2.2, 2.1.x

Platform All

Fixed In: 2.4.4.2

The internal structures used to implement equality indexes, btrees, cache portions of the values of certain data types within their structure, to avoid faulting in the actual object for some comparisons.

The code that caches Date instances was incorrect, resulting in the year portion of the Date not being cached. This did not cause any functional errors, but performance could be impacted as the Date object always had to be faulted into memory for comparison.

Note that this is Date, not DateTime. DateTime values are not cached in the btree, so in most cases Date will provide better performance.

Workaround:

Clustering the Dates involved in an index will reduce page demand and improve performance.

Bug 40781 - Performance of ordering on multi-predicate indexed queries

Product: GemStone/S 64 Bit

Versions: 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.3, 3.1.0.2, 3.1.0.1, 3.1, 3.0.1, 3.0, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.4.3, 2.4.4, 2.4.3, 2.4.2, 2.4.1, 2.4, 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.4, 2.2.x, 2.2, 2.1.5, 2.1.4, 2.1.x, 2.0.x, 1.2.6, 1.2.5, 1.x

Fixed In: 3.0

Impact: Informational

When performing a multiple predicate indexed query, for the best performance, the predicates should be ordered so that the most restrictive predicate is evaluated first. Predicates are evaluated from last to first (not the intuitive evaluation order), so the most restrictive predicate should be last in the query code.

For example, in the following query

```
 AllCustomers select:
  {:customer |
  (customer.status==#valid) & (customer.lastName=aName)}
```

The query predicate on lastName is performed first, resulting in a much smaller set that the second predicate must perform the query over.

Workaround:

Please examine your existing multi-term queries to ensure the ordering is optional for performance. Note that the ordering has changed in 3.0, so you should reexamine your predicate terms for performance after upgrading from 2.x to 3.x.

Bug 40735 - transactionless reported as invalid transaction mode

Product: GemStone/S 64 Bit

Versions: 2.4.3, 2.4.2, 2.4.1, 2.4, 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.x, 2.2, 2.1.4, 2.1.x, 2.1, 2.0.x, 2.0, 1.2.5, 1.2.4, 1.2.x, 1.1.14, 1.1.x

Platform All

Fixed In: 2.4.4

Under some conditions of heavy memory use, due to insufficent protection from garbage collection of intermediate results within a C function, it was possible to get an error #'transactionless' is not recognized as a valid transactionMode. when setting the transactionMode to #transactionless. This did not affect other transaction modes, for which the symbols are reserved OOPs.

Workaround:

Retrying the operation should succeed.

Bug 40703 - Possible hot hang in Singletrip traversal from GBS

Product: GemStone/S 64 Bit

Versions: 2.4.3, 2.4.2, 2.4.1, 2.4, 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5

Platform GBS 7.2 and above

Fixed In: 2.4.4

When GBS 7.2 or above session is logged in to GemStone/S 64 Bit version 2.2.5 and above, under very specific circumstances a gem could hang consuming CPU when responding to a request from GBS.

The circumstances require a graph of previously-replicated objects forming a circular reference path, with a replication spec on one or more objects with a combined minimum level sufficient to reach around the circular path. If another previously-replicated object, not part of the circular path but referencing an object in the circular path, was modified on the server during the request from GBS, the gem would enter infinite recursion.

The immediate cause could be something otherwise innocuous such as an instance variable update.

Workaround:

Modifying the replication specs for involved objects to remove "min" settings will avoid the problem.

Bug 40684 - DateTime asTime loses milliseconds

Product: GemStone/S 64 Bit

Versions: 2.4.3, 2.4.2, 2.4.1, 2.4, 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.x, 2.2, 2.1.4, 2.1.x, 2.1, 2.0.x, 2.0, 1.2.5, 1.2.4, 1.2.x, 1.1.14, 1.1.x

Fixed In: 2.4.4

Creating a Time based on a DateTime instance loses the milliseconds component of the Time.

Bug 40664 - Hang in decodeFromUTF8 with invalid data

Product: GemStone/S 64 Bit

Versions: 2.4.3, 2.4.2, 2.4.1, 2.4, 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.x, 2.2, 2.1.4, 2.1.x, 2.1, 2.0.x, 2.0

Platform All

Fixed In: 2.4.4

The method String >> decodeFromUTF8 may hang if the receiver contains certain invalid UTF-8 sequences.

Bug 40575 - SortedCollection >> includes: may fail to find elements

Product: GemStone/S 64 Bit

Versions: 2.4.3, 2.4.2, 2.4.1, 2.4, 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.4, 2.2.3, 2.2.x, 2.2, 2.1.4, 2.1.x, 2.0.x, 1.x

Platform All

Fixed In: 2.4.4

SortedCollection includes may fail to find an element, if the elements in the SortedCollection define = according to one instance variable, but the SortedCollection uses a sort block that compares a different instance variable. This is true using a default sort block if the elements in the SortedCollection define <= to use a instance variable other than the one used for the = comparison.

This is since SortedCollection >> includes: performs a binary sort based on the sorted order, so if the argument does not compare as = to an element in the vicinty of the insert location, no further searching is done.

Workaround:

Use caution in defining the = and <= methods for elements that will be in a SortedCollection, and in the sortblock method. If there is a need to define these differently, <SortedCollection> asArray includes: will return the correct result. Alternatively, subclass SortedCollection and reimplement includes: as a linear search.

Bug 40559 - Class versions do not retain class (instance) variable values

Product: GemStone/S 64 Bit

Versions: 2.4.3, 2.4.2, 2.4.1, 2.4, 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.x, 2.2, 2.1.4, 2.1.x, 2.1, 2.0.5, 2.0.4, 2.0.3, 2.0.x, 1.2.4, 1.2.x, 1.1.x, 1.x

Platform All

Fixed In: 2.4.4

When a new version of a class is created, the values within class or class instance variables is not carried to the new class version.

Bug 40553 - Migrating invariant objects in indexes may corrupt indexes

Product: GemStone/S 64 Bit

Versions: 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.3, 3.1.0.2, 3.1.0.1, 3.1, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.4.3, 2.4.4, 2.4.3, 2.4.2, 2.4.1, 2.4, 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.x, 2.2, 2.1.x

Platform All

Fixed In: 3.2

Invariant objects may be added to indexed collections and participate in indexes, but they do not get dependencyMaps since they cannot be modified. However, it is currently permitted to migrate invariant objects. If invarient objects are in an indexed collection, and are migrated, the index may not be correctly updated due to the lack of dependency list. This means the index may be corrupted regarding the invariant object.

Workaround:

Do not migrate instances that are invariant and participate in indexed collections. Drop indexes on collections that contain invariant objects prior to migration, and re-index after migration is complete.

Bug 40550 - copyReplaceFrom:to:with: incorrect for empty with: argument

Product: GemStone/S 64 Bit

Versions: 2.4.2, 2.4.1, 2.4, 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.x, 2.2, 2.1.2, 2.1.x, 2.0.x, 1.2.4, 1.x

Platform All

Fixed In: 2.4.3

If the with: argument to SequenceableCollection >> copyReplaceFrom:to:with: is an empty collection, this method incorrectly returned the receiver.

Workaround:

File in the following as SystemUser and commit:

```
category: 'Copying'
method: SequenceableCollection
copyReplaceFrom: startIndex to: stopIndex with: aSequenceableCollection

  "Returns a copy of the receiver in which all elements in the
  receiver between indexes startIndex and stopIndex inclusive have
  been replaced by those contained in aSequenceableCollection."

  | selfSize seqCollSize targetIndex result |
  startIndex > stopIndex ifTrue:
    [^startIndex _error: #rtErrBadCopyFromTo args: #[stopIndex]].

  seqCollSize := aSequenceableCollection size.
  result := self species new.
  targetIndex := 1.

  startIndex > 1 ifTrue:
    [self
        copyFrom: 1
        to: startIndex - 1
        into: result
        startingAt: targetIndex.
     targetIndex := targetIndex + startIndex - 1].

    0 < seqCollSize ifTrue: [
         aSequenceableCollection
              copyFrom: 1
              to: seqCollSize
              into: result
              startingAt: targetIndex.
    ].
  targetIndex := targetIndex + seqCollSize.

  selfSize := self size.
```

```
   stopIndex < selfSize ifTrue:
      [self
          copyFrom: stopIndex + 1
          to: selfSize
          into: result
          startingAt: targetIndex].

   ^result
%
```

Bug 40503 - Risk in modifying GsCurrentSession>>initialize

Product: GemStone/S 64 Bit

Versions: 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.3, 3.1.0.2, 3.1.0.1, 3.1, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.4.3, 2.4.4, 2.4.3, 2.4.2, 2.4.1, 2.4, 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.4, 2.2.3, 2.2.2, 2.2.1, 2.2

Fixed In: n/a

Impact: Critical

The method GsCurrentSession>>initialize provides a way for you to add application specific behavior for every login to the repository in version 2.x and 3.0.x.

However, unhandled errors in the application-specific modifications may cause session initializaion to be incorrect for all subsequent logins to the repository. This incorrect initialization may allow login, but disable all commits for this session. This is not recoverable; you must restore from backup taken before the modifications were committed.

In version 3.1, loginHook: has been added to UserProfile instances allowing you to more safely attach login behavior. In version 3.1 and later, you should not modify GsCurrentSession>>initialize.

Workaround:

There is no workaround, you must restore from backup to recover. If this is not possible for critical applications, contact GemStone Technical Support.

Make a backup of the repository prior to making any modifications to this method.

Leaving a separate idle session logged in as SystemUser, while modifications are being tested, will allow these modifications to be removed to recover the system.

Only executing the modifications for users other than SystemUser, and handling errors in the modifications, will also help avoid situations requiring restore from backup. For example,

```
(self userProfile userId = 'SystemUser') ifFalse: [
  [user code]
    on: (Globals at: #Error)
    do: [:ex | ex return]
   ].
```

Bug 40473 - Simultaneous logins by the same userId can result in disabled account

Product: GemStone/S 64 Bit

Versions: 2.4.2, 2.4.1, 2.4, 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.x, 2.2, 2.1.4, 2.1.x, 2.1, 2.0.x, 2.0

Fixed In: 2.4.3

If two logins for the same userId occur within the same clock second, a logic error in the code compares the password age, rather than the time since last login, to the staleAccountAgeLimit. So, if the time since the last password change is longer than the staleAccountAgeLimit, the account's login may be disabled with the reason "StaleAccount".

This bug requires a passwordAgeLimit that is larger than the staleAccountAgeLimit, as well as near-simultaneous logins by a single userid, and so is rare in practice.

Workaround:

Setting a passwordAgeLimit that is smaller than the staleAccountAgeLimit will avoid the problem. Or, manually resetting the password (to the same password, if that is allowed by the security configuration), will reset the password change date.

Bug 40454 - Failed commit has small risk of losing data

Product: GemStone/S 64 Bit

Versions: 2.4.1, 2.4, 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.x, 2.2, 2.1.4, 2.1.x, 2.1, 2.0.x, 2.0

Platform All

Fixed In: 2.4.2

Impact: Critical

If a failed commit is not immediately followed by an abort, there is a pathway through the code where a newly created object may not be written out to the repository. This results in object does not exist errors, and if the OOP is re-used, logical corruption.

This bug is in GCI code so affects GBS applications. While conditions for this bug include modifications to objects to refer to newly created objects, commit failure, and later successful commit, there are other conditions required; this bug appears to be rare and difficult to reproduce.

Workaround:

Ensure that all commit failures are followed immediately by abort.

Bug 40442 - UnorderedCollections upgraded from 32-bit may fail audit

Product: GemStone/S 64 Bit

Versions: 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.4.3, 2.4.4, 2.4.3, 2.4.2, 2.4.1, 2.4, 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.x, 2.2, 2.1.4, 2.1.x, 2.1, 2.0.x, 2.0, 1.2.5, 1.2.4, 1.2.x, 1.1.14, 1.1.x

Platform All

Fixed In: 2.4.5

UnorderedCollections (NSCs or NonSequentialCollections) with more than 2K elements are composed internally with a tree structure. The conversion from 32-bit GemStone/S did not correctly set a flag value in an interior node of this tree. This did not cause any problems with the behavior of the collection; this value was not actually used in operations on the collection.

However, running NSC audit code:
  <anUnorderedCollection> auditInternalStructures
  <anUnorderedCollection> auditInternalStructuresWithRepair: true would
report a failure.

Also, running with slow (debugging) executables could result in assertion failures.

Workaround:

The NSC repair code will fix this problem:
<anUnorderedCollection> repairInternalStructures
<anUnorderedCollection> auditInternalStructuresWithRepair: true

Bug 40430 - libposix-aio version mismatch may cause extreme numbers of small tranlogs

Product: GemStone/S 64 Bit

Versions: 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.4.3, 2.4.4, 2.4.3, 2.4.2, 2.4.1, 2.4, 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.4, 2.2.3, 2.2.2, 2.2.1, 2.2

Platform Linux

Fixed In: 3.0, 2.4.7

When the optimized POSIX AIO library is installed, GemStone will automatically use it for asynchronous I/O, which can improve performance.

However, if there is a difference in versions between a the customer's libposix-aio.so and the posix-aio header files against which GemStone was compiled, it can cause bad behavior from the AIO subsystem. This has been observed to cause the creation of vast numbers of very small transaction logs (1536 bytes).

Workaround:

If you observe the symptoms - many small tranlogs - remove the libposix-aio.so to verify the cause, or start the stone with the special -s option. You can run without it, or replace it with the correct version of libposix-aio.so. If you are unable to determine the appropriate version, contact GemStone Technical Support.

Bug 40417 - Statmonitor data does not include system stats for linked topaz with s-bit set

Product: GemStone/S 64 Bit

Versions: 3.6.3, 3.6.2, 3.6.1, 3.6, 3.5.7, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.x, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.4.3, 2.4.4, 2.4.3, 2.4.2, 2.4.1

Platform All

System statistics may not be collected for linked topaz sessions, if topaz is started as a user other than the user owning the GemStone processes, the users are not in the same group, and the s-bit is set for the topaz executable. This is since the /proc/<pid> files that contain the system stats information are owned by the userId that starts topaz. If the userId that started statmonitor does not have permission to access these files, no system statistics are available.

Workaround:

Running statmonitor as root will allow it to access the files.

Bug 40401 - Stone shutdown during restoreFromLogs may corrupt extents

Product: GemStone/S 64 Bit

Versions: 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.4, 2.2.3, 2.2.x, 2.2, 2.1.4, 2.1.x, 2.0.x, 1.2.4, 1.2, 1.1.x, 1.x

Fixed In: 2.4

When the stone is in restoring transaction logs, under some rare internal conditions the pageservers may not write all the required pages to the extents that are needed as part of a checkpoint. If the stone is shut down, this missing information may cause the stone to fail to restart. This problem may occur during orderly shutdowns as well as crashes. If the stone does not shutdown, and a later checkpoint is correct, there are no consequences of this bug.

This bug has never been observed in customer systems, it was uncovered during testing.

Workaround:

If the problem occurs, the extents are corrupt, and you will need to restart from the backup and replay all the transaction logs again.
Reducing the number of times you shut down a warm standby reduces the risk.

Bug 40394 - continueTransaction can result in stone crash

Product: GemStone/S 64 Bit

Versions: 2.4.1, 2.4, 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.x, 2.2, 2.1.4, 2.1.x, 2.1, 2.0.5, 2.0.4, 2.0.x, 2.0

Platform All

Fixed In: 2.4.2

Impact: Critical

Transactions with many indexing updates that execute continueTransaction could potentially encounter a stone crash during the subsequent commit.

Updates to indexes are written to a buffer. The internal record keeping regarding the transaction status did not correctly handle the case where this buffer was filled up before a continueTransaction was executed. This situation resulted in a missing "begin" record. Later, when this transaction is committing, the stone discovered the missing "begin". To avoid risk of corruption, the stone crashed, with error messages in the stone log error including:

In cancelBeginRecord: no outstanding begin for session NN

Workaround:

Avoid use of continueTransaction, particularly with large transactions involving indexing updates.

Bug 40389 - GsFile >> flush may incorrectly appear to succeed on disk full from Linux

Product: GemStone/S 64 Bit

Versions: 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.3, 3.1.0.2, 3.1.0.1, 3.1, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.4.3, 2.4.4, 2.4.3, 2.4.2, 2.4.1, 2.4, 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.4, 2.2.3, 2.2.2, 2.2.1, 2.2.x, 2.1.x, 1.2.5, 1.2.4, 1.x

Platform Linux

Fixed In: n/a

For sessions running on Linux, a GsFile >> flush may not return nil (implying that the flush succeed), when the flush did not occur and no bytes were written to disk. This is the case if the target instance of GsFile is open on a disk that is full, and cannot be written to.

This is an error in the Linux operating system call fflush().

Workaround:

The GsFile >> close method will error if the disk is full.

Bug 40364 - GsFile operations on disk full cause gem crash

Product: GemStone/S 64 Bit

Versions: 2.4, 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.4, 2.2.3, 2.2.2, 2.2.x, 2.1.4, 2.1, 2.0.x, 1.2.5, 1.2.4, 1.2.x, 1.x

Platform Linux

Fixed In: 2.4.1

On Linux, if GsFile operates on a file that is on a disk that is completely full, subsequent session logout will cause the gem session to crash with the error "double free or corruption". This crash may require the gem or linked session to be terminated with a kill -9.

The problem is due to specifics of linux I/O. When a file is opened on the OS, the gem session keeps a reference so that it can be closed on logout, even if the GsFile is left open. When the file that is opened is on a full disk, the OS closes it immediately. Then when the session logs out, it requests the close again. Most operating systems ignore the second close, but linux performs the second close, resulting in a fatal memory deallocation error.

This behavior may not be present on all distributions of Linux

Workaround:

Avoid accessing full disks from Linux.

Bug 40273 - DateTimes created with milliseconds >= 2^31 lose precision

Product: GemStone/S 64 Bit

Versions: 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.4, 2.2.3, 2.2.2

Platform All

Fixed In: 2.4

If the millisecond field in a DateTime is set to a value equal to or greater than 2^31 (2147483648), the resulting DateTime loses the millisecond component.

Bug 40255 - STN_COMMIT_QUEUE_THRESHOLD configured value not used

Product: GemStone/S 64 Bit

Versions: 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.x, 2.2.5, 2.2.4, 2.2.x, 2.2, 2.1.4, 2.1.x, 2.1, 2.0

Platform ALL

Fixed In: 2.4

The configurative parameter STN_COMMIT_QUEUE_THRESHOLD is overwritten internally and as a result, the configured value is not actually used. The configured setting may be reported in the stone log and returned by the runtime query, masking the problem.

Bug 40217 - Unclear error for incorrect path with parallel index creation

Product: GemStone/S 64 Bit

Versions: 3.6.3, 3.6.2, 3.6.1, 3.6, 3.5.7, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.x, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.x

Platform All

If you are creating indexes using the parallel index creation protocol, and there is an incorrect path specification for an index that is not the first one in the list of indexes to be created, the error message does not provide useful information for tracking down the actual cause of the error.

The errors may be "Illegal attempt to execute a protected method" or "Attempt to insert a key into the B-tree that was an invalid class for which the B-tree was created".

Bug 40201 - Avoid GciInterface class >> _objectForOop:

Product: GemStone/S 64 Bit

Versions: 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.x, 2.2, 2.1.5, 2.1.4, 2.1.x, 2.0.x, 1.2.6, 1.2.5, 1.x

Platform All

Fixed In: 2.4

Impact: Informational

The methods GciInterface class >> _objectForOop: and GciInterface class >> _oopForObject: are not supported in GemStone/S 64 Bit, and the results should not be relied on.

Workaround:

Use the equivalent Object methods Object class >> _objectForOop: and Object >> asOop.

Bug 40178 - MFC may run in transaction on Linux

Product: GemStone/S 64 Bit

Versions: 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.x, 2.2, 2.1.5, 2.1.4, 2.1.x, 1.2.6, 1.2.5, 1.2.x

Platform Linux

Fixed In: 2.4

When markForCollection is started, the MFC code normally takes the session out of transaction, so it can respond to sigAborts and avoid a commit record backlog.

On Linux, the code that took the session out of transaction silently failed. As a result, if the session that started an MFC is in transaction - such as one started with the default transaction mode, which is #autobegin - it performs the markForCollection in transaction, and is immune for sigAborts. If there is other activity on the repository, this may result in a large commit record backlog, repository growth and potentially repository shutdown due to running out of space.

Workaround:

Before running a markForCollection, manually go out of transaction. For example, execute the following:

System transactionMode: #manualBegin

Bug 40160 - pageaudit with -f reports success after displaying failure messages

Product: GemStone/S 64 Bit

Versions: 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.x, 2.2, 2.1.5, 2.1.4, 2.1.x, 2.0.x, 1.2.6, 1.2.5, 1.x

Platform All

Fixed In: 2.4

Using pageaudit -f option causes the pageaudit to continue to audit after encountering an error. However, the final report of pageaudit in this case may say the pageaudit was successful, although earlier output indicates audit failures were found.

Workaround:

If running pageaudit with the -f option, carefully check all output rather than relying on the final message.

Bug 40122 - MFC may ignore sigAborts during initial stage

Product: GemStone/S 64 Bit

Versions: 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.4, 2.2.3, 2.2.x, 2.1.4, 2.1.x

Platform All

Fixed In: 2.4

Impact: Critical

During an initial stage of MFC, the MFC gem reads the dependency map, which contains information on indexes. This is normally fast, but on large systems with many/very large indexes, it may take some time to read this table.

During this brief initial stage of MFC, the MFC gem may ignore a sigAbort from the Stone. The stone, having sent the sigAbort and not having received acknowlegement, does not send a second sigAbort, even if the commit record backlog gets very high and the MFC gem is holding the oldest commit record.

If the MFC gem does not get a sigAbort during the dependencyMap state, once the MFC progressCount begins to increment, the MFC gem will respond to sigAborts normally.

Workaround:

Starting MFC during periods of no or low activity, when the commit record backlog limit is not reached and so the stone is not sending sigAborts, avoids the problem.

Sending a manual sigAbort to the MFC session, once it is past the dependencyMap read state, will force the session to abort and release the commit record.

System Class >> sendSigAbortToSession: (mfcSessionId negated)

Bug 40096 - Risk of Stone hang or crash on Gem SIGTERM

Product: GemStone/S 64 Bit

Versions: 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.x, 2.2, 2.1.5, 2.1.4, 2.1.x, 2.1, 2.0.x, 1.2.6, 1.2.5, 1.2.x, 1.x

Platform All

Fixed In: 2.4

On a heavily loaded system, if a Gem on the same host as the Stone gets a SIGTERM while waiting for a shared memory (SMC) response from the Stone, there is a possible race condition that can hang or bring down the Stone. This is because the same semaphore is used to signal Gems waiting for SMC communcations, and waiting for spin locks.

The SIGTERM interrupts the Gem's wait for the SMC response from the Stone, and the Gem continues handling the SIGTERM, which can involve waiting on spin locks to get page frames, etc. as part of shutdown. However, when the Stone finally completes the SMC response and signals the Gem, the Gem assumes this is the signal that the spin lock it was waiting for is now is available, resulting inspin lock problems that may hang or crash the Stone, depending on the specific spin lock.

Workaround:

This is a rare condition; the system can recover from some times of stuck or problem spin locks.

Tuning your system to lessen the number of waits, offloading Gems to a Gem server machine, and avoiding sending SIGTERMs to Gems, will make the problem less likely.

There is no risk of corruption resulting from this bug.

Bug 40035 - Corrupt object error triggered by GBS flush operations

Product: GemStone/S 64 Bit

Versions: 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.x, 2.2, 2.1.5, 2.1.4, 2.1.x, 2.1, 2.0.x, 1.2.6, 1.2.5, 1.2.x, 1.2, 1.1.14, 1.1.x, 1.1, 1.0

Platform All

Fixed In: 2.3.1.6

Impact: Critical

There is an obscure bug in the gem process that in very rare conditions can cause GBS object flushes to the gem to trigger corrupt object errors of the form:

GbsObjErrCorruptObj - The object with object ID XXXXXXX is corrupt.
Reason: "attempt to modify object with invalid view"

Attempts to commit after this error will trigger:

GbsRtErrCommitDisallowed - Further commits have been disabled for this session because: 'CorruptObj error'

There is no actual corruption, but your session will need to log out and any work done in that session cannot be commited.

Workaround:

No workaround. If your application is subject to this error, please contact GemStone Techical Support.

Bug 40030 - Remote login authorization failures

Product: GemStone/S 64 Bit

Versions: 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.3, 3.1.0.2, 3.1.0.1, 3.1, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.x, 2.4, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.4, 2.2.3, 2.2.2, 2.2.1, 2.2

Platform All

With the netldi running as root, logins where the gem is on a different node than the stone, including remote linked logins as well as remote RPC logins, may encounter authorization failures in starting up one of the pgsvr processes. This will result in errors on login similar to: "Repository is not attached".

Workaround:

Running the netldi in guest mode with a captive account does not have this problem.

Setting the authorization an environment variable allows login, e.g.:
   setenv GEMSTONE_NRS_ALL "#auth:username@password"

The authorization lookup is done on machine IP address rather than name.
In the .netrc file, use the IP address rather than (or in addition to) the machine name.

Bug 40025 - Indexes incorrect for LargePositiveIntegers in SmallInteger range

Product: GemStone/S 64 Bit

Versions: 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.x, 2.2, 2.1.x, 2.0.x

Platform All

Fixed In: 2.4

The indexed comparision for SmallIntegers and LargeIntegers that
are within a portion of the new range of SmallIntegers is incorrect.
This results in incorrect indexed query results.

LargeIntegers that are within the new SmallInteger range are normally
only present if you have upgraded from an earlier version of GemStone without specifically finding and converting LargeIntegers. Using SmallIntegers is faster and smaller, we strongly recommend that
this conversion be done.

Workaround:

Convert any references to LargeIntegers that are within the SmallInteger range into SmallIntegers, and rebuild the indexes on that value.

Bug 40004 - Race condition in stone startup in recovery mode can corrupt tranlog

Product: GemStone/S 64 Bit

Versions: 3.0.1, 3.0, 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.4.3, 2.4.4.2, 2.4.4.1, 2.4.4, 2.4.3, 2.4.2.1, 2.4.2, 2.4.1, 2.4

Platform All Platforms

Fixed In: 3.1, 2.4.6

There is a very rare race condition present where a gcgem can attempt to
write a tranlog record while the system is transitioning from recovery
to normal operation. This can leave a block of zeros written to the tranlog
that will cause later attempts to read the tranlog to become confused and
think the EOF is at this point. This applies to system recovery, copydbf,
and attempts to read the tranlog via the printlogs.sh and searchlogs.sh
tranlog analysis scripts.

For printlogs/searchlogs, the following entry will be present:

validateLogRead: EOF because pageKind(0) != REP_LOG_RECORD(16) at XXXX.X

Bug 40001 - GsFile >> contents returns nil for empty file

Product: GemStone/S 64 Bit

Versions: 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.4, 2.2.3., 2.2, 2.x

Platform All

Fixed In: 2.4

If a physical file is empty (the file system size is 0), GsFile >> contents
for that file returns nil instead of an empty string.

Bug 40000 - SIGTERM log message in seconds offset rather than DateTime

Product: GemStone/S 64 Bit

Versions: 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.x, 2.2, 2.1.5, 2.1.4, 2.1.x, 2.1

Platform All

Fixed In: n/a

Impact: Informational

When a SIGTERM is sent to a process, this is recorded in the process log with something similar to:

```
  GS Signal Handler: Signal 15, SIGTERM, Software Termination Signal
  Received at 1242925247 in thread 1
```

The timestamp is printed in the internal time, which is seconds offset since January 1, 1970. This is not translated into a readable timestamp, to avoid the small risk of system calls deadlocking as a result of that processing.

Workaround:

To translate the seconds offset into a DateTime, use the following expression:

```
(DateTime fromStringGmt: '01/01/1970 00:00:00') addSeconds: <secondsOffset>
```

This will give you the time in GMT, which you may then need to translate into your local time.

Bug 39980 - SelectBlock queries may error due to incorrect method names

Product: GemStone/S 64 Bit

Versions: 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.3, 2.2.x, 2.2, 2.1.x, 2.1

Platform All

Fixed In: 2.4

Certain queries using select blocks may encounter does not understand errors, due to incorrectly updated method names.

Only a few specific queries are affected; queries with multiple terms, of which two are internally condensed into a range query, and in which the internal evaluation order results in the range query not being the first predicate evaluated.

Workaround:

file in the following as SystemUser and commit:

```
category: 'Indexing Support'
method: Object
_idxGreaterThanAndLessThanValues: valueArray

"This comparison operation is by the indexing subsystem to determine if the
receiver is within the given range of values.  The 'valueArray' argument is a
four-element Array consisting of #[ val1, bool1, val2, bool2 ].  val1 is the
lower bound value and bool1 specifies whether equal values are allowed.  val2
is the upper value and bool2 specifies whether equal values are allowed.
Returns true if the receiver satisfies the criteria."

(valueArray at: 2)
   ifTrue: [
      (self _idxForCompareGreaterThanOrEqualTo: (valueArray at: 1))
```

```
        ifTrue: [
           (valueArray at: 4)
              ifTrue: [ ^ self _idxForCompareLessThanOrEqualTo: (valueArray at: 3) ]
              ifFalse: [ ^ self _idxForCompareLessThan: (valueArray at: 3) ]
        ]
        ifFalse: [ ^ false ]
  ]
  ifFalse: [
      (self _idxForCompareGreaterThan: (valueArray at: 1))
        ifTrue: [
           (valueArray at: 4)
              ifTrue: [ ^ self _idxForCompareLessThanOrEqualTo: (valueArray at: 3) ]
              ifFalse: [ ^ self _idxForCompareLessThan: (valueArray at: 3) ]
        ]
        ifFalse: [ ^ false ]
  ]
%
```

Bug 39943 - Index audit did not catch corruption in Btree cached values

Product: GemStone/S 64 Bit

Versions: 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.x, 2.2.5, 2.2.4, 2.2.x, 2.2, 2.1.4, 2.1.x, 2.1

Fixed In: 2.4

Btrees cache a portion of the value within their structures, which in cases of obvious differences between values, avoids the need to fetch the object itself and thus improves query performance.

However, the index audit code audits the actual value, not the cached value. If the cached portion in the btree was incorrect, while the actual value was correct, indexed queries could return incorrect results while the index audit did not detect the problem.

A case of such discrepency between the actual and cached values occurs when upgrading a repository with indexes from version 2.2.x or earlier to version 2.3 or later. Due to the introduction of QuadByteStrings in version 2.3, the btree structures were modified, resulting in corruption. See bugnote for 39915 for details.

Bug 39941 - Gem building an index may be killed as if idle

Product: GemStone/S 64 Bit

Versions: 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.x, 2.2, 2.1.4, 2.1.3, 2.1.2, 2.1.1, 2.1

Platform All

Fixed In: 2.4

If the time it takes for a gem to build an index is longer than the STN_GEM_TIMEOUT,
the gem may be killed by the stone in spite of the gem's activity.

Workaround:

Increase the STN_GEM_TIMEOUT while building large indexes.

Bug 39920 - UserProfile >> isDisabled may return false negative

Product: GemStone/S 64 Bit

Versions: 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.x, 2.2.5, 2.2.4, 2.2.x, 2.2, 2.1.4, 2.1.x, 2.1, 2.0

Platform All

Fixed In: 2.4

If a UserProfile meets the criteria to be disabled, such as an expired password, but has not made an failed attempted to log in, the internal account status remained active and the isDisabled method incorrect returned false.

Bug 39915 - Indexes must be rebuilt after upgrade from pre-2.3 versions

Product: GemStone/S 64 Bit

Versions: 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.4.3, 2.4.4, 2.4.3, 2.4.2, 2.4.1, 2.4, 2.3.1.6, 2.3.1, 2.3

Platform All

Fixed In: n/a

Impact: Critical

As part of the implementation of Extended Characters (QuadByteStrings) in version 2.3, internal indexing structures were modified to allow additional bytes in the lookup caches. As a result, if indexed collections are upgraded from a version earlier than 2.3, to 2.3. or later, without index rebuild, indexed queries on these collections will produce incorrect results or generate an error.

There are no errors during the upgrade itself, and auditIndexes does not detect this problem.

Workaround:

When upgrading from a version prior to 2.3 to version 2.3 or later, indexes must be dropped prior to the upgrade and rebuild afterwards.

This step is incorrectly omitted from the upgrade instructions in the Install Guide.

Bug 39914 - Index creation fails for ScaledDecimals or Fractions outside SmallInteger range

Product: GemStone/S 64 Bit

Versions: 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.x, 2.2.5, 2.2.4, 2.2.3, 2.2.2, 2.2.x, 2.2, 2.1.4, 2.1.x, 2.1, 2.0.4, 2.0.2, 2.0

Platform All

Fixed In: 2.3.1.6

If an index is created with a last element class of ScaledDecimal or Fraction, and the ScaledDecimal or Fraction contains a denominator outside the SmallInteger range (larger than 1152921504606846975), then index creation will fail with the error:
  ObjErrCorruptObj - The object with object ID 20 is corrupt. Reason: 'FetchSmallInt_ bad value'

Workaround:

Normalize the ScaledDecimal or Fraction to reduce the denominator to less than 1152921504606846975.

Bug 39913 - Passivating a SmallFloat results in an error

Product: GemStone/S 64 Bit

Versions: 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.x, 2.2, 2.1.x, 2.0.x

Fixed In: 2.3.1.6

Attempting to passivate an instance of SmallFloat results in a message not understood error on asStringLocaleC

SmallFloats are deprecated in GemStone/S 64 Bit v2.0 and above, but may exist as a result of conversion from an earlier version or product.

Bug 39880 - System>>_getAndClearGciDirtySet: may encounter ObjDoesNotExist

Product: GemStone/S 64 Bit

Versions: 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.4, 2.2.3, 2.2.x, 2.2, 2.1.4, 2.1.x, 2.0.x, 1.x

Platform All

Fixed In: 2.3.1.6

Impact: Critical

If an object is removed from the PureExportSet, subsequent fetch of the ExportedDirtySet still attempts to return the removed object. If the removed object has been garbage collected, this will result in an object does not exist error.

Bug 39849 - Conversion failure on repositories originating in gss 4.x and upgraded from 6.2 or later

Product: GemStone/S 64 Bit

Versions: 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.4, 2.2.3, 2.2.2

Platform All

Fixed In: 2.3.1.6

Repositories originating in much earlier versions of GemStone/S may have Segment layouts in SystemRepository that are very different from the current GS/S 5.1.x and 6.x layouts. While GemStone/S is flexible about Segment order, GemStone/S 64 Bit versions 2.2 and above, which support Segments, require the more recent Segment layout.

The conversion process from 6.1.x to GemStone/S 64 Bit correctly rearranged the Segments as part of the preconversion process. Code was inadvertently omitted in the preconversion script to convert from 6.2 and above to GemStone/S 64 Bit, so conversion fails.

Workaround:

Contact Gemstone Technical Support for the updated version of convprep6x.

The fix is to add the line

```
  input $GEMSTONE_64/upgrade/segmentorder_61to20.gs
```

to convprep6x at line 201, following the filein of SmallDouble.

Bug 39826 - _sessionCacheStatAt:put: truncates large SmallIntegers

Product: GemStone/S 64 Bit

Versions: 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.x, 2.2, 2.1.4, 2.1.x, 2.1, 2.0.5, 2.0.4, 2.0.x, 2.0

Fixed In: 2.4

When passed as an argument System class >> _sessionCacheStatAt:put:, SmallIntegers larger than 4294967296 (2^32) have the bits above 2^32 discarded.

Workaround:

If using very large values in _sessionCacheStatAt:put:, round appropriately, or split the number and use two slots if exact precision is required.

Bug 39802 - addAllToStoneLog: errors for large arguments

Product: GemStone/S 64 Bit

Versions: 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.4, 2.2.3, 2.2.2, 2.2.x, 2.1.4, 2.1.x, 2.0.4, 2.0.x, 1.2.5, 1.2.4, 1.1.14, 1.x

Platform All

Fixed In: 2.4

Calling the method System class >> addAllToStoneLog: with an argument of length greater than 16269 results in an error.

Workaround:

File in the following as SystemUser and commit.

```
classmethod System
addAllToStoneLog: aString

"Appends text to the Stone's informational log file.  First, this method writes
a banner that identifies the session from which aString came.  It then appends
aString itself.  The argument must be a kind of String or DoubleByteString."

| logicalSize chunkSize offset endOffset |

aString basicSize <= 16270 "virtual machine constant" ifTrue:[
  ^ self _addAllToStoneLog: aString
  ].
aString _validateClasses:#[ String, DoubleByteString].
logicalSize := aString size .
chunkSize := 16270 "virtual machine constant" .
(aString isKindOf: DoubleByteString) ifTrue:[ chunkSize := 16270 / 2 ].

offset := 1.
[offset <= logicalSize] whileTrue:[
  endOffset := offset + chunkSize - 1 .
  endOffset > logicalSize ifTrue:[ endOffset := logicalSize ].
  self _addAllToStoneLog: (aString copyFrom: offset to: endOffset).
  offset := offset + chunkSize
  ].
%
```

Bug 39760 - Warning in stone log for tranlogs on raw partitions

Product: GemStone/S 64 Bit

Versions: 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.x, 2.2, 2.1.5, 2.1.4, 2.1.x, 2.1, 2.0.x, 1.2.6, 1.2.5, 1.2.x, 1.1.x, 1.1, 1.0

Platform All

Fixed In: 2.4

Impact: Informational

When transaction logs are on raw partitions and the path does not start with /dev/, the stone log may include messages of the form:

```
 WARNING: for fileId NNNNNN,
          filename = /devices/pseudo/zfs@0:1c,raw
  This file is not a GemStone transaction log file:
  GemStone got page kind 0 but expected page kind 16 .
```

The check for raw partitions within this code is incorrect, and this warning can be ignored.

Bug 39750 - GEM_FREE_FRAME_CACHE_SIZE default is calculated incorrectly

Product: GemStone/S 64 Bit

Versions: 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.x, 2.2, 2.1.4, 2.1.x, 2.1, 2.0.x, 2.0

Fixed In: 2.4

The calculation of the default free frame cache size is incorrect for shared page cache sizes between 100MB and 200MB.

A setting of -1 for GEM_FREE_FRAME_CACHE_SIZE means to calculate a default of 0 for caches smaller than 100MB, and 10 for caches larger than 100MB. A calculation error introduced when the page increased from 8K to 6K resulted in a threshold of 200MB rather than 100MB being used in this calculation.

Workaround:

Explicitly setting the desired GEM_FREE_FRAME_CACHE_SIZE for your cache sizes avoids the incorrect calculation.

Bug 39743 - Core file may still be created

Product: GemStone/S 64 Bit

Versions: 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.4, 2.2.3, 2.2.2, 2.2.x, 2.1.4, 2.1.x, 2.0.4, 2.0.x, 1.2.5, 1.2.4, 1.1.14, 1.x

Platform All

Fixed In: 2.4

When GemStone processes encounter fatal internal errors and terminate, they should only create core files if GS_WRITE_CORE_FILE is set. However, under some conditions involving multiple or recusive SIGSEGV, SIGBUS, or SIGILL errors, a core file can still be generated. This can cause disk space problems with large caches.

Bug 39727 - findReferencePathToObject: can produce corrupt object error

Product: GemStone/S 64 Bit

Versions: 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.x, 2.2, 2.1.x, 2.0.x, 1.2.5, 1.2.4, 1.2.x, 1.2, 1.1.14, 1.x

Platform All

Fixed In: 2.4

When running Repository >> findReferencePathToObject:, it is possible for reference paths containing large collections to produce an invalid corrupt object error. This is due to incorrect handle of large objects (objects that do not fit on a page and therefore are implemented internally using special data structures).

Under these circumstances, the error does not indicate corruption in your repository, but the session will have to log out and cannot commit.

Bug 39725 - Spaces in cache name corrupts statmonitor file

Product: GemStone/S 64 Bit

Versions: 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.4, 2.2.3, 2.2.2, 2.2.x, 2.1.4, 2.1.x, 2.0.4, 2.0.x, 1.2.5, 1.2.4, 1.1.14, 1.x

Platform All

Fixed In: 2.4

If a gem or GCI app sets its cache name to a String containing spaces, a statmonitor generated from this system file is corrupt. If there are more than two spaces, the statmonitor file cannot be loaded by VSD. If the cache name has one or two spaces, it can be read by VSD, but the data for that session is incorrect; the statistic value is not reported under the correct statistic name.

Cache names are set using the method System _cacheName: <aString>.

Workaround:

Do not set cache names to Strings containing spaces.

Bug 39678 - Transactionless sessions result in shortened LostOT timeout

Product: GemStone/S 64 Bit

Versions: 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.x, 2.2, 2.1.x, 2.0.x, 1.2.5, 1.2.4, 1.2.x, 1.2, 1.1.14, 1.x

Platform All

Fixed In: 2.4

If all gems are in transactionless mode, the timeout interval is shortened to 1/4 the size of the configured STN_LOST_OT_TIMEOUT. As transactionless sessions were originaly designed for pooled gems, this was intended to keep inactive pooled gems current; but it also affects sessions in transactionless mode for other reasons. This has been updated to check for sessions logged in as Nameless user instead, which use should be limited to pooled gems

Bug 39672 - Signal handler chaining could pass incorrect number of arguments

Product: GemStone/S 64 Bit

Versions: 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.4, 2.2.3, 2.2.2, 2.2.x, 2.1.4, 2.1.x, 2.0.4, 2.0.x, 1.2.5, 1.2.4, 1.1.14, 1.x

Platform All

Fixed In: 2.4

When signal handlers are chained, either one or three arguments should be passed, specified in SA_SIGINFO. GemStone signal handlers did not check, and always passed three arguments. This could be incorrect for customer installed signal handlers.

Bug 39642 - Conversion causes hashed lookups on 8K Strings to fail

Product: GemStone/S 64 Bit

Versions: 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.4.3, 2.4.4, 2.4.3, 2.4.2, 2.4.1, 2.4, 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.x, 2.2, 2.1.5, 2.1.4, 2.1.x, 2.0.x

Platform All

Fixed In: n/a

String hash in GemStone/S and GemStone/S 64 Bit v1.x uses the first 8K
bytes of the String, while in GemStone/S 64 Bit v2.x, the first 16K bytes
are used.

If a GSS or GS64 1.x repository contains Sets that include Strings over
8K, or dictionaries containing keys that are Strings over 8K, after conversion
to GS64 2.x, hashed lookups on these keys will fail.

Workaround:

Avoid using Strings over 8K in Sets or as dictionary keys. If you must do so, manually rebuild the Set or dictionary after conversion.

Bug 39636 - Incorrect restore point in timeToRestoreTo:

Product: GemStone/S 64 Bit

Versions: 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.x, 2.2, 2.1.4, 2.1.x, 2.0.4, 2.0.x, 1.2.x, 1.1.14, 1.x

Platform All

Fixed In: 2.4

When the DateTime specified as an argument to Repository >> timeToRestoreTo: has a different offset from GMT as the current time, the calculation of the seconds offset, used internally to perform the restore, is incorrect by the amount of offset.

This occurs for example when one time is during Daylight Savings Time and the other is not; in this case the results will be restored to a time that is incorrect by one hour.

Workaround:

The following is the version of the method containing the fix. File in as SystemUser and commit.

```
category: 'Backup and Restore'
method: Repository
timeToRestoreTo: aDateTime

"Sets the time at which restoreFromCurrentLogs , restoreFromArchiveLogs,
and restoreToEndOfLog: will stop.
```

```
The restore will stop at the first checkpoint which originally occurred at or
after aDateTime. If timeToRestoreTo: has not been used since
restoreFromBackup: completed, then restores will proceed to the end of the
specified transaction log(s).

An error is generated if aDateTime precedes the time of the last restored
checkpoint, as shown by restoreStatus. An error is generated if the receiver
is not in restore-from-log state.

Execution of restoreFromBackup: or commitRestore will cancel the effect of any
previous execution of timeToRestoreTo:.

If restore has stopped at a time specified by this method, then a subsequent
restore may be used to continue restoring past the time specified by the last
timeToRestoreTo:. Alternatively, timeToRestoreTo: can be used to specify
another point in time before continuing the restore.

This method requires the FileControl privilege."

| aTimeT |

"Convert offset from 1901 to 1970 using constant derived from
(DateTime newGmtWithYear: 1970 dayOfYear: 1 milliseconds: 0) asSecondsGmt"

aTimeT := aDateTime asSecondsGmt - 2177452800.

(aTimeT < 0 _or:[ aTimeT > 16r7fffffff]) ifTrue:[
aDateTime _error: #rtErrArgOutOfRange
].

^ self _restoreFrom: aTimeT opcode: 8
%
```

Bug 39521 - Incorrect reason for disabled account

Product: GemStone/S 64 Bit

Versions: 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.4, 2.2.3, 2.2.x, 2.1.4, 2.1.x, 2.0.x, 1.2.5, 1.x

Platform All

Fixed In: 2.4

After logins are disabled, the reason is recorded on the user's account, and further attempt to log in will fail. If user then attempts to login, after the configured number of failed login attempts, the original reason for the account disable is overwritten by a different reason, the failed login attempts.

Bug 39484 - Compiler error on QuadByteString

Product: GemStone/S 64 Bit

Versions: 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.4.3, 2.4.4, 2.4.3, 2.4.2, 2.4.1, 2.4, 2.3.1.6, 2.3.1, 2.3

Platform All

Fixed In: 3.0

Impact: Informational

Source code in the form of QuadByteStrings cannot be processed by the GemStone Smalltalk compiler, and will return an error.

Workaround:

Ensure that compiled code is String or DoubleByteString.

Bug 39478 - Commit just after login fails

Product: GemStone/S 64 Bit

Versions: 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.x, 2.2, 2.1.x, 2.1, 2.0.x, 1.2.5, 1.2.4, 1.2.x, 1.1.x

Platform All

Fixed In: 2.3

During the login process, a new session updates the account's last login time, write-locks the security data information. and commits. In scenarios with many logins occurring simultaneously, the write lock may fail or be dirty, or the commit may fail. The code did not handle these scenarios and several subsequent problems could occur:
- The first user commit after login by the new session may fail, since no abort was done following the failed commit.
- The write lock may remain on the security data information.
- User information, such as login count, may not get updated. This affects results such as number of logins permitted before a password change is required.

Bug 39476 - Hashed lookup fails on String/DoubleByteStrings containing ÿ

Product: GemStone/S 64 Bit

Versions: 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.4, 2.2.3, 2.2.2, 2.2.1, 2.2., 2.2, 2.1.4, 2.1.x, 1.2.5, 1.2.4, 1.2.x, 1.1.x, earlier versions

Platform All

Fixed In: 2.3

There is an error in the hashing algorithm for DoubleByteStrings containing the Character with ASCII value 255 - the character ÿ. As a result, a String containing ÿ and an otherwise equal DoubleByteString do not hash to the same value. This means that in hashed collections (such as Bag and KeyValueDictionary), elements that are DoubleByteStrings containing ÿ cannot be looked up in the collection using the equivalent String, and vice versa.

Indexes do not use hashing and are not affected.

Bug 39473 - findString:startingAt: incorrect for empty string argument

Product: GemStone/S 64 Bit

Versions: 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.4, 2.2.3, 2.2.2, 2.2.x, 2.1.4, 2.1.x, 2.0.4, 2.0.x, 1.2.5, 1.2.4, 1.1.14, 1.x

Platform All

Fixed In: 2.3.1

When CharacterCollection >> findString:startingAt: is called with a search argument that is an empty string '', it returns 1, rather than 0.

Other Smalltalk implementations of findString:startingAt:, and GemStone's method SequenableCollection >> indexOfSubCollection:startingAt: return 0 for this case.

Bug 39446 - copydbf -i on active tranlog hangs

Product: GemStone/S 64 Bit

Versions: 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5, 2.2.x, 2.2, 2.1.x, 2.1, 2.0.x, 2.0

Platform All

Fixed In: 2.3

copydbf -i or -I attempts to get a lock on the tranlog it is reading, which fails for the currently active tranlog. copydbf then incorrectly processes this as if the tranlog was compressed, and reads the tranlog using the gzip functions, which are very slow.

Bug 39426 - renameAssociationFrom:to: unsafe

Product: GemStone/S 64 Bit

Versions: 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.x, 2.2, 2.1.4, 2.1.x, 2.0.4, 2.0.x, 1.2.5, 1.2.4, 1.x

Platform All

Fixed In: 2.3

If you use SymbolDictionary renameAssociationFrom:to:, and specify a target name that already exists, it removes the association that is being renamed, as well as returning an error.

Bug 39359 - Risk of corruption from AIO page server write failure

Product: GemStone/S 64 Bit

Versions: 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.4, 2.2.3, 2.2.x, 2.1.4, 2.1.x, 2.0.x, 1.2.5, 1.x

Platform All

Fixed In: 2.3

Impact: Critical

If an AIO page server gets a write failure when writing to an extent during a checkpoint, the error may be ignored. This is related to the way write errors are handled, and the particular sequence of operations and retries between the stone and the AIO page server in this situation.

This case is extremely rare.

Bug 39355 - Change in character collation sequence can break indexes

Product: GemStone/S 64 Bit

Versions: 3.1, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.4.3, 2.4.4, 2.4.3, 2.4.2, 2.4.1, 2.4, 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.4, 2.2.3, 2.2.2, 2.2.1, 2.2

Platform All platforms

Fixed In: n/a

Impact: Critical

A change in the collation sequence of baseline extended ASCII characters (range 0-255) was inadvertently introduced in GS/64 2.2.0 with the introduction of Unicode character support. The change moved the collation sequence of characters with diacritic marks so that they would sort adjacent to their associated "plain vanilla" characters (originally they sorted towards the end of the sequence following their ASCII code).

The affected characters are:

192: LATIN CAPITAL LETTER A WITH GRAVE
193: LATIN CAPITAL LETTER A WITH ACUTE
194: LATIN CAPITAL LETTER A WITH CIRCUMFLEX
195: LATIN CAPITAL LETTER A WITH TILDE
196: LATIN CAPITAL LETTER A WITH DIAERESIS
197: LATIN CAPITAL LETTER A WITH RING ABOVE

170: FEMININE ORDINAL INDICATOR (treated like small a)
224: LATIN SMALL LETTER A WITH GRAVE
225: LATIN SMALL LETTER A WITH ACUTE
226: LATIN SMALL LETTER A WITH CIRCUMFLEX
227: LATIN SMALL LETTER A WITH TILDE
228: LATIN SMALL LETTER A WITH DIAERESIS
229: LATIN SMALL LETTER A WITH RING ABOVE

199: LATIN CAPITAL LETTER C WITH CEDILLA
231: LATIN SMALL LETTER C WITH CEDILLA

200: LATIN CAPITAL LETTER E WITH GRAVE
201: LATIN CAPITAL LETTER E WITH ACUTE
202: LATIN CAPITAL LETTER E WITH CIRCUMFLEX
203: LATIN CAPITAL LETTER E WITH DIAERESIS

232: LATIN SMALL LETTER E WITH GRAVE
233: LATIN SMALL LETTER E WITH ACUTE
234: LATIN SMALL LETTER E WITH CIRCUMFLEX
235: LATIN SMALL LETTER E WITH DIAERESIS

204: LATIN CAPITAL LETTER I WITH GRAVE
205: LATIN CAPITAL LETTER I WITH ACUTE
206: LATIN CAPITAL LETTER I WITH CIRCUMFLEX
207: LATIN CAPITAL LETTER I WITH DIAERESIS

236: LATIN SMALL LETTER I WITH GRAVE
237: LATIN SMALL LETTER I WITH ACUTE
238: LATIN SMALL LETTER I WITH CIRCUMFLEX
239: LATIN SMALL LETTER I WITH DIAERESIS

209: LATIN CAPITAL LETTER N WITH TILDE
241: LATIN SMALL LETTER N WITH TILDE

210: LATIN CAPITAL LETTER O WITH GRAVE

211: LATIN CAPITAL LETTER O WITH ACUTE
212: LATIN CAPITAL LETTER O WITH CIRCUMFLEX
213: LATIN CAPITAL LETTER O WITH TILDE
214: LATIN CAPITAL LETTER O WITH DIAERESIS

186: MASCULINE ORDINAL INDICATOR (treated like small o)
242: LATIN SMALL LETTER O WITH GRAVE
243: LATIN SMALL LETTER O WITH ACUTE
244: LATIN SMALL LETTER O WITH CIRCUMFLEX
245: LATIN SMALL LETTER O WITH TILDE
246: LATIN SMALL LETTER O WITH DIAERESIS

217: LATIN CAPITAL LETTER U WITH GRAVE
218: LATIN CAPITAL LETTER U WITH ACUTE
219: LATIN CAPITAL LETTER U WITH CIRCUMFLEX
220: LATIN CAPITAL LETTER U WITH DIAERESIS

249: LATIN SMALL LETTER U WITH GRAVE
250: LATIN SMALL LETTER U WITH ACUTE
251: LATIN SMALL LETTER U WITH CIRCUMFLEX
252: LATIN SMALL LETTER U WITH DIAERESIS

221: LATIN CAPITAL LETTER Y WITH ACUTE

253: LATIN SMALL LETTER Y WITH ACUTE
255: LATIN SMALL LETTER Y WITH DIAERESIS

Due to this change, repositories converted from a pre-2.2.0 system to a post-2.2.0 system will exhibit the following problems:

* Encoded integers representing ANY string generated using Array>>_insertEncodingForString:arraySize: (primitive 542) on the pre-2.2.0 system will not generate the correct decoded String when using Array>>_decodeKeyAt:decoding:into: (primitive 631) on the post-2.2.0 system.

* Indexed collections generated on the pre-2.2.0 system on strings which contain these characters will not return correct search results on the post-2.2.0 system.

Workaround:

There are two approaches to work around these problems:

1. It is possible to reconfigure the post-2.2.0 system to use the original collation sequence. For this option to work, it must be applied before any new elements are added to the indexed collections or new encoded values generated. The procedure is as follows:

    1. Acquire the appropriate passivate data file from GemStone Techical Support according to your system version:

       OldCollateCharTable22.dat - for GS/64 2.2.X
       OldCollateCharTable23.dat - for GS/64 2.3.X and later

2. Login to the system as SystemUser and execute the following:

```
Character activateCharTablesFromFile: 'OldCollateCharTable2X.dat'.
System commitTransaction.
```

All sessions that login subsequently will use the "new" original collation sequence.

2. Delete and reconstruct any affected indexes, recalculate any encoded values, using the new collation sequence.

Bug 39292 - Upgrade/conversion problems with Segments

Product: GemStone/S 64 Bit

Versions: 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.4, 2.2.3, 2.2.2, 2.2.1, 2.2

Platform All

Fixed In: 2.3

Impact: Critical

During the upgrade or conversion process from 1.x or 2.0.x to 2.2.x, Segments are set up, since Segments are not used in the originating repository but are supported in 2.2.x with new protocol, and a Segment 20 is now required.

However, when the originating repository was previously upgraded from a version that did support Segments, such as GemStone/S 6.x, there were a number of problems, including:
- Several extra Segments were created.
- When the originating repository Segment count plus number of created Segments added up to 20, an incorrect limit check resulted in an error.
- A duplicate Segment remained in the Repository.
- Segment groups and authorization were not correctly setup for all Segments in all cases.

Additional Segment issues problems existed in upgrading Repositories that originally started on much older versions of GemStone, such as 4.x.

Workaround:

If no errors occur during conversion, and you do not intend to use Segments for security, the Segment problems are not serious.

Contact GemStone Technical Support for Segment audit code and upgrade/conversion files with fixes.

Bug 39249 - Problems due to C++ compiler name mangling

Product: GemStone/S 64 Bit

Versions: 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.4.3, 2.4.4, 2.4.3, 2.4.2, 2.4.1, 2.4, 2.3.1, 2.3

Platform All

Fixed In: 3.0

If a C++ compiler other than the one GemStone is compiled with is used
to compile a GCI application, the C++ name mangling may be different and
result in undefined symbol errrors. (#39249)

Workaround:

Version 2.4 and later include the file $GEMSTONE/include/gcicpp.hc. Including
this file in your code will avoid the problem.

Bug 39227 - Error during SortedCollection>>resort can corrupt collection

Product: GemStone/S 64 Bit

Versions: 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.x, 2.2, 2.1.4, 2.1.x, 2.0.4, 2.0.x, 1.2.5, 1.2.4, 1.2.3, 1.2.x, 1.1.14, 1.x

Platform All

Fixed In: 2.3.1

If an error occurs during the resort of a SortedCollection, the collection may be corrupted.

Workaround:

If an error occurs, do not commit the transaction.

Bug 39098 - Operations may return PlusSignalingNaN or PlusQuietNaN

Product: GemStone/S 64 Bit

Versions: 3.0.1, 3.0, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.4.3, 2.4.4, 2.4.3, 2.4.2, 2.4.1, 2.4, 2.3.1.6, 2.3.1, 2.3

Platform HP

Fixed In: n/a

Impact: Informational

When running on HP-UX, mathematical operations that return NaNs may return
different flavors, depending on the specific OS version and chipset. For
example,
  (5.0 / 0) * 0 * 1 may return PlusSignalingNaN or PlusQuietNaN

Workaround:

These return values are provided by the operating system, outside of GemStone's
control. Ensure your application can handle the OS results, and verify
before moving to new OS systems or versions.

Bug 38928 - kill -USR1 fails to write stack to log for stone process

Product: GemStone/S 64 Bit

Versions: 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.4, 2.2.3, 2.2.2, 2.2.1, 2.2

Platform All

Fixed In: 2.3

Sending kill -USR1 <pid> to a GemStone process will normally write the C level and Smalltalk stack traces to the process log file. Other changes inadvertently disabled this feature for the stone's process in version 2.2 and later.

Workaround:

pstack <pid> or the equivalent will extract the stack trace from the stone's process.

Bug 38721 - AllUsers usersInGroup: errors

Product: GemStone/S 64 Bit

Versions: 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.4, 2.2.3, 2.2.2, 2.2.1, 2.2., 2.2, 2.1.4, 2.1.x, 1.2.5, 1.2.4, 1.2.x, 1.1.14, 1.x

Platform All

Fixed In: 2.3

The method UserProfileSet >> usersInGroup: invokes methods that are now disallowed, resulting in an error.

Workaround:

File in the following as SystemUser and commit:

```
category: 'Group Membership'
method: UserProfileSet
usersInGroup: aGroupString

"Returns all the elements of the receiver that are in the group represented by
aGroupString.  If the current session does not have the authorizations required
for this operation, raises an error."

| theGroup |
theGroup := AllGroups _validateGroupString: aGroupString .
^ self select: [:each | (each groups includesIdentical: theGroup)]
%
```

Bug 38633 - numArgs incorrectly returns 0 when selector has embedded $_

Product: GemStone/S 64 Bit

Versions: 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.x, 2.2, 2.1.x, 2.1, 2.0.x, 1.2.5, 1.2.4, 1.2.x, 1.1.x

Platform All

Fixed In: 2.3

If a method selector contains $_, other than as the first character of a keyword, the numArgs method returns 0 rather than the actual number of keywords.

Bug 38359 - Nested block variable resolution with to:do: may be incorrect

Product: GemStone/S 64 Bit

Versions: 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.4.3, 2.4.x, 2.3.1.6, 2.2.6, 2.2.5.4, 2.x, 1.x

Fixed In: 3.0

The resolution of variables in nested blocks is not correct in some cases involving to:do: optimizations.

For example, the following code returns #(nil, nil, nil, 4); the correct result is #(1 2 3 4).

```
 | a b |
 a := Array new: 4.
 b := Array new: 4.
 1 to: a size do: [:i |
        b at: i put: [10 timesRepeat: [a at: i put: i]]
 ].
 1 to: b size do: [:j | (b at: j) value].
 a
```

Bug 38279 - IndexManager autoCommit not initialized during conversion

Product: GemStone/S 64 Bit

Versions: 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.x, 2.2, 2.1.4, 2.1.x, 2.1

Platform All

Fixed In: 2.3

During conversion, IndexManager autoCommit is not initialized, which leads to errors when it is accessed.

Bug 38143 - Install on Linux could leave MANPATH incorrect

Product: GemStone/S 64 Bit

Versions: 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.4, 2.2.3, 2.2.2, 2.2.1, 2.2., 2.2, 2.1.4, 2.1.x, 1.2.5, 1.2.4, 1.2.x, 1.1.14, 1.1.x

Platform Linux

Fixed In: 2.3

On Linux, the system default man path is used when MANPATH is not set, or when MANPATH starts or ends with ':' or contains '::' (that is, contains an empty substring). GemStone modifies the MANPATH during installation, to add $GEMSTONE/doc. However, if there was previously no MANPATH, GemStone incorrectly sets a default MANPATH that does not include ':' or '::', resulting in the system not being able to find man information.

Workaround:

manually reset the MANPATH

Bug 37620 - GcGem login failures during conversion

Product: GemStone/S 64 Bit

Versions: 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.4.3, 2.4.4, 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.4, 2.2.3, 2.2.2, 2.2.1, 2.2

Platform All

Fixed In: 3.0

Impact: Informational

After the low level conversion process is complete and the stone has been started up, but before the image upgrade is performed, the system is not in a consistent state. Per the configured settings, the stone will continuouely attempt to start the Admin and Reclaim GcGems, but these will encounter login failures until the upgadeImage step of conversion is performed.

Workaround:

The login failures can safely be ignored; after image upgrade, logins will succeed.

Bug 37558 - Error on attempt to login as SymbolUser

Product: GemStone/S 64 Bit

Versions: 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5, 2.2.4, 2.2.3, 2.2.2, 2.2.1, 2.2

Platform All

Fixed In: n/a

Impact: Informational

In past releases, while it was not supported to log in explicity
as SymbolUser, it was possible to do so. This represents a
security risk, so this is no longer permitted. Only the SymbolGem
can log in as SymbolUser.

Workaround:

The only thing you may want to do as SymbolUser is examine
AllSymbols. You can do this by:
  ((AllUsers userWithId:'SymbolUser') resolveSymbol:#AllSymbols) value

Bug 37462 - postconv fails on very large strings in 1.1.x conversion

Product: GemStone/S 64 Bit

Versions: 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.4, 2.2.3, 2.2.2, 2.2.1, 2.2, 2.2, 2.1.x, 2.1

The postconv step of conversion may fail for originating 1.1.x
repositories with extremely large strings, with an error "VM temp
obj memory is full". The large object coversion rebuilds objects in
memory, with the risk of running out of memory for sufficiently
large objects.

The repository is safe to use even though the large object/s were
not converted; the old large object node layout will still work.

Workaround:

You can increase the amount of temporary object cache from the
default of 200000 by for the conversion process using the
postconv -t option. See the installation instructions for more information.

If this is not sufficient to allow postconv to complete, contact
GemStone Technical Support.

Bug 37082 - DateTime creation on boundaries of Daylight Savings Time

Product: GemStone/S 64 Bit

Versions: 3.6.3, 3.6.2, 3.6.1, 3.6, 3.5.7, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.x, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.x, 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.x, earlier versions

Platform All

Creation of local DateTimes at the start and end of DST are problematic.

Between the start date and time of DST and one hour later, there is no local time. Attempts to create a local DateTime in this non-existent range do not fail, they return the DateTime one hour earlier.

Between the end date and time of DST and one hour earlier, the local Time is ambiguous; there are two distinct DateTimes that are correct representations of the local time. The current creation mechanism returns the DateTime of the later of the two. Consequentially, there are certain GMT DateTimes (the earlier of the two for an ambiguous local time) that it is not possible to create using the DateTime local time creation methods

Workaround:

Use the DateTime gmt creation methods to avoid the inherent ambiguity of the local time at the DST boundaries.

Bug 37069 - SAG missing requirement to disable epoch during offline FDC

Product: GemStone/S 64 Bit

Versions: 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.x, 2.2, 2.1.4, 2.1.3, 2.1.1, 2.1

Platform All

Fixed In: n/a

The System Adminstration Guide provides information on running FDC on an offline copy of the production database. It omits to mention the requirement that epoch garbage collection must be disabled during this period on the production database, to avoid the FDC process returning objects that do not exist or have been reused. This should not cause corruption, but the extra processing will cause MGC performance to degrade to the point of uselessness.

Epoch GC is disabled by default.

Bug 36900 - Very large hashed dictionary out of memory on rebuildTable:

Product: GemStone/S 64 Bit

Versions: 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.3, 3.1.0.2, 3.1.0.1, 3.1, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.4.3, 2.4.4, 2.4.3, 2.4.2, 2.4.1, 2.4, 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.x, 2.2, 2.1.4, 2.1.x, 2.1, 2.0.4, 2.0.3, 2.0.x, 1.2.4, 1.2.x, 1.1.x, 1.x

Platform All

Impact: Informational

If a very large hashed dictionary (such as a KeyValueDictionary) grows larger, such that a rebuildTable: is required, it may not be possible to do the rebuildTable: without an out of memory error, even with maximum TOC, since the entire collection may not be able to fit in memory.

Workaround:

Reset the collisionLimit of the collection to a much larger value than the table size. The rebuildTable: optimizes performance by reducing the number of collisions in table lookup, it is not required for correct behavior.

rebuildTable: is called when at:put: is invoked and the collection has too many collisions, so for large collections, you should manually control collection size rather than allowing the automatic growth. Set the collection size larger than you expect to ever need it, and size the collisionLimit much larger than the collection size to avoid rebuild.

If you need a larger table size when the collection is larger than will
fit into memory, you will need to create a new, properly sized dictionary
and add the elements to it, with interim commits.

Bug 36892 - Create/remove indexes with uncommitted objects may result in errors

Product: GemStone/S 64 Bit

Versions: 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.x, 2.2, 2.1.4, 2.1.x, 2.1, 2.0.4, 2.0.3, 2.0.x

Platform All

Fixed In: 2.3

If indexes are created on uncommitted collections containing
uncommitted objects, queries on these collections will create
results that include OOPs for the objects. If the results are
saved, but the collection or any objects in the collection are
removed without committing or the commit fails, this may result in
incorrect answeres or object does not exist errors.

Workaround:

Ensure that queries on indexed collections only operate on committed
collections, and that no objects have been added to the collection
since the last commit in this session.

Bug 36769 - Remote caches do not use SHR_PAGE_CACHE_LOCKED setting

Product: GemStone/S 64 Bit

Versions: 3.6.3, 3.6.2, 3.6.1, 3.6, 3.5.7, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.x, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.x, 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.x, earlier versions

Platform All

A true setting for the config param SHR_PAGE_CACHE_LOCKED does
not get picked up by a shared page cache remote from the stone;
this setting is always false.

Workaround:

Edit the startshrpcmon script used on the remote machine to
hardcode this parameter; or more properly make a copy of the script
and modify services.dat to use your customized script.

Bug 36649 - shrinkExtents error

Product: GemStone/S 64 Bit

Versions: 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.6, 2.4.4.5, 2.4.4.4, 2.4.4.3, 2.4.4, 2.4.3, 2.4.2, 2.4.1, 2.4, 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.4, 2.2.3, 2.2.2, 2.2.1, 2.2

Platform All

Fixed In: n/a

The method Repository >> shrinkExtents may return an error

```
  ERROR, pagesNeedRemove = 1
```

This does not indicate a serious problem, and the operation should succeed if retried.

Note that we do not recommend running shrinkExtents unless necessary. Backup and restore is a more reliable way to make extent files smaller. shrinkExtents is deprecated in 3.0 and above.

Bug 36626 - Topaz command number parsing confused by locale

Product: GemStone/S 64 Bit

Versions: 3.6.3, 3.6.2, 3.6.1, 3.6, 3.5.7, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.x, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.x, 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.x

Platform All

The topaz parser does not correctly handle numbers with a non-period decimal point locale setting. This affects expressions such as

```
topaz 1> send 12.34 asString
```

This type of expression will not work in a non-period decimal point locale; in a non-period locale, neither period nor comma decimal points will parse correctly.

Note also that Smalltalk syntax specifies the period as the decimal separator so numbers entered in a "run" or "printit" code block must also use a period. Locale allows you to read or display Float values, using GemStone code, with the appropriate decimal point.

Bug 36441 - Character >> isVowel not correct for Characters with codePoints greater than 255

Product: GemStone/S 64 Bit

Versions: 3.6.3, 3.6.2, 3.6.1, 3.6, 3.5.7, 3.5.6, 3.5.5, 3.5.4, 3.5.3, 3.5.2, 3.5.1, 3.5, 3.4.5, 3.4.4, 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.x, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.4, 2.4.x, 2.3.1.6, 2.3.x, 2.2.5.4, 2.2.x, earlier versions

Impact: Informational

Character >> isVowel does not return correct results for Characters with codePoints greater than 255. "Vowel" is not a unicode or general property of a Character; the method applies specifically to English, and is retained to support legacy applications.

Bug 35604 - AIX 5.3 technology level 5 problem - PageManager shutdown

Product: GemStone/S 64 Bit

Versions: 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.x, 2.2, 2.1.4, 2.1.x, 2.1, 2.0.x, 2.0, 1.2.4, 1.2.x, 1.1.x

Platform AIX

Fixed In: not applicable

AIX 5.3, technology level 5 (5.3.005) has a bug that causes the PageManager to shut down, which causes the stone to shut down, approximately 12 hours and 15 minutes after startup. These are clean shutdowns due to socket closure.

This bug is fixed in AIX Service Pack 5300-05-03 - Technology level 5, service pack 3. The relevant APAR is "IY89429 AIX STOPS RESPONDING TO KEEPALIVES"

Workaround:

If you upgrade to AIX 5.3 technology level 5, be sure to use service pack 3 or later.

Bug 35227 - Gems with static user action code die with SIGILL

Product: GemStone/S 64 Bit

Versions: 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.x, 2.2, 2.1.4, 2.1.3, 2.1.2, 2.1.1, 2.1

Platform AIX

Fixed In: 2.3

Gem executables statically liked with user action code (as described in the GemBuilder for C manual, Appendix C) die with SIGILL, illegal instruction.

Workaround:

Use dynamically linked user action libraries, as described in the GemBuilder for C manual, Chapter 3.

Bug 35226 - Statically linked custom gems broken on Linux

Product: GemStone/S 64 Bit

Versions: 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.x, 2.2, 2.1.4, 2.1.3, 2.1.2, 2.1.1, 2.1

Platform Linux

Fixed In: 2.3

Statically linked custom gems do not work correctly on Linux and may encounter a SIGSEGV error when run. This may occur even though the compile and link steps report no error.

Workaround:

No workaround

Bug 35165 - Getting C-level stack trace can hang on AIX

Product: GemStone/S 64 Bit

Versions: 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.x, 2.2, 2.1.4, 2.1.x, 2.1, 2.0.x, 2.0, 1.2.4, 1.2.x, 1.1.x

Platform AIX

Fixed In: not applicable

GemStone/S 64 Bit allows printing of the C-level stack trace on AIX, using the procstack function. The C-level stack trace is printed when sending -SIGUSR1 to a GemStone process, or when printing stack traces to the log due to low memory conditions, or during error conditions.

AIX version 5.3 has a bug that causes the procstack function to hang. Customers on AIX version 5.3 should download and install APAR IY83603.

This APAR is included in AIX 5.3 Technology Level 5, however, 5.3.005 has a separate problem, and should not be used.

Workaround:

Install the APAR. Avoiding sending SIGUSR1 or other conditions that will cause a trace to be printing may help but are unlikely to be sufficient.

Bug 34268 - Assigning object to extent may not succeed

Product: GemStone/S 64 Bit

Versions: 2.2.5.4, 2.2.5.3, 2.2.5.2, 2.2.5.1, 2.2.5, 2.2.x, 2.2, 2.1.4, 2.1.3, 2.1.x, 2.0.5, 2.0.4, 2.0.x, 2.0

Platform All

Fixed In: n/a

Impact: Informational

Earlier products and versions supported using ClusterBuckets to allow assigning an object to a particular extent, using ClusterBucket>>extentId: or ClusterBucket>>newForExtent:. These methods are deprecated. Clustering will attempt to use the specified extent, but is not guarenteed.

Bug 33429 - Topaz with both linked and rpc sessions may encounter error handling problems

Product: GemStone/S 64 Bit

Versions: 3.4.3, 3.4.2, 3.4.1, 3.4, 3.3.9, 3.3.8, 3.3.7, 3.3.6, 3.3.5, 3.3.4, 3.3.3, 3.3.1, 3.3, 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.x, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.x, 2.3.1.6, 2.3.1, 2.3, 2.2.5.4, 2.2.x, earlier versions

Platform All

If there are both linked and RPC sessions active in the same linked topaz (topaz -l), and the RPC session encounters an error, it is possible for the GCI error handling to become confused about which session had the error and report it on the linked session.

GBJ

Bug 49414 - GciTransportError on GBJ methods executing server code calling user actions

Product: GemBuilder for Java

Versions: 3.1.3

Platform All Platforms

If an application calls a GBJ method which executes server-side smalltalk code that in turn calls a user action, a GciTransportError will be triggered similar to:

```
Exception in thread "main" com.gemstone.gbj.GbjRuntimeException: a GciTransportError occurred (error 2728), client user action not supported during non-blocking GCI call actionNameOop 5264129
    at com.gemstone.gbj.GbjException.setup(GbjException.java:150)
    at com.gemstone.gbj.GbjSession.handleExecutionException(GbjSession.java:487)
    at com.gemstone.gbj.GbjSession.perform(GbjSession.java:1435)
    at com.gemstone.gbj.GbjObject.perform(GbjObject.java:476)
    at com.gemstone.gbj.GbjObject.sendMsg(GbjObject.java:526)
```

Examination of the GCI call actionNameOop will indicate the name of the user action being called.

Affected GBJ methods include:

GbsSession:
  execute
  doit
  printit

GbjObject
  execute
  perform
  sendMsg

Bug 48138 - Printout of oop in GbjObject toString goes negative

Product: GemBuilder for Java

Versions: 3.1.3, 3.1.2, 3.1.1, 3.1, 3.0.1, 3.0

Platform All Platforms

The printout of oops in the GbjObject asString( ) method can wrap-around to negative values. For example, the asString( ) printouts for the minimum and maximum SmallInteger values look like:

```
  MIN: GbjObject(oop:-9223372036854775806/-1152921504606846976/sid:1)
  MAX: GbjObject(oop:9223372036854775802/1152921504606846975/sid:1)
```

The oop for MIN should be printed as: 9223372036854775810.

Bug 47289 - GBJ Error: startindex/numObjs out of range in getObjs() using GbjIterator

Product: GemBuilder for Java

Versions: 3.1.3, 3.1.2, 3.1.1, 3.1, 3.0.1, 3.0

Platform All Platforms

When using an instance of GbjIterator to enumerate over a large collection, if the collection is larger than GbjIterator.BatchSize (defaults to 1000), then the following exception is raised when attempting to enumerate to an element beyond the batch size:

```
com.gemstone.gbjgci.GbjGciException: GBJ Error: startIndex/numObjs out of range in getObjs()
    at com.gemstone.gbjgci.GbjGciInterface.newException(GbjGciInterface.java:324)
    at com.gemstone.gbjgci.GbjGciInterface.getObjs(GbjGciInterface.java:888)
    at com.gemstone.gbjgci.GbjGciSession.getObjs(GbjGciSession.java:486)
    at com.gemstone.gbj.GbjIterator.getNextBatch(GbjIterator.java:112)
    at com.gemstone.gbj.GbjIterator.nextElement(GbjIterator.java:102)
    at Test15.doit(Test15.java:65)
    at Test15.main(Test15.java:80)
```

Workaround:

Increase the size of GbjIterator.BatchSize to a value greater than the largest expected collection size. This value should not exceed the setting for bufSize used in a GbjSession.initialize( ) or GbjGciInterface.initialize( ) call (see javadocs for details - default value is 100000).

Alternatively, use a java.utils.Enumeration instead.

Bug 47084 - Enumeration over large collection of registered class instances can fail with ClassCastException

Product: GemBuilder for Java

Versions: 3.1.3, 3.1.2, 3.1.1, 3.1, 3.0.1, 3.0

Platform All Platforms

IF:

1. Using registerStub to establish a correspondence between a java and GS Smalltalk class
2. Enumerating over a collection of instances of this class replicated from the server
3. The size of the preload buffer (set in GbjSession.initialize( )) is smaller than the number of objects in the replicated collection

THEN:

The retrieval of elements that were not preloaded will fail with a java.lang.ClassCastException.

Example:

```
  GbjSession s;
  // ( code to login session s here )
  s.registerStub(new TestObject(), "TestObject");
  GbjCollection c = (GbjCollection)s.doit( <code to return large collection of TestObjects> );
  for (Enumeration e = c.elements(); e.hasMoreElements(); ) {
   TestObject x = (TestObject)e.nextElement(); << will fail here
  }
```

Workaround:

If feasible, increase the size of the preload buffer to accomodate the full collection.

If not practical, refactor code to retrieve a subsection of the full collection in batches that are small enough to fit in the preload buffer.

Bug 46551 - GbjObject.stringValue( ) on really big numbers is wrong

Product: GemBuilder for Java

Versions: 3.1.3

Platform All Platforms

Using the GbjObject.stringValue( ) method will return incorrect results on objects that are LargeIntegers on GS 64-bit or LargePositiveIntegers/LargeNegativeIntegers on GS 32-bit.

Workaround:

Use GbjObject.asString( ) instead.

Bug 46531 - Problems using GbjLauncher programatically

Product: GemBuilder for Java

Versions: 3.1.3, 3.1.2, 3.1.1, 3.1, 3.0.1, 3.0

Platform All Platforms

Calling up a GbjLauncher programatically via java code "new GbjLauncher(aGbjSession)" will bring up the GbjLauncher window, but then performing almost any menu or command from this GbjLauncher will cause the window to hang and a java stack similar to the following to be generated in stderr:

```
Exception in thread "AWT-EventQueue-0" java.lang.ClassCastException: com.gemstone.gbj.GbjObject cannot be cast to com.gemstone.tools.GbxToolsSupportImpl
    at com.gemstone.tools.GbxToolsSupportImpl.supportObjectForSession_(GbxToolsSupportImpl.java:43)
    at com.gemstone.tools.GbxTool.serverSupport(GbxTool.java:724)
[Subsequent stack depends on the particular command/menu option selected]
```

Workaround:

Call the GbjLauncher directly from the OS:

```
unix>  java com.gemstone.tools.GbjLauncher
```

Bug 46522 - Windows RegCreateKeyEx warning

Product: GemBuilder for Java

Versions: 3.1.3

Platform Windows Only

Impact: Informational

When running GBJ on a Windows client you may observe the warning message:

```
WARNING: Could not open/create prefs root node Software\JavaSoft\Prefs at root 0x90000002. Windows RegCreateKeyEx(...) returned error code 5.
```

This indicates that the user does not have Administrator privileges needed to create the java Prefs registry used by the GbjGciPreferences logging utility.

To resolve this, run GBJ once as Administrator. This will create the Prefs registry so that all subsequent GBJ will not complain.

Bug 46513 - GBJ library not delivered with GS/32 servers

Product: GemBuilder for Java

Versions: 3.1.3

Platform All Platforms

Impact: Informational

The GbjGci library is not included as part of the server release for the 32-bit product GS/S 6.7 and later releases. Customers using GBJ on these servers should contact GemTalk Technical Support for the correct library.

Bug 46507 - GBJ library load problems on Windows

Product: GemBuilder for Java

Versions: 3.1.3

Platform Windows Only

Due to the way GS/64 constructs paths, there are library load problems when running GBJ on Windows when using a GS/64 server. These problems cause a fatal runtime exception in the GBJ Tools.

The specific error will depend on if you are using a 32-bit or 64-bit java:

On Windows 64-bit java/GBJ runs, will fail with an error similar to: DynLibLoad() failed. error: LoadLibrary(<GSDIRECTORY>/lib/libgcirpc-<GSVERSION>-64.dll) failed with errcode 126

On Windows 32-bit java/GBJ runs, will fail with an error similar to: DynLibLoad() failed. error: LoadLibrary(<GSDIRECTORY>\bin\libgcirpc-<GSVERSION>-32.dll) failed with errcode 126

Workaround:

For 64-bit java: copy the GCI library, libgcirpc-N.N.N-64.dll, from the %GEMSTONE%/bin directory to %GEMSTONE%/lib directory. Only the specific file libgcirpc-N.N.N-64.dll needs to be copied.

For 32-bit java: copy the GCI library, libgcirpc-N.N.N-32.dll, from the %GEMSTONE%/bin32 directory to %GEMSTONE%/lib32 directory. Only the specific file libgcirpc-N.N.N-32.dll is needed.

Bug 46468 - GbjDebugger / client forwarder problems

Product: GemBuilder for Java

Versions: 3.1.2

Platform All Platforms

Fixed In: 3.1.3

Use of client forwarders or the GbjDebugger tool can result in the following stack:

```
  Exception in thread "AWT-EventQueue-0" com.gemstone.gbjgci.GbjGciException: GBJ  Error: a InternalError occurred
(error 2101), The object with object ID XXXXXXXX  does not exist.
    at com.gemstone.gbjgci.GbjGciException.setup(GbjGciException.java:76)
    at com.gemstone.gbjgci.GbjGciInterface.serverContinue(GbjGciInterface.java:1277)
    at com.gemstone.tools.GbxToolsSupportImpl.continueProcess_(GbxToolsSupportImpl.java:922)
```

This issue required server side fixes, done on the following versions:

6.7.1
3.2.16
3.3.2
3.4

Bug 46368 - GbjGciPreferences breaks on libgbjgci library load problems

Product: GemBuilder for Java

Versions: 3.1.2, 3.1.1

Platform All Platforms

Fixed In: 3.1.3

GbjGciPreferences commands will attempt to load the $GEMSTONE/lib/libgbjgci* library. If there are library load problems, this will cause the GbjGciPreferences command to fail.

Workaround:

Review the java stack trace generated and resolve the underlying libgbjgci* library load issue.

Bug 46359 - Using a 64-bit Java with GBJ on 32-bit GS/S fails with UnsatisfiedLinkError

Product: GemBuilder for Java

Versions: 3.1.3

Platform All Platforms

The GbjGci Library built for GS/S is 32-bit. Attempting to run GBJ using a 64-bit Java will fail with the error:

java.lang.UnsatisfiedLinkError: $GEMSTONE/lib/libgbjgci313.so: ld.so.1: java: fatal: $GEMSTONE/lib/libgbjgci313.so: wrong ELF class: ELFCLASS32 (Possible cause: architecture word width mismatch)

Oracle no longer provides a 32-bit Java for Solaris as of version 1.8. Because of this, GBJ will not run on 32-bit GS/S on Solaris (but see workaround).

Workaround:

Use a 32-bit Java executable.

If you must run Java on Solaris (either x86 or SPARC), contact GemTalk Technical Support, which can generate a special version of GBJ using the older 1.7 version of the Java SDK. You will need to run 32-bit Java version 1.7, which *is* available.

Bug 46281 - GBJ Diagnostic logging doesn't work

Product: GemBuilder for Java

Versions: 3.0

Platform All Platforms

Fixed In: 3.1

GBJ 3.0 introduced a new feature to allow enabling of diagnostic debugging messages using the new GbjGciPreferences class. To see details of this, execute:

> java com.gemstone.gbjgci.GbjGciPreferences help

This feature does not work properly in GBJ 3.0 but is fixed in 3.1 and later.

Workaround:

Upgrade to 3.1 or later.

Bug 46200 - GbjBroker gslist status "killed" if prior broker dies and is restarted

Product: GemBuilder for Java

Versions: 2.3, 2.2, 2.1.2.1, 2.1.2, 2.1.1, 2.1, 2.0

Platform All Platforms

Fixed In: 3.0

If the GbjBroker process is killed or dies suddenly, gslist with options -l -v or -x will still continue to report it with a status of "killed" unless a gslist -c is performed to clear it.

If a new GbjBroker is started without doing a gslist -c, gslist will still continue to show a status of "killed" even though the new GbjBroker is working correctly.

Workaround:

Remember to "gslist -c" before starting a new GbjBroker.

Alternatively, file-in as SystemUser and commit the following code:

```
! Bugfix for 46200
!
! Add code to unpublish earlier published data in case prior session died
!
category: 'Private'
method: GsbServerSocket
publishOnGsList

    self shouldPublishOnGsList ifTrue:
        [
         (System _publish: self name
                    port: self port
                     log: self gslistLogName asString
                 options: self gslistOptionsString asString)
          ifFalse: [
              "Problem publishing new data -- unpublish old info"
              self unpublishOnGsList.
              System _publish: self name
                         port: self port
                          log: self gslistLogName asString
                      options: self gslistOptionsString asString ]]
%
```

Bug 45898 - GBJ 3.1 library on GS/64 2.4.7 / AIX doesn't load

Product: GemBuilder for Java

Versions: 3.1

Platform AIX only

The GBJ 3.1 library on GS/64 2.4.7 running on AIX will not load properly, failing with errors similar to:

```
[ FINE ] Loading GbjGciInterface library java.lang.UnsatisfiedLinkError: gbjgci31 (rtld: 0712-001 Symbol GcsiInit was referenced
    from module $GEMSTONE/lib/libgbjgci31.so(), but a runtime definition
    of the symbol was not found.
```

Workaround:

Contact GemTalk Technical Support for a replacement libgbjgci31.so library.

Bug 45749 - ClassCastException when using client forwarders

Product: GemBuilder for Java

Versions: 3.1.1, 3.1, 3.0.1, 3.0

Platform All Platforms

Fixed In: 3.1.2

Trying to use a client forwarder in GS smalltalk code to a java method on the client will currently trigger a java.lang.ClassCastException.

Bug 45748 - GbjSession>>perform can fail with #rtErrNoExistingSymbol

Product: GemBuilder for Java

Versions: 3.1.3, 3.1.2, 3.1.1, 3.1, 3.0.1, 3.0

Platform All Platforms

If GbjSession>>perform is called using a method name that has not already been canonicalized as a symbol on the server, then the operation will fail with a GbjRuntimeException encapsulating a #rtErrNoExistingSymbol error (2404) instead of the expected #rtErrDoesNotUnderstand (2010).

This also applies to the underlying GbjGciSession>>perform call.

Workaround:

Fix java code to call a method that *does* exist on the server.

Bug 45354 - GbjBroker can get object does not exist errors during gem spawning

Product: GemBuilder for Java

Versions: 2.3, 2.2, 2.1.2, 2.1.1, 2.1, 2.0

Platform All Platforms

Fixed In: 3.0

Under rare conditions the GbjBroker may record an object does not exist error in its log and thereafter be unable to spawn additional gems for the gem pool. The stack in the broker log will look like:

```
The object with object ID XXXXXXX does not exist.

1 ComplexVCBlock in GsbFPLFramework >> defaultEventErrorHandler @ 73 [GsMethod OOP 343397585] 2
```

ComplexBlock in GsbFPLFramework >> handleEvent: @ 18 [GsMethod OOP 98351469] 3 Object >> _gsReturnTos @ 1 [GsMethod OOP 1004741] 4 GciInterface >> nbLogin: @ 2 [GsMethod OOP 2674473] 5 GsbSCGemManager >> spawnGem: @ 6 [GsMethod OOP 245249] 6 GsbGemManager >> spawnGems:spacing: @ 13 [GsMethod OOP 240969] 7 GsbGemManager >> spawnGems: @ 8 [GsMethod OOP 240477] 8 GsbFPLClientSocket >> spawnGems: @ 4 [GsMethod OOP 246189] 9 GsbFPLMessage >> addGems: @ 6 [GsMethod OOP 5167333] 10 GsbFPLMessage >> dispatch @ 26 [GsMethod OOP 5167393] 11 GsbFPLtcmsMessage >> dispatch @ 187 [GsMethod OOP 710449] 12 GsbFPLMessage >> dispatchMarshalled @ 6 [GsMethod OOP 300353] 13 GsbFPLClientSocket >> processString: @ 7 [GsMethod OOP 244721] 14 GsbFPLClientSocket >> process @ 37 [GsMethod OOP 242481] 15 GsbClientSocket >> handleConnectedEvent @ 2 [GsMethod OOP 248293] 16 GsbClientSocket >> handleEvent @ 4 [GsMethod OOP 248105] 17 GsbFPLFramework >> handleEvent: @ 28 [GsMethod OOP 98351469] 18 GsbFPLFramework >> eventLoop: @ 22 [GsMethod OOP 234533] 19 GsbFPLFramework >> start:configKey: @ 38 [GsMethod OOP 233713] 20 GsbFPLFramework >> startBroker: @ 2 [GsMethod OOP 233649] 21 GsbFPLFramework class >> startBroker: @ 3 [GsMethod OOP 238761]

Workaround:

Shutdown and restart the GbjBroker.

Bug 44659 - Setting GbjSession.montiorSessionSignals to true will trigger "GBJ classes not loaded" error

Product: GemBuilder for Java

Versions: 3.1.1, 3.1

Platform All Platforms

Fixed In: 3.1.2

Gem-to-gem and object change notification are activated for a GbjSession by setting the boolean flag monitorSessionSignals to true. But currently, doing so triggers the following spurious error during session login:

```
SEVERE: GBJ classes not loaded in GS/S Server java.lang.NullPointerException
    at com.gemstone.gbj.GbjSession.testForSignals(GbjSession.java:619)
    at com.gemstone.gbj.GbjSession.execute(GbjSession.java:1246)
    at com.gemstone.gbj.GbjSession.execute(GbjSession.java:123)
    at com.gemstone.gbjgci.GbjGciSession.login(GbjGciSession.java:348)
    at com.gemstone.gbj.GbjSession.connect(GbjSession.java:942)
```

Bug 44646 - Using monitorChangedObjects can cause GCI 2203 errors

Product: GemBuilder for Java

Versions: 3.1.1, 3.1, 3.0.1, 3.0

Platform All Platforms

Fixed In: 3.1.2

When activating object change monitoring via a GbjSession's monitorChangedObjects flag, if the GbjSession connect() call is immediately followed by another method that accesses the server (such as execute), there is a possibility that either the primary or GbjPoller threads will trigger a GbjException on GemStone error 2203 - "GCI operation requested while a nonblocking or thread safe call in progress for the session".

Workaround:

Follow the GbjSession connect( ) call with Thread.sleep(X) to allow time for the GbjPoller thread to complete its setup. Experiment with values for X (in milliseconds) to determine an optimal value.

Bug 44618 - Problem with GbjSession registerStub( )

Product: GemBuilder for Java

Versions: 3.1.1, 3.1, 3.0.1, 3.0

Platform All Platforms

Fixed In: 3.1.2

Using the GbjSession method registerStub( ) will raise the GbjGciException: "GCI Interface Error/Object table put on oop zero".

Bug 44577 - Android apps calling getObjs( ) can trigger JNI error

Product: GemBuilder for Java

Versions: 3.1.1

Platform Android

Fixed In: 3.1.2

Android applications calling function getObjs( ) that retrieve more than 512 objects can exceed the capacity of the JNI internal local reference table. The resulting error printout looks like this:

```
  JNI ERROR (app bug): local reference table overflow (max=512)
 JNI local reference table (0x2a228720) dump:
 Last 10 entries (of 512):
  [ list of the last 10 entries in the table ]
 Summary:
  [ summary of the class types of the 512 entries in the table ]
 Failed adding to JNI local ref table (has 512 entries)
  "main" prio=5 tid=1 RUNNABLE
```

Workaround:

Break getObjs( ) calls into chunks that retrieve no more than 512 entries at a time.

Bug 44576 - Android demo application GbjTest does a force close on orientation changes

Product: GemBuilder for Java

Versions: 3.1.1

Platform Android

Fixed In: 3.1.2

The supplied Android demo application GbjTest does a force close when the orientation of the phone changes from portrait to landscape.

If the android log file is being monitored, the call stack looks like:

```
java.lang.RuntimeException: Unable to start activity
ComponentInfo{com.gemstone.android.gbjtest/com.gemstone.android.gbjtest.GbjTestActivity}:
com.gemstone.gbj.GbjException: GbjSession already initialized
  at android.app.ActivityThread.performLaunchActivity(ActivityThread.java:2211)
  at android.app.ActivityThread.handleLaunchActivity(ActivityThread.java:2261)
  at android.app.ActivityThread.handleRelaunchActivity(ActivityThread.java:3740)
  at android.app.ActivityThread.access$700(ActivityThread.java:141)
  at android.app.ActivityThread$H.handleMessage(ActivityThread.java:1262)
  at android.os.Handler.dispatchMessage(Handler.java:99)
  at android.os.Looper.loop(Looper.java:137)
  at android.app.ActivityThread.main(ActivityThread.java:5103)
  at java.lang.reflect.Method.invokeNative(Native Method)
  at java.lang.reflect.Method.invoke(Method.java:525)
  at com.android.internal.os.ZygoteInit$MethodAndArgsCaller.run(ZygoteInit.java:737)
  at com.android.internal.os.ZygoteInit.main(ZygoteInit.java:553)
  at dalvik.system.NativeStart.main(Native Method) Caused by: com.gemstone.gbj.GbjException: GbjSession already initialized
  at com.gemstone.gbj.GbjSession.initialize(GbjSession.java:793)
  at com.gemstone.android.gbjtest.GSInterface.initialize(GSInterface.java:15)
  at com.gemstone.android.gbjtest.GbjTestActivity.onCreate(GbjTestActivity.java:112)
  at android.app.Activity.performCreate(Activity.java:5133)
  at android.app.Instrumentation.callActivityOnCreate(Instrumentation.java:1087)
  at android.app.ActivityThread.performLaunchActivity(ActivityThread.java:2175)
```

Workaround:

Avoid turning the phone around while running the GbjTest app to avoid resetting the orientation from portrait to landscape.

Bug 44547 - GBJ/Android limited to Android 4.2 Jellybean (API level 17) and later

Product: GemBuilder for Java

Versions: 3.1.1

Platform Android

Fixed In: 3.1.2

Versions of Android prior to 4.2 Jellybean use a version of SSL that is incompatible with the SSL version used in GemStone. Attempts to run a GBJ/Android app on a phone/tablet with a version of Android earlier than 4.2 will fail when the app attempts to login and load the SSL library. The error generated is:

Error during login:, ssl library initialization failed

Workaround:

Upgrade the device to Android 4.2 or later.

Bug 44526 - Host specification on Android must be IP addresses

Product: GemBuilder for Java

Versions: 3.1.3, 3.1.2, 3.1.1

When specifying the host in GBJ/Android, you must use the IP address; host name lookup is not currently possible.

Bug 44523 - Android demo project GbjTest doesn't load into Eclipse/ADT

Product: GemBuilder for Java

Versions: 3.1.1

Platform Android

Fixed In: 3.1.2

Due to changes made in the directory layout of Android projects in Eclipse/ADT, the Android demo project GbjTest will not build properly. This appears to impact Eclipse/ADT versions released later than 3/21/2014.

Workaround:

Revert back to the 3/21/2014 version of the Eclipse/ADT bundle.

Bug 44503 - GBJ 2.3 does not work on GS/64 3.0 and later

Product: GemBuilder for Java

Versions: 2.3, 2.2, 2.1.2, 2.1, 2.0

Platform All Platforms

Fixed In: 3.0

Due to design changes in GS/64 3.0, GBJ 2.3 and earlier versions will not work properly (the GbjBroker process will crash).

Workaround:

Upgrade to a newer version of GBJ.

Bug 44000 - Linked logins on GBJ

Product: GemBuilder for Java

Versions: 3.1.3, 3.1.2, 3.1.1

Platform All Platforms

Impact: Informational

Due to library load restrictions on AIX, you cannot use linked sessions on that platform.

On all other platforms, you can generate a linked session by doing the following:

1. Call GbjSession.initialize( ) with true as the first argument.
2. Set the gemnetName of the associated GbjParameter object to "gcilnkobj".

For an example see the following code section.

Workaround:

```
// Example code for generating a linked session
//
    GbjSession.initialize(true, "MyGBJApplication", 10000, 10000, 10);
    GbjParameters p = new GbjParameters();
    p.serverName = "MyStone";
    p.userName = "DataCurator";
    p.password = "swordfish";
    p.gemnetName = "gcilnkobj";
    GbjSession s = new GbjSession(p);
    s.connect();
```

Bug 43387 - GBJ can't find GCSI library on GS/64 versions 3.1 and later

Product: GemBuilder for Java

Versions: 3.1

Platform All Platforms

Fixed In: 3.1.1

When using GBJ's cache stat functionality, the $GEMSTONE/lib/libgcsi.so library is loaded to provide low-level support. Starting with GS/64 3.1, a new naming convention was developed for GemStone libraries that included the version number and compiled version type (32 or 64). Because of the name change, GBJ can no longer find the file and will trigger a file not found error when attempting to load it.

Workaround:

In the $GEMSTONE/lib directory, create a symbolic link to the libgci-<version>-64.so file called libgcsi.so.

Bug 41956 - GBJ can miss static exception handlers, especially SigAbort

Product: GemBuilder for Java

Versions: 2.3, 2.2, 2.1.2.1, 2.1.2, 2.1.1, 2.1, 2.0

Platform All platforms

GBJ sets up several dynamic exception handlers to trap any error that might occur during event processing, particularly for problems that might occur during message transmission between the java client and the gem server.

Static exception handlers defined in your application will not be triggered if their associated exception occurs while GBJ's dynamic handlers are in effect.

In addition, if your static exception handler is for #rtErrSignalAbort (error 6009) and sigAbort handling is enabled using System>>enableSignaledAbortError, the triggering of this error while GBJ dynamic exception handlers are in effect can cause spurious errors when the flow of control during message processing is disrupted by the return from the dynamic exception handler. These errors include
"Out of Sequence" errors on the java client or references to a non-existent key in aGbjTransportBuffer in gem smalltalk code.

Workaround:

The following code resolves this problem on GBJ 2.3 for sigAbort errors. Customers using earlier versions should check that the modified methods are consistent with their version. If your problem involves a different error, replace "6009" with the appropriate error number.

```
! Fix for sigAbort variation of bug 41956
!
! Note: can be extended to cover other exceptions as needed..
!
! This code is based off of GBJ 2.3
!
! File-in as SystemUser and commit
!
category: 'Event Handling'
method: GsbFramework
handleEvent: anEvent

  Exception category: nil number: nil do:
    [ :ex :cat :num :args | | response |
   "fix 41956: resignal sigAborts"
   (num == 6009)
    ifTrue: [
       ex resignal: cat number: num args: args ]
    ifFalse: [

       response := self eventErrorHandler
          value: ex
          value: cat
          value: num
          value: args.
       response == nil ifFalse:
          [^response]].
   ]. "fix 41956"

   (myConfiguration at: #verbose) ifTrue: [
     self log: #GsbEvent object: anEvent].
   self computeEventDateTime.
   anEvent handleEvent.
   self heartbeat
%
category: 'Startup'
method: GbjServer
startup3
   "Run the SMF until the Java client quits."

   | pm reporting |
   Exception category: nil number: nil do: [ :ex :cat :num :args |

   "fix 41956: resignal sigAborts"
   (num == 6009)
    ifTrue: [
       ex resignal: cat number: num args: args ]
    ifFalse: [

      (GbjError isEventError: cat number: num)
      ifTrue: [    "GemStone event error"
         self event: (GbjError category: cat number: num args: args)
]
      ifFalse: [
         self logError: (GbjError category: cat number: num args: args).
         self close.
         ^0 ].
      nil ].

    ]. "fix 41956"

   Exception category: GbjSignals number: nil do: [ :ex :cat :num :args
|
      ex remove.
      ((num = clientClosedConnection) _or: [num = clientTimedOut])
      ifTrue: [
         self log: (GbjError category: cat number: num args: args) messageText.
         self close.
```

```
            ^0 ].
        ex resignal: cat number: num args: args.
        nil ].
    self profileServer ifTrue: [
        self log: 'Server profile requested by client'.
        pm := ProfMonitor new.
        pm startMonitoring.
        ].
    self framework eventLoop.
    (pm ~~ nil) ifTrue: [
        reporting := true.
        pm stopMonitoring; gatherResults; removeFile.
        self log: (pm reportDownTo: 1).
        pm removeResults.
        ].
    self close.
    ^0
%
```

Bug 41890 - Problems loading GBJ library on AIX

Product: GemBuilder for Java

Versions: 3.1.1, 3.1, 3.0.2, 3.0.1, 3.0

Platform AIX

Impact: Critical

Due to limitations in how AIX handles the loading of shared libraries, coupled with the need to load the libgbjgci library through Java, there can be problems loading this library, depending on the version of GBJ, GS/64, and the Java JDK that you are using.

Workaround:

If you are having difficulties, please contact Gemtalk Technical support, with the versions of GBJ, GS/64 and JDK that you are using, along with the Java stack trace.

Bug 41874 - GBJ library loading problems on Darwin

Product: GemBuilder for Java

Versions: 3.1.1, 3.1, 3.0.1, 3.0

Platform Darwin

Fixed In: 3.1.2

Impact: Critical

Attempts to run GBJ 3.0 and later on Darwin on versions of the GS/64 server prior to 3.2 will fail with the warning message: "Problem loading GbjGciLibrary".

Workaround:

For GBJ version 3.1 and 3.1.1 on GS/64 versions 3.2 and later, the GbjGciLibrary will load correctly, but the user must include $GEMSTONE/lib as part of the java.library.path supplied during the java call. For example:

```
unix>  java -d64 -Djava.library.path=$GEMSTONE/lib  MyGbjProgram
```

This is not required with 3.1.2 and later, which use Java v8.

Bug 41796 - Excessive blocking between GBJ sessions

Product: GemBuilder for Java

Versions: 3.0.1, 3.0

Platform All platforms

Fixed In: 3.1

In GBJ 3.0, if a java client is running multiple sessions, a long operation in one session (for example, executing a time-consuming GS smalltalk command) will cause operations in other sessions to block until the first one completes.

Workaround:

No workaround.

Bug 41786 - GBJ state information conflicts with GemEnterprise

Product: GemBuilder for Java

Versions: 3.0.1, 3.0

Platform All platforms

Fixed In: 3.1

GBJ caches state information in one of the System sessionState fields. In 3.0 the field used was changed from 19 to 8, which is the field used by GemEnterprise.

Customers still using GemEnterprise should modify the method GbjGciInterface>>sessionStateIndex to use a different field, as documented in the comment field for that method.

Workaround:

No workaround.

Bug 41757 - isLoggedIn() can incorrectly report true

Product: GemBuilder for Java

Versions: 3.0.1, 3.0

Platform All platforms

Fixed In: 3.1

The GbjGciSession method isLoggedIn() can sometimes incorrectly report true if the gem session has been externally terminated or killed and the GBJ client has not recently attempted to communicate with it.

Workaround:

Execute a no-op smalltalk method on the gem to confirm if still connected.

Bug 41755 - Asynchronous signals delayed on quiet GBJ client

Product: GemBuilder for Java

Versions: 3.0.1, 3.0

Platform All platforms

Fixed In: 3.1

Impact: Critical

With the 3.0 GbjGciInterface redesign, asynchronous signals are currently only signaled when GS smalltalk code is being executed or when the GBJ client communicates with the gem. This can cause delays in the receipt and/or processing of these signals if there is no communication between the GBJ client and gem.

Effected signals include SigAbort, LostOTRoot, gem-to-gem signals, and object change notification.

Workaround:

Spawn a separate GBJ java client thread that periodically executes a no-op GS Smalltalk operation to trigger any pending asynchronous exceptions. For example:

```
   aGbsSession execute("nil");
```

Bug 41734 - Problems with GBJ client/server class mapping

Product: GemBuilder for Java

Versions: 3.0.1, 3.0

Platform All platforms

Fixed In: 3.1

GBJ provides a mechanism where the application can map a GS server class to a particular java class on the client, using calls to mapGsToJavaClass() and registerStub(). In 3.0 this mapping does not work properly, resulting in all objects being mapping to instances of GbjObject.

GBJ also uses this mechanism to map collection classes to GbjCollection. Since this doesn't work properly, java operations involving Enumeration will also fail.

Workaround:

No workaround.

Bug 40956 - gslist -v raises exception in GbjBroker

Product: GemBuilder for Java

Versions: 2.3, 2.2, 2.1.2.1, 2.1.2, 2.1.1, 2.1, 2.0

Platform All platforms, GS/64 2.4 and later only

Fixed In: 3.0

Starting in GS/64 2.4, a change was made in socket behavior that causes an exception to be raised in the GbjBroker when the "gslist -v" command pings the GbjBroker's socket. The GbjBroker traps the exception and continues normally, but the resulting error message and stack traces will clutter up the GbjBroker log, and there is a tiny increase in memory allocation. This is normally not a problem for most customers, but those that make extensive use of "gslist -v" for monitoring the GbjBroker status may note the memory growth over an extended period of time.

Workaround:

File-in and commit the following code as SystemUser:

```
! Fix for bug 40956
! Perform a socket read: 0 call and check result
! against zero before performing isConnected
!
category: 'Servicing'
method: GsbServerSocket
establishService: aGsbClientSocket
  "Establish a service for aGsbClientSocket."
  | fw |
  fw := self framework.
  " 40956: add read: 0 check "
  (((aGsbClientSocket read: 0 into: String new) = 0) and:
     [aGsbClientSocket isConnected]) ifFalse: [
    " The new socket disconnected, probably from a gslist -v query.
      Just clean it up using a low-level close. "
    (fw configuration at: #verbose) ifTrue: [
      fw log: #GsbClientSocketDisconnected
         arguments: #[aGsbClientSocket]].
    aGsbClientSocket _zeroArgPrim: 2.
    ^ self ].
  aGsbClientSocket linger: true length: 10.    "*** TODO *** Why 10?"
  (fw configuration at: #verbose) ifTrue: [
    fw log: #GsbClientSocketConnectionEstablished
       arguments: #[aGsbClientSocket, GsbFramework dateTimeNow]].
  self serviceBlock value: aGsbClientSocket
%
```

Bug 40614 - GbjLauncher occationally triggers NullPointerException

Product: GemBuilder for Java

Versions: 3.1.1, 3.1, 3.0.1, 3.0, 2.3, 2.2, 2.1.2, 2.1.1, 2.1

Platform All platforms

Fixed In: 3.1.2

When starting the GbjLauncher, it will occationally trip over a NullPointerException during initialization of its Java Swing graphics components. A java stack trace is printed to stdout.

The GbjLauncher will continue to startup normally, and the exception can be safely ignored.

Workaround:

No workaround is necessary, as the GbjLauncher will continue to startup and operate normally.

Bug 35500 - GsFile method failures on client operations

Product: GemBuilder for Java

Versions: 2.3, 2.2, 2.1.2, 2.1.1, 2.1, 2.0, 1.2.1

Impact: Informational

To avoid a security loophole, client side GsFile operations are disallowed. Attempts to execute GsFile openWrite:, for example, will fail with the error:

```
  An attempt was made to execute the client user action #GsfGetId
  from a session that is not allowed to do this.
```

Bug 33189 - GbjBroker problem behavior during/after gslist -v

Product: GemBuilder for Java

Versions: 2.1.2, 2.1.1, 2.1, 2.0

Platform All

Fixed In: 2.2

When a "gslist -v" command is performed while a GbjBroker is running, a disconnected socket is left behind in the broker. This will result in error messages in the GbjBroker log about the disconnected socket, and the left-over socket will tie up memory and socket resources on the system. In addition, on HPUX the disconnected socket will cause the GbjBroker process to run "hot", consuming CPU cycles as the GbjBroker attempts to poll the socket.

Workaround:

To avoid these problems, file-in and commit the following code as SystemUser:

```
category: 'Servicing'
method: GsbServerSocket
establishService: aGsbClientSocket
  "Establish a service for aGsbClientSocket."
  | fw |
  fw := self framework.
  aGsbClientSocket isConnected ifFalse: [
    " The new socket disconnected, probably from a gslist -v query.
      Just clean it up using a low-level close. "
    (fw configuration at: #verbose) ifTrue: [
      fw log: #GsbClientSocketDisconnected
         arguments: #[aGsbClientSocket]].
    aGsbClientSocket _zeroArgPrim: 2.
    ^ self ].
  aGsbClientSocket linger: true length: 10.
  (fw configuration at: #verbose) ifTrue: [
    fw log: #GsbClientSocketConnectionEstablished
       arguments: #[aGsbClientSocket, GsbFramework dateTimeNow]].
  self serviceBlock value: aGsbClientSocket
%
```

Bug 30351 - Stale gslist info prevents current info from being registered

Product: GemBuilder for Java

Versions: 2.3, 2.2, 2.1.2, 2.1.1, 2.1, 2.0, 1.2.1, 1.2

Platform All

Fixed In: 3.0

When a GBJ broker dies or is killed, it is listed under gslist as "killed". When a new broker is started, the old gslist entry remains, and the new broker is not listed.

The expected behavior is for the new broker to fail to start until the "killed" broker's info is cleared by issuing 'gslist -c'.

Workaround:

Before starting the GBJ broker, execute 'gslist -c' to clear any stale gslist information.

Bug 30012 - GBJ corrupts characters in strings with characters with codepoints over 127

Product: GemBuilder for Java

Versions: 3.1.2, 3.1.1, 3.1, 3.0.1, 3.0, 2.3, 2.2, 2.1.2.1, 2.1.2, 2.1.1, 2.1

Platform All Platforms

Fixed In: 3.1.3

Strings containing characters with codepoints > 127 can become corrupted when transferred from the GBJ client to the server or from server to GBJ client. Corruption varies according to version of GBJ.

As of GBJ 3.0:

For extended ASCII characters (codepoint 128 to 255):
* Server-to-GBJ-client: No problems.
* GBJ-client-to-Server: Extended ASCII characters converted into UTF-8 multi-character codes.

For 16-bit characters (codepoint 256-0xFFFF):
* GBJ-client-to-Server: No problems, result is a DoubleByteString.
* Server-to-GBJ-client: Characters are byte-swapped on little-endian platforms (Linux, Windows).

Bug 29166 - Windows startgbj does not produce output

Product: GemBuilder for Java

Versions: 2.3, 2.2, 2.1.2, 2.1.1, 2.1

Platform Windows

Fixed In: 3.0

The startgbj executable fails to show the output of the GBJ Broker process, so it is not immediately clear whether the startup has succeeded.

Workaround:

Check for GBJ Broker startup by using gslist and reading the Broker's logfile (gbjbroker.log by default).

Bug 28515 - Windows Broker hangs reading Unix gbj.ini

Product: GemBuilder for Java

Versions: 2.3, 2.2, 2.1.2, 2.1.1, 2.1, 2.0

Platform Windows

Fixed In: 3.0

The Smalltalk process that runs the GBJ Broker will hang when trying to parse a gbj.ini file with Unix line endings and fail to start its service loop.

Workaround:

Make sure the gbj.ini file has Windows line endings. If you need to transfer it from a Unix host, make sure you ASCII mode when using ftp.

Bug 28089 - Debugging continues beyond application code

Product: GemBuilder for Java

Versions: 2.3, 2.2, 2.1.2, 2.1.1, 2.1, 2.0, 1.2.1, 1.2, 1.1, 1.0

Platform All

When debugging application code with the GemBuilder for Java debugger, debugging continues beyond application code and into system code. This can lead to unrecoverable conditions where the client gets endless single step errors.

Workaround:

When debugging, do not step beyond the context of executed code. If the client becomes unusable, logging out and logging in again is necessary.

Bug 27761 - GbjSession instances not reusable

Product: GemBuilder for Java

Versions: 2.1.2, 2.1.1, 2.1, 2.0, 1.2.1, 1.2

Platform All

Fixed In: 2.2

After sending close() to an instance of GbjSession, subsequent connect() messages fail with a GbjRuntimeException. The message from this exception is "Attempt to send message to GemStone while a message is being transmitted."

Workaround:

Do not attempt to reuse GbjSession instances.

Bug 27514 - Anomalies in GBJ Debugger highlighting

Product: GemBuilder for Java

Versions: 3.1.1, 3.1, 3.0.1, 3.0, 2.3, 2.2, 2.1.2, 2.1.1, 2.1, 2.0

Fixed In: 3.1.2

When using the GBJ debugger to step through code, the highlight for the current break point may not correctly cover the current token. Specifically, the highlight may cover only a portion of the token, or it may also include the whitespace before the token. Since the highlighted area is always between the last token and the current token, you can still use the debugger to step into or over code.

Workaround:

No workaround.

Bug 17980 - Problems with file-in in GBJ tools

Product: GemBuilder for Java

Versions: 3.1.3, 3.1.2, 3.1.1, 3.1, 3.0.1, 3.0, 2.3, 2.2, 2.1.2, 2.1.1, 2.1, 2.0, 1.2.1, 1.2, 1.1

Platform All

The GemBuilder for Java interface to GemStone S has some known problems with filing in GemStone code. Some GemStone tokens may not be recognized and it is possible to receive errors on legal filein code.
If you experience filein problems with legal GemStone filein code, the recommended work-around is to use topaz to file the code in.
If your filein contains an actual error (typo, syntax, etc.) it is not possible to continue execution past the error - ie, if you select continue from the error notifier or the debugger, execution terminates.

Workaround:

Use topaz to file-in your code.

Bug 14391 - Rogue sessions must be stopped manually

Product: GemBuilder for Java

Versions: 2.3, 2.2, 2.1.2, 2.1.1, 2.1, 2.0, 1.2.1, 1.2, 1.1, 1.0

Platform All

It is not possible from GBJ to interrupt GemStone activity.
Therefore, if your GemStone Smalltalk code gets into an infinite loop or becomes otherwise unresponsive there is no recourse but to kill the Gem process. Closing the Java application is not sufficient since the "rogue" Gem will continue to run, tying up System resources.
Use the Unix 'kill' command or the Windows NT Task Manager to terminate the Gem. The process id can be found in the Gem session's log file.

Workaround:

No workaround

GemStone

Bug 49693 - On AIX unable to start shared page cache larger than 2GB

Product: GemStone/S

Versions: 6.7.2

Platform AIX

Fixed In: 6.7.2.1

Impact: Critical

Due to a compiler optimization, attempting to start a shared page cache over 2GB on AIX results in SIGSEGV failure in the SPC monitor. While the exact cache limit depends on various demands on memory, GemStone can generally reach a shared page cache size of 2.9 GB on AIX.

Workaround:

Upgrade to v6.7.2.1.

Bug 49661 - System timeGmt2005 overflows to LargeInteger on 1/5/2022

Product: GemStone/S

Versions: 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, older versions

Fixed In: 6.7.2

Impact: Informational

On January 5, 2022, the number of seconds since January 1, 2005 will exceed the value of SmallInteger maximumValue. At this time, the method System class>>timeGmt2005 will begin returning instances of LargePositiveInteger.

If your application uses the result of timeGmt2005 to assign to an instvar or element that is constrained to SmallInteger, it will trigger a constraint violation exception.

Since SmallIntegers are specials (where the oop directly encodes the numeric value), while LargePositiveIntegers are full objects assigned to an oop, the change to LargePositiveIntegers will use up OOPs within your repository, as well as extent space. There are also performance costs in performing arithmetic on LargePositiveIntegers vs. SmallIntegers.

Workaround:

Before January 2022, it is critical that customers examine their application/s for uses of timeGmt2005, and determine the impact on the application. Upgrading to 6.7.2 or later before this changeover is strongly recommended.

The GemStone 6.7.2 release includes fixes for the similar issue impacting RcQueues (see bugnote for 46377.

The 6.7.2 release also includes the new method System class >> timeGmt2020, which provides the seconds since January 1, 2020, in seconds. It will return a SmallInteger until January 4, 2037.

Note that GemStone/S 64 Bit has a much larger SmallInteger range and is not subject to similar issues; converting to GemStone/S 64 Bit is strongly recommended for all 32-bit GemStone/S applications.

Bug 47826 - 6.7.0 to 6.7.1 upgrade does not require file-in

Product: GemStone/S

Versions: 6.7.1

Platform All Platforms

Impact: Informational

There are no image changes between GS/32 versions 6.7.0 to 6.7.1, so a file-in of kernel methods during an upgrade is not necessary. But we do still need to update internal version information and the (Globals at: #DbfHistory) String.

If you are performing a 6.7.0 to 6.7.1 upgrade, you may safely ignore the normal upgrade procedure and instead use the following instructions:

1. Request the script upgradeImage670to671 from GemTalk Technical Support.

2. Stop your 6.7.0 stone and netldi.

3. Update your $GEMSTONE to the 6.7.1 release.

4. Do a normal startstone (do NOT use startstone -C).

5. Execute the upgradeImage670to671 script:

```
  unix>  upgradeImage670to671 -s <stone> -p <passwordForSystemUser>
```

Note that you *do not* need to change the SystemUser password back to the default 'swordfish' to do the upgrade. If the password *is* 'swordfish' you can omit the -p <password> parameters when calling the upgradeImage670to671 script.

This completes the 6.7.0 to 6.7.1 upgrade procedure. Note that (Globals at: #DbfHistory) will have the following appended to indicate this:

```
GemStone 6.7 to 6.7.1 QUICK upgrade (no kernel file-in) completed at DD/MM/20YR HR:MN:SC
```

Bug 47691 - SIGSEGV in AIX extend_brk( ) call

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7

Platform AIX only

Impact: Informational

When using a large shared page cache close to the maximum size allowed on AIX (2.75GB), there is a slight risk that a stone or gem process may crash with a SIGSEGV in the AIX system call extend_brk( ), used for allocating C-heap memory. This occurs when all available 32-bit memory space has been utilized.

Log entry will look similar to:

```
 GS Signal Handler: Unhandled signal 11, SIGSEGV Received at XXXXXXXXX
 HostFaultHandler: signal (int) = 11
   info->si_signo = 11 = 0xb
   info->si_code =  51 = 0x33
   info->si_errno = 0 = 0x0
   info->si_addr = 0x4  pc = 0x2001109c
 Registers saved from frame receiving the signal:
 Begin attempt to print C-level stack for PID XXXXXXX at: <date> 13501244:
  /opt/gemstone/GemStone6.7.0-RISC6000.AIX/sys/stoned -e<config file>
 0x2011b2c0  waitpid(??, ??, ??) + 0x1c0
 0x1000b228  forkAndWait(const char*,char* const*)() + 0x68
 0x1000ae3c  HostPrintCStack(long)() + 0x1dc
 0x1000a33c  HostCoredump(int)() + 0xadc
 0x1000d6ac  sigCoreExit(int,int,siginfo_t*,void*)() + 0x16c
 0x1000bc80  HostFaultHandler() + 0x640
 <signal> 0x2001109c  extend_brk(??, ??, ??) + 0x2dc
```

. . .

Workaround:

Upgrade to GemStone/64 for the larger 64-bit memory space.

Alternatively, reduce the size of the SPC to allow more room for C heap growth.

Bug 47243 - Login attempts when logins suspended generates clutter in topaz/gem and page manager logs

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.3, 6.5.2, 6.5.1, 6.5

Platform All Platforms

Impact: Informational

When attempting to login while logins are suspended, the client will actually attempt to login 20 times under the covers (once per second) before giving up. Each of these hidden login attempts will generate messages to both the topaz/gem log and the page manager log.

The topaz/gem log will see entries like:

```
topaz> login
[Info]: LNK client/gem GCI levels = 55/55

[Info]: Logging out session -1 at 10/27/2017 03:13:34 PM PDT

[Info]: Logging out session -1 at 10/27/2017 03:13:34 PM PDT

login failed due to error 4053(logins suspended), retrying for 20 seconds

[Info]: LNK client/gem GCI levels = 55/55

[Info]: Logging out session -1 at 10/27/2017 03:13:35 PM PDT

[Info]: Logging out session -1 at 10/27/2017 03:13:35 PM PDT
```

(repeat the last 3 lines 19 more times)

```
-----------------------------------------------------
GemStone: Error         Fatal
Login failed:  GemStone logins are currently disabled. Error Category: 3613 [GemStone] Number: 4053 Arg Count: 0
Context : 10 topaz>
```

While the page manager log will contain entries like:

```
[10/27/2017 03:13:55 PM.086 PDT]
   START setLostOt: sess 5 gem -1 pgsvr -1 shutdown 1 isRemote 0
   END setLostOt: wrotePteLocal 0
```

(repeated 20 times)

Bug 46931 - Problems reading Unix-formatted text files on Windows

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.5.8, 6.5.7.5, 6.5.7, 6.3.1, 6.1.6, 6.1.5

Platform Windows only

Unix-formatted text files using just the single line-feed character as a line separator cannot be properly read using GsFile. There are problems with the underlying Windows OS call used to retrieve position, which will in turn cause incorrect results for other GsFile reading methods (position, atEnd, next*).

Workaround:

1. Convert the Unix-formatted file into a standard Windows format using a utility such as unix2dos.

2. Open the file in binary mode. For example:

file := GsFile open: 'filename' mode: 'rb'.

Note that if you open a Windows-formatted text file in binary mode you'll get separate CR and LF characters on subsequent #next calls rather than a single LF.

Bug 46377 - RcQueues have issues on January 5, 2022

Product: GemStone/S

Versions: 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, older versions

Fixed In: 6.7.2

On January 5, 2022, the number of seconds since January 1, 2005 (as returned, for example, by the method System class>>timeGmt2005) will exceed the value of SmallInteger maximumValue. RcQueues use this value to mark the time elements are added to the queue.

After this change, the createTime in RcQueueEntries will roll to a negative SmallInteger (starting with negative SmallInteger maximumValue - 1), and continue to increase.

While the RcQueue will continue to work, elements added after the rollover will be returned out of order, before elements added before the rollover.

Workaround:

The GemStone v6.7.2 release provides a new class for elements, RcQueueEntry2020, which uses the seconds since 2020. These will be automatically used following upgrade to 6.7.2, and avoids any issues at the time of changeover.

Note that in January 2037, the createTime for RcQueueEntry2020 will roll over in turn.

Bug 46359 - Using a 64-bit Java with GBJ on 32-bit GS/S fails with UnsatisfiedLinkError

Product: GemBuilder for Java

Versions: 3.1.3

Platform All Platforms

The GbjGci Library built for GS/S is 32-bit. Attempting to run GBJ using a 64-bit Java will fail with the error:

java.lang.UnsatisfiedLinkError: $GEMSTONE/lib/libgbjgci313.so: ld.so.1: java: fatal: $GEMSTONE/lib/libgbjgci313.so: wrong ELF class: ELFCLASS32 (Possible cause: architecture word width mismatch)

Oracle no longer provides a 32-bit Java for Solaris as of version 1.8. Because of this, GBJ will not run on 32-bit GS/S on Solaris (but see workaround).

Workaround:

Use a 32-bit Java executable.

If you must run Java on Solaris (either x86 or SPARC), contact GemTalk Technical Support, which can generate a special version of GBJ using the older 1.7 version of the Java SDK. You will need to run 32-bit Java version 1.7, which *is* available.

Bug 46264 - GBJ 3.0 library load fails on Windows 64-bit machines

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7, 6.5.6, 6.5.5

Platform Windows 64-bit only

Impact: Informational

When running GBJ 3.0 on a 64-bit Windows machine, make sure you are running with a 32-bit java VM (use the java -d32 flag to insure this). Using a 64-bit java VM will cause the GbjGci Library load to fail with the following generic error:

```
-----------------------------
WARNING: Problem loading GbjGciLibrary com.gemstone.gbj.GbjException: Problem loading GbjGciLibrary at com.gemstone.gbj.GbjSession.initialize(GbjSession.java:719) at com.gemstone.gbj.GbjSession.connect(GbjSession.java:818)
. . .
-----------------------------
```

Workaround:

Use a 32-bit java VM (java -d32)

Bug 45907 - installgs fails due to incorrect environment variable

Product: GemStone/S

Versions: 6.7

Fixed In: 6.7.1

The installgs script errors on startup with a ErrUtlReset: fopen() failure, on $GEMSTONE/sys/english66.err:

The source of the problem is an incorrect version number. An environment variable that sets the version number in the setup script, was inadvertently not updated for the 6.7 release, and still specifies version 6.6.

Workaround:

edit $GEMSTONE/bin/misc.sh to modify the version number:

```
. . .
GEMSTONE_VERSION=66
. . .
```

should be

```
GEMSTONE_VERSION=67
```

Bug 45786 - GsFile contentsOfDirectory: can fail to find entries on NFS file servers

Product: GemStone/S

Versions: 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.x

Fixed In: 6.7

GsFile contentsOfDirectory: can fail to find entries on NFS file servers when these send back offsets that overflow the 32-bit field used by readdir( ).

Bug 45759 - "Zombie" sessions can cause CR backlogs

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5

Platform All Platforms

There is a small risk that a session can hang during logout, causing the stone to be confused as to if the session is alive or dead, hence the nickname "zombie" session.

These zombie sessions can still hold on to a transactional view and cause a CR backlog. The zombie may not respond to a System stopSession:, and even when the OS process is killed, the stone may still think the session exists.

The following tools may be useful in dealing with zombies:

System allZombieSessions
System allZombieSessionsRefOldestCr
System stopZombieSession: <sessionID>

Bug 45744 - Inconsistent lock status after failed commit on temporary locked object

Product: GemStone/S 64 Bit

Versions: 3.2.16, 3.2.15, 3.2.14, 3.2.13, 3.2.12, 3.2.11, 3.2.10, 3.2.9, 3.2.8.1, 3.2.8, 3.2.7, 3.2.6, 3.2.5, 3.2.4, 3.2.3, 3.2.2, 3.2.1, 3.2, 3.1.0.6, 3.1.0.5, 3.1.0.4, 3.1.0.3, 3.1.0.2, 3.1.0.1, 3.1, 3.0.1, 3.0, 2.4.8, 2.4.7, 2.4.6, 2.4.5.1, 2.4.5, 2.4.4.8, 2.4.4.7, 2.4.4.4, 2.4.x, 2.3.1.6, 2.3.x, 2.2.5.4, 2.2.x, 2.x, 1.x

Fixed In: 3.3

If you perform the following sequence:
create a temporary object
lock this temporary object using methods such as System writeLock:, etc.
attempt a commit, but encounter a commit failure.
abort the session

Then the lock on the temporary object remains, but is in a invalid state. The lock can't be removed, and methods to access lock status such as sessionLocks are inconsistent.

Workaround:

Removing the lock before aborting avoids problems. Using System removeLocksForSession allows you to clear the state.

Bug 44611 - MinusQuietNaN and MinusSignalingNaN are printed incorrectly

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, older versions

Platform All

When inspecting or printing MinusQuietNaN and MinusSignalingNaN they are displayed as the Plus... counterpart, although the bits of the values are correct.

Bug 44609 - Problems mixing 6.6.4 / 6.6.5 components

Product: GemStone/S

Versions: 6.6.5

Platform All Platforms

Normally, when new versions of GemStone are released that contain changes in internal data format and/or communication protocols between GemStone processes, an internal compatibility level is incremented so that attempts to mix-and-match incompatible versions (for example, trying to login a 6.6 topaz to a 6.5 stone) will generate a #gsErrBadGemStoneVersion error (4044) with the message:

"The Gem and Stone versions are incompatible. Stone version X.X.X , gem version X.X.X"

This level was not properly incremented in 6.6.5, so that the correct error message is not generated, but instead other problems occur. Depending on the combination of components, possible problems include:

1. The correct error, but the gem version is listed as "unknown".
2. The login attempt hangs until the netldi times out.
3. The login succeeds, but due to changes in the format of cache statistics in the SPC, information on these will be incorrect.

Customers are cautioned to be extra careful not to allow this to happen, since the usual internal safeguard is not in place.

Bug 44548 - Multiple set-valued comparison predicates conjoined with & produce inconsistent results

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.x, 5.0

Due to the design of the query code, multiple predicates that are conjoined with &, that query on the same set-valued collection path, and which use a comparison rather than equality type operator, may produce inconsistent results.

Note that queries that can be restructured by optimization to remove the &, or reduce the predicates to one for the collection path, will not have this problem.

Workaround:

We recommend avoiding set-valued indexes in general. Single-predicate queries, or queries that use =, ~=, ==, and ~~ are not affected by this case.

Bug 44498 - Nested block variable resolution with to:do: may be incorrect

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.3, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.2.x, 6.1.6, 6.1.5, 6.1.x

The resolution of variables in nested blocks is not correct in some cases involving to:do: optimizations.

For example, the following code returns #(nil, nil, nil, 4); the correct result is #(1 2 3 4).

```
 | a b |
 a := Array new: 4.
 b := Array new: 4.
 1 to: a size do: [:i |
       b at: i put: [10 timesRepeat: [a at: i put: i]]
 ].
 1 to: b size do: [:j | (b at: j) value].
 a
```

Bug 43463 - Logins during commitRestore can cause commitRestore to fail

Product: GemStone/S

Versions: 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.3.1, 6.1.6, 6.1.5

Platform All Platforms

Fixed In: 6.6.5

Repository>>commitRestore requires that the session performing the commitRestore be the only session logged in. It does a suspendLogins, and then checks that it is the only one, failing with #rtErrNotOnlyUser (2140) if it isn't.

But under certain conditions, a session login that takes a long time can be missed by this check. It may have started logging in just before the suspendLogins, but not yet completed the process when the commitRestore session checks to see if it is the only one. This can happen with remote gems when they have to set up a remote cache.

The presence of this second session will cause the commitRestore to fail with the error:

RecovCommitRestore: problem disposing commit records.

Workaround:

Before performing the commitRestore, you should do the following to guarantee that there is only one session:

```
  System suspendLogins.
  System sleep: 5.
  System stopOtherSessions.
```

The 5 second sleep is to give any sessions that sneak by the suspendLogins time to complete their login, so that they will be properly terminated by the stopOtherSessions. Depending on your site you may have to make the sleep longer.

Bug 43268 - SPC Monitor startup failures on ftok( )

Product: GemStone/S

Versions: 6.6.3.3, 6.6.3.2, 6.6.3.1, 6.6.3, 6.6.2.1, 6.6.2, 6.6.1, 6.6

Platform AIX only

Fixed In: 6.6.4

On AIX, startstone will occasionally fail due to a problem with the startup of the shrpcmonitor process. The SHRPCMON log will contain the following error message:

GemStone could not retrieve the segment key using ftok(). path = /opt/gemstone/locks/<lockname> prodid = 0, error = .
[SpcMon trace]: ... cache creation failed ...

Workaround:

No workaround, but a subsequent startstone will usually work.

Bug 42841 - Incorrect values for cache stats FreeTempOopCount and FreePersistentOopCount

Product: GemStone/S

Versions: 6.6.3.2, 6.6.3

Platform All

Fixed In: 6.6.3.3

The value of the cache stats FreeTempOopCount and FreePersistentOopCount are incorrect; FreeTempOopCount is updated incorrectly and will have incorrectly large values, while FreePersistentOopCount is not always updated and will be incorrectly low.

Bug 42708 - Excessive #mfcGcPageBufSize causes ALL data to be removed

Product: GemStone/S

Versions: 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.1.x

Platform All

Fixed In: 6.6.3.2

Impact: Critical

If you run markForCollection with an extremely oversized #mfcGcPageBufSize, ALL data in your repository is removed. The specific example observed was an accidental test with a #mfcGcPageBufSize value of 8000000, which is not in any sense a reasonable value; since this is in pages, this represents 65GB, not including other memory requirements for MFC.

Most excessively large #mfcGcPageBufSize values result in malloc() errors, but due to the Integer rollover there is a range in which this error can occur.

Workaround:

If you experiment with unreasonably large #mfcGcPageBufSize values and encounter this problem, you will need to restore from backup.

Bug 42354 - restoreFromBackup(s):newSystemUserPassword can fail

Product: GemStone/S

Versions: 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6

Platform All platforms

Fixed In: 6.6.5

Under certain conditions, the use of a backup restore method that includes the keyword newSystemUserPassword: can fail with a #rtErrBadArgKind (error number 2094). This includes:

SystemRepository>>restoreFromBackup:newSystemUserPassword:
SystemRepository>>restoreFromBackups:newSystemUserPassword:

The failure occurs after the stone has committed the newly restored DB, during the follow-up transaction to change the SystemUser password to it's new value. The resulting DB is OK, but the SystemUser password still retains its original value from the backup file.

Workaround:

No workaround really necessary -- just remember that the resulting DB still retains the original SystemUser password as recorded in the backup file. You can avoid the error altogether by just using the restore methods without the keyword newSystemUserPassword:.

Bug 41794 - Turning on stale account aging may disable accounts

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.x, 5.1.5.1, 5.1.5, earlier versions

Platform All

The lastLoginTime of an account is only updated if the repository has stale account aging or password aging enabled. The update of the lastLoginTime on login requires a commit, which is not always desireable.

As a result, if a stale account age limit is set in a repository that did not previously have either check, the lastLoginTime of accounts that log in frequently may be still set to a date well in the past, which results in the account being disabled immediately.

Workaround:

Decisions to enable or disabled account and password aging should be done with forethought, in any case.

If you are setting the stale account age limit on a repository that has been in use, then you should send lastLoginTime: to each UserProfile, to initialize the lastLoginTime correctly. The method lastLoginTime: was added in version 6.6

To turn on account aging safely on versions prior to 6.6, an initial period with the stale account age limit set to a large value, or with password age check enabled but not account age limits, will allow accounts time to login in with updates to the lastLoginTime.

Bug 41366 - VSD problems with Windows 7

Product: GemStone/S

Versions: 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.3, 6.5.2, 6.5.1, 6.5, 6.x

Platform Windows

Fixed In: 6.7

Impact: Informational

When running older versions of VSD on Windows 7, the VSD tool crashes on startup. Note that VSD for Windows is distributed by download from the support site, is it not part of the server distribution in versions earlier than 6.7.

Workaround:

Recent version of the VSD are available on the website, gemtalksystems.com/vsd. While these were recompiled and distributed for the 64-bit product, they are compatible with the 32-bit product.

Bug 41342 - Encoded size ByteArray comparison method primitive failures

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.3, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.2, 6.2.1, 6.1.6, 6.1.5, 6.1.4

Platform All platforms

When using one of the two encoded size ByteArray comparison methods with
indexing arguments other than one, they will occationally fail with #rtErrPrimFailed
(2258).

The problem methods are:

ByteArray>>compareStringAt:to:startingAt:sizeBytes:useCase:
ByteArray>>shortStringAt:compareWith:startingAt:opCode:

Using a value other than 1 as an argument to *StringAt: or startingAt:
will expose the bug.

Workaround:

No workaround.

Bug 41019 - Performance of ordering on multi-predicate indexed queries

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.5, 6.5.4, 6.5.3, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.2, 6.2.1, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.x, 5.x

Fixed In: n/a

Impact: Informational

When performing a multiple predicate indexed query, for the best performance, the predicates should be ordered so that the most restrictive predicate is evaluated first. Predicates are evaluated from last to first (not the intuitive evaluation order), so the most restrictive predicate should be last in the query code.

For example, in the following query

```
 AllCustomers select:
   {:customer |
   (customer.status==#valid) & (customer.lastName=aName)}
```

The query predicate on lastName is performed first, resulting in a much smaller set that the second predicate must perform the query over.

Workaround:

To avoid perturbing existing tuned queries, we will not be changing this. Please examine your existing multi-term queries to ensure the ordering is optional for performance.

Bug 40652 - stack corruption returning from ClientForwarder send

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.1.6, 6.1.5, 6.1.x

Platform All

Occasionally, under unknown circumstances, the value returned back into GemStone from a client forwarder send will be a two element array containing the expected result, rather than the result itself. When this happens, the first element of the array is the selector of the client forwarder send, and the second element is the expected result. This usually subsequently results in a "Does not understand" error in GemStone.

Workaround:

In GemStone code, check if the result of a client forwarder send is a two element array, and if so, use the second element as the result.

Bug 39900 - Remote cross-platform MFC has risk of gem crash

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.3, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.2, 6.2.1, 6.2, 6.1.6, 6.1.5, 6.1.4, 6.1.3, 6.1.2, 6.1.1, 6.1

Platform All

Running a markForCollection (MFC) from a remote gem that is on a machine with a different byte order than the stone, and therefore requires swizzle, can result in a gem crash with a preceeding message in the log:

"ObjSupSwizzlePageGet error on page 16778280 oop 10944580 (message can be ignored if no further problems)"

Byte swizzling is required when a remote gem is Windows or Linux, and the stone on Solaris, AIX or HP (or vice versa).

Workaround:

Avoid running MFC from gems that are remote from the stone and on a machine with a different byte ordering.

Bug 39740 - ByteArray with DateTimes can't be used cross platform

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.3, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.2, 6.2.1, 6.2, 6.1.6, 6.1.5, 6.1.4, 6.1.3, 6.1.2, 6.1.1, 6.1

Platform Linux, Windows

ByteArray stores and reads DateTime instances in native byte order, which means the resulting ByteArrays cannot be moved between Intel (Linux or Windows) and non-Intel platforms.

Bug 39679 - Logout with logins suspended rereads configuration file

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.1.5, 6.1.4, 6.1.3, 6.1.x

Platform All

Executing stopOtherSessions suspends logins in the repository. If the user that performed the stopOtherSessions logs out, logins are reenabled automatically. The code that reenables logins under these circumstances re-reads the configuration file. If you have modified the configuration programatically in your repository, but not updated the configuration file, these changes will be lost.

Workaround:

If you do not modify the configuration programatically, there is no issue.

If a user executes stopOtherSessions, they should explicity reenable logins by calling System resumeLogins, before logging out.

Bug 39639 - ByteArray methods may fail

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.4, 6.1.3, 6.1.2, 6.1.1, 6.1.x, 6.0.1, 6.0.x, 5.1.5.1, 5.1.x

Platform Windows

Calling the method ByteArray >> at:put:signed:width: with a width less than four results in only 0 showing up in the resulting ByteArray.

Bug 39356 - Change in character collation sequence can break indexes

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2

Platform All platforms

Fixed In: n/a

Impact: Critical

A change in the collation sequence of baseline extended ASCII characters (range 0-255) was inadvertently introduced in GS/S 6.2 with the introduction of Unicode character support. The change moved the collation sequence of characters with diacritic marks so that they would sort adjacent to their associated "plain vanilla" characters (originally they sorted towards the end of the sequence following their ASCII code).

The affected characters are:

192: LATIN CAPITAL LETTER A WITH GRAVE
193: LATIN CAPITAL LETTER A WITH ACUTE
194: LATIN CAPITAL LETTER A WITH CIRCUMFLEX
195: LATIN CAPITAL LETTER A WITH TILDE
196: LATIN CAPITAL LETTER A WITH DIAERESIS
197: LATIN CAPITAL LETTER A WITH RING ABOVE

170: FEMININE ORDINAL INDICATOR (treated like small a)
224: LATIN SMALL LETTER A WITH GRAVE
225: LATIN SMALL LETTER A WITH ACUTE
226: LATIN SMALL LETTER A WITH CIRCUMFLEX
227: LATIN SMALL LETTER A WITH TILDE
228: LATIN SMALL LETTER A WITH DIAERESIS
229: LATIN SMALL LETTER A WITH RING ABOVE

199: LATIN CAPITAL LETTER C WITH CEDILLA
231: LATIN SMALL LETTER C WITH CEDILLA

200: LATIN CAPITAL LETTER E WITH GRAVE
201: LATIN CAPITAL LETTER E WITH ACUTE
202: LATIN CAPITAL LETTER E WITH CIRCUMFLEX
203: LATIN CAPITAL LETTER E WITH DIAERESIS

232: LATIN SMALL LETTER E WITH GRAVE
233: LATIN SMALL LETTER E WITH ACUTE
234: LATIN SMALL LETTER E WITH CIRCUMFLEX
235: LATIN SMALL LETTER E WITH DIAERESIS

204: LATIN CAPITAL LETTER I WITH GRAVE
205: LATIN CAPITAL LETTER I WITH ACUTE
206: LATIN CAPITAL LETTER I WITH CIRCUMFLEX
207: LATIN CAPITAL LETTER I WITH DIAERESIS

236: LATIN SMALL LETTER I WITH GRAVE
237: LATIN SMALL LETTER I WITH ACUTE
238: LATIN SMALL LETTER I WITH CIRCUMFLEX
239: LATIN SMALL LETTER I WITH DIAERESIS

209: LATIN CAPITAL LETTER N WITH TILDE
241: LATIN SMALL LETTER N WITH TILDE

210: LATIN CAPITAL LETTER O WITH GRAVE
211: LATIN CAPITAL LETTER O WITH ACUTE
212: LATIN CAPITAL LETTER O WITH CIRCUMFLEX
213: LATIN CAPITAL LETTER O WITH TILDE
214: LATIN CAPITAL LETTER O WITH DIAERESIS

186: MASCULINE ORDINAL INDICATOR (treated like small o)
242: LATIN SMALL LETTER O WITH GRAVE
243: LATIN SMALL LETTER O WITH ACUTE
244: LATIN SMALL LETTER O WITH CIRCUMFLEX
245: LATIN SMALL LETTER O WITH TILDE

246: LATIN SMALL LETTER O WITH DIAERESIS

217: LATIN CAPITAL LETTER U WITH GRAVE
218: LATIN CAPITAL LETTER U WITH ACUTE
219: LATIN CAPITAL LETTER U WITH CIRCUMFLEX
220: LATIN CAPITAL LETTER U WITH DIAERESIS

249: LATIN SMALL LETTER U WITH GRAVE
250: LATIN SMALL LETTER U WITH ACUTE
251: LATIN SMALL LETTER U WITH CIRCUMFLEX
252: LATIN SMALL LETTER U WITH DIAERESIS

221: LATIN CAPITAL LETTER Y WITH ACUTE

253: LATIN SMALL LETTER Y WITH ACUTE
255: LATIN SMALL LETTER Y WITH DIAERESIS

Due to this change, repositories converted from a pre-6.2 system to a post-6.2 system will exhibit the following problems:

* Encoded integers representing ANY string generated using Array>>_insertEncodingForString:arraySize: (primitive 542) on the pre-6.2 system will not generate the correct decoded String when using Array>>_decodeKeyAt:decoding:into: (primitive 831) on the post-6.2 system.

* Indexed collections generated on the pre-6.2 system on strings which contain these characters will not return correct search results on the post-6.2 system.

Workaround:

There are two approaches to work around these problems:

1. It is possible to reconfigure the post-6.2 system to use the original collation sequence. For this option to work, it must be applied before any new elements are added to the indexed collections or new encoded values generated. The procedure is as follows:

   1. Acquire the following passivate data file from GemStone Techical Support:

      OldCollateCharTable.dat

   2. Login to the system as SystemUser and execute the following:

      ```
      Character activateCharTablesFromFile: 'OldCollateCharTable.dat'.
      System commitTransaction.
      ```

   All sessions that login subsequently will use the "new" original collation sequence.

2. Delete and reconstruct any affected indexes, recalculate any encoded values, using the new collation sequence.

Bug 38988 - VSD no longer shipped with Windows server

Product: GemStone/S

Versions: 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3

Platform Windows

Fixed In: 6.7

Impact: Informational

Due to changes in the build process, the Visual Stat Display application (VSD) is no longer shipped as part of the Windows server. It is available by download on the support site at:
http://gemtalksystems.com/vsd/

Bug 38086 - Assigning object to extent may not succeed

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2

Platform All

Fixed In: n/a

Impact: Informational

Earlier versions supported using ClusterBuckets to allow assigning an object to a particular extent, using ClusterBucket>>extentId: or ClusterBucket>>newForExtent:. These methods are deprecated. Clustering will attempt to use the specified extent, but is not guarenteed.

Bug 37973 - Multiple Stones can attach to SAN extents causing corruption

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.2, 6.0.1, 6.0

Platform ALL Platforms

Impact: Critical

When starting up a GemStone system, the stone performs an initial check that it has sole access to the extents by opening the first extent in exclusive write mode. If the open fails, the stone assumes that another stone and associated processes are using that DB and will shutdown.

Unfortunately, some SAN (Storage Area Network) configurations can cause this check to fail when the two stones are on different machines. This may be the case in a primary/fail-over configuration that uses the SAN to make the extents/tranlogs appear as local files to both.

Allowing the second stone to startup and proceed to make changes to the extents and tranlogs will cause massive corruption to the extents and loss of data in the tranlogs.

Workaround:

If you are using a SAN as part of a primary/fail-over configuration as described above, you should provide some additional check to your architecture to prevent accidently starting the fail-over stone while the primary is still running (or vise-versa).

Alternatively, you should configure the tranlogs to be in different directories (perhaps having to copy the final tranlog from the primary to the fail-over directory), to prevent overwriting the tranlog and loosing data needed for a restore from backup / tranlog replay.

Bug 36966 - Remote caches do not use SHR_PAGE_CACHE_LOCKED setting

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.2, 6.0.1, 6.0

Platform All

A true setting for the config param SHR_PAGE_CACHE_LOCKED does not get picked up by a shared page cache remote from the stone; this setting is always false.

Workaround:

Edit the startshrpcmon script used on the remote machine to hardcode this parameter; or more properly make a copy of the script and modify services.dat to use your customized script.

Bug 36273 - Use of TZ environment variable results in incorrect Times.

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.1.5, 6.1.x, 6.0.x, 5.1.5.1, 5.1.5

Platform Windows

If the TZ environment variable is set, the current time as calculated for topaz headers and from executing "Time now" will be incorrect.

Workaround:

Avoid use of the TZ environment variable when running GemStone sessions under Windows.

Bug 34314 - auditInternalStructures fails for some UnorderedCollection subclasses

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5

Platform All

While the methods auditInternalStructures and auditInternalStructuresWithRepair: are implemented in UnorderedCollection, they do not apply for all subclasses of UnorderedCollection. Due to the way certain subclasses are implemented, these subclasses are not subject to the bug that this audit method detects, and the audit may incorrectly report failure.

Classes for which the audit does not work are:
- Bag
- Set
- RcIdentityBag

Bug 34067 - performOnServer: fails if TMP not writable

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.1.5, 6.1.x, 6.0.x, 5.1.5.1

Platform Windows

The method System class >> performOnServer: creates a temporary file in C:/TMP during processing of the command string. If C:/TMP is not wriable by the user, this method will not return an error, fail to perform the desired operations, and return an empty string.

Workaround:

Change the environment variables TMP and TEMP to a directory that the user has full access to.

Bug 33866 - copydbf to Linux character raw partitions fails

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5

Platform Linux

Attempting to copydbf to a character raw partition on Linux fails with an ENODEV error.

Workaround:

On Linux, configure raw partitions as block devices rather than character devices.

Bug 33644 - During backup, delayed response to lack of free space

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.x, 5.1.5.1, 5.1.5

Platform All

When backup code is executing, it does not respond to messages regarding a shortage of free space. The message is received and processed when backup is complete, which may be some time after the actual free space shortage has occurred and been resolved.
While the backup may report failure with an #rtErrFreeSpaceThreshold error, it is likely that the backup file will be valid and usable.

Workaround:

Avoid running out of free space. If you do see this error on backup completion, the backup is likely to be valid, but if possible we recommend making a new backup.

Bug 33298 - restoreReclaimPages ties up pages until next tranlog

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.x, 5.1.5.1

Platform All

The command SystemRepository>>restoreReclaimPages generates a commit to record its results. A few additional pages are used in recording that commit record. But the pages reclaimed, together with the pages in the commit record, do not become available to the system until the stone disposes that commit record.

Unfortunately, the stone doesn't actually process commit records unless actively replaying a tranlog or the system exits restore mode and is brought on-line. So repeated use of restoreReclaimPages will tie up more and more pages in the associated commit records until the next tranlog is replayed. In systems that repeat restoreReclaimPages while waiting for the next tranlog, you can actually run out of free pages when you might be expecting to get some back.

Workaround:

Either limit the number of iterations of restoreReclaimPages used between two tranlog restores, or use statmonitor/vsd to monitor the available FreePages, and stop using restoreReclaimPages when the value becomes too low.

Bug 33065 - Error on restore of backups of repositories writing tranlogs to /dev/null

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.x, 5.1.5.1, 5.1.5

Platform All

If you take a backup on a repository that has the tranlog directories set to /dev/null, then restore that backup, the commitRestore will return an error "Repository not in restore state". However, the backup will have been restored successfully and you can continue with use of the restored repository.

Bug 32155 - GciSoftBreak will not interrupt sleeping session

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.1.5, 6.1.x, 6.0.x, 5.1.5.1

Platform Windows

On Unix platforms, the GCI function GciSoftBreak() will interrupt a
sleeping session (using System>>sleep: or Delay>>wait). But on Windows,
the session waits until the sleep is completed before responding to
the GciSoftBreak().

Workaround:

Recode applications to use "restless sleep."
For example, instead of using:

```
   System sleep: 60.
```

use instead:

```
   1 to: 60 do: [:i | System sleep: 1 ].
```

Bug 31738 - Exception #rtErrDoesNotUnderstand can fail with SEGV

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.x, 5.1.5.1

Platform All

The following code sequence will generate a SEGV error:

```
   System enableSignaledAbortError.
   Exception undefinedMethod: arg1 next: arg2 next: arg3.
```

where:

1. The undefined method is a method of 3 or more arguments that would
   normally generate an #rtErrDoesNotUnderstand error.
2. One of the arguments passed to the method is a block containing a
   return value (for example: [^ nil]).

This obscure bug was uncovered from the following example (note accidental
captalization of "category"):

```
   System enableSignaledAbortError.
   Exception Category: nil number: 6009 do: [ ^ nil ].
```

Workaround:

Don't do this.

Bug 31689 - doesNotUnderstand can corrupt stack

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.x, 5.1.5.1

Platform All

There is a very small timing window during which the running of not connected
set GC during execution of a doesNotUnderstand method can cause stack

corruption. This is often exhibited as another #rtErrDoesNotUnderstand error during execution of the doesNotUnderstand method.

In practice this error is extreamly rare, and is only exposed in applications that make frequent use of doesNotUnderstand and have frequent occurrences of not connected set GC.

Workaround:

Recode application to minimize use of doesNotUnderstand, and/or reduce the possibility of not connected set GC. Ways of reducing not connected set GC include:

1. If possible, decrease the number of temporary objects generated during a single transaction.

2. Increase the size of the gem's temporary object cache via GEM_TEMPOBJ_CACHE_SIZE.

3. Increase the settings for GEM_NOT_CONNECTED_THRESHOLD and GEM_NOT_CONNECTED_DELTA.

Bug 31348 - Idle sessions don't immediately exit on stopSession:

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.x, 5.1.5.1

Platform All

Impact: Informational

Idle sessions (sessions waiting for user input) do not immediately exit when a stopSession: is received. The stone will wait for one minute, and then consider the session "dead" and remove it from it's internal tables. The session itself (either a gem or topaz process) will continue to live until the next user input is received, after which it will generate an error, depending on if the input is received before or after the 1 minute interval:

```
  < 1 minute (lnk/rpc): 4059 / #gsErrSessionShutdown
  > 1 minute     (rpc): 4137 / #netErr Network partner has disconnected
                 (lnk): 4035 / #gsErrStnNetLost
```

Workaround:

No workaround

Bug 31180 - Creating symbols via GciCreateByteObj fails

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.x, 5.1.5.1

Platform All

Using the GCI function GciCreateByteObj to create a symbol will fail with a SEGV error.

Workaround:

Use the GciNewSymbol() function to create new symbols.

Bug 30982 - Problems with debug step command on untilTrue:/untilFalse:

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.x, 5.1.5.1

Platform All

The GemStone debugging step operator, used from either the GBS debugger or the topaz step command, exhibits various problems when stepping though untilTrue:/untilFalse: statements and their associated blocks.

In some cases, the debugger gets stuck on the untilTrue:/untilFalse: statement. Repeated uses of the step command will not advance the program counter beyond the untilTrue:/untilFalse: statement.

In other cases, spurious execution errors are triggered by statements within the block.

Workaround:

Use breakpoints and "continue" from one breakpoint to the next, rather than stepping through the code.

Bug 30896 - String>>asNumber fails with radix > 10 and character digits

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.x, 5.1.5.1, 5.1.5

Platform All

String >> asNumber can be used to convert Strings containing Integers in radix format into Integers. However, for values that contains character digits, such as '16rFF', it returns incorrect results. The code fails to understand the characters and defaults to handling the String as a Float, returning the radix portion only (in this case, 1.6000000000000000E+01)

Workaround:

Use Integer>>fromString:

Bug 30820 - Unused temporary variable result in incorrect instance variable assignment

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.x, 5.1.5.1, 5.1.5, 5.1.4

Platform All

In a method of the form:

```
 keyword: aValue
   | unusedTemporary |
  instanceVariable := aValue
```

If a temporary variable is declared, but not referenced, and there is no further code in the method, the instance variable assignment will not occur correctly. This is due to an error in the byte codes that the compiler generates for the unused temporary variable.

Workaround:

Do not leave declarations of unused temporary varables in methods; in any case, unused temporary variables waste resources.

Bug 30710 - ProcessorTime is always 0

Product: GemStone/S

Versions: 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.1.5, 6.1.4, 6.1.3, 6.1.2

Platform Windows

The stone/gem statistic ProcessorTime is not recorded on Windows 2000 or XP, and will always be 0.

Bug 30202 - GsFile doesn't use 64 bit file I/O

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.2, 6.0.1, 6.0

Platform Unix

The GemStone Smalltalk class GsFile does not use 64 bit system file I/O, so GsFile operations on files larger than 2 GB will fail.

In most cases, the failure will manifest itself as methods returning nil, such as when trying to open or get the status of a file (GsFile>>openReadOnServer:, GsFile>>sizeOf: or GsFile>>exists:onServer:).

On Linux, trying to write a file beyond the 2 GB boundary will cause the gem process to dump core.

This bug only applies to GsFile. GemStone properly handles large database files (tranlog, extent and backup files).

Workaround:

Use System(C)>>performOnServer: to invoke operating system shell commands to check status of large files. Do not attempt to write large files with GsFile.

Bug 29855 - Tranlog sequences can contain "Fork-In-Time"

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.x, 5.1.5.1, 5.1.5, 5.1.4

Platform All

GemStone/S is quite flexible in allowing customers to restore backup files and replay tranlogs into an existing system that has a pre-established tranlog sequence. But this flexibility can cause problems when attempting to restore from a chronologically earlier backup file and then replaying tranlogs that encompass the results of the prior restore.

Here's an example sequence (there are many others):

Note: t<n> indicates repository events generating tranlog<n>.dbf

1. generate backup1
2. generate t1, t2, t3
3. generate backup2
4. generate t4, t5, t6
5. restore backup2
6. commitRestore (without replaying tranlogs t4, t5, t6)
   (note at this time, DB is at same state as step 3)
7. generate t7, t8, t9
8. restore backup1
9. replay tranlogs t1 through t9

Note that in terms of the repository lifecycle, there are really two time-lines here:

t1, t2, t3, t4, t5, t6
t1, t2, t3, t7, t8, t9

with a fork-in-time produced at the end of t3. During step 9 the replay of (t7, t8, t9) is likely to produce problems (described below).

Note that if at step 5 we also restored (t4, t5, t6), then the resulting sequence *would* be replayable without problems. It's when you break the continuity of the tranlog chain that difficulties arise.

Also note that after restoring backup1 in step 8, we *could* safely replay t1 through t6 without problems, but that changes made in (t7, t8, t9) would be lost.

Currently GS/S doesn't have any way of internally distinquishing the fork. When replaying the complete sequence, object changes made in (t4, t5, t6) may be logically inconsistent with those made in (t7, t8, t9). Possible errors are wide-ranging, usually with hard failures during the tranlog replay reported in the stone log. These may include:

1. UTL_ASSERT/UTL_GUARANTEE errors

2. Errors of the form:

```
recovery/restore: invalid operation XXXXXXXXXX
Transaction expected to abort.
```

non-empty invalidObjs in recover.c:commitTran

(Note that these error messages are not limited to fork-in-time problems
and may not necessarily indicate this bug)

In the worst case, errors may not show during tranlog replay, but the final repository may be corrupted in obscure ways. If the corruption is structural, it may be detected by an objectAudit. Otherwise, the corruption may go undetected unless picked up by application code.

This problem scenario usually occurs when a customer has a problem with their most recent backup file and is forced to restore from an earlier backup.

Workaround:

1. Be aware of the fork-in-time phenonemon and avoid restore/replay operations that would create a fork.
2. When restoring into an on-going tranlog sequence, only restore a backup file generated earlier within that same sequence, and then replay *all* tranlogs in that sequence generated since that backup.
3. If for some reason you can't follow guideline 2, then realize that you will not be able to restore from an earlier backup and replay tranlogs beyond the point of the initially restored backup.

To simplify the explanation of the fork-in-time problem, the above description placed the fork-in-time between tranlogs. In practice, the fork-in-time can occur anywhere within a tranlog, depending upon when the associated backup file was generated. So no, you *can't* just delete and rename tranlog files to get around this problem ;-)

Bug 29854 - Cannot backup/restore to tape on Linux

Product: GemStone/S

Versions: 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.4, 6.1.3, 6.1.2, 6.1.1, 6.1

Platform Linux

Gemstone/S executables do not recognize Linux tape devices, and therefore backups to and restores from tape on Linux do not work.

Workaround:

perform fullBackupTo: to disk, then tar this backup to the tape. To restore, tar from tape to disk and perform restoreFromBackup: on the disk version.

Bug 29830 - SIGUSR1 stack print mechanism fails during logout

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.2, 6.0.1, 6.0

Platform All

Since GemStone/S 6.0, sending "kill -16" (SIGUSR1) to a topaz/gem process

will cause that process to print its smalltalk stack to stdout. On Solaris,
this feature has been extended to also print the C-level stack to stdout.

But when the topaz/gem process is logging out, using this mechanism can
cause a core dump, preceeded by the warning message:

```
  PomOtGetSharedObjPage: root not loaded
```

Workaround:

Don't use the SIGUSR1 signal on a gem/topaz process when it's logging out.

Bug 29752 - Cross-platform pageaudit has false-positive errors.

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.4, 6.1.3, 6.1.2, 6.1.1

Platform All

Extent files have a particular byte ordering based on the native byte order for the host they are on. If pageaudit is executed on a host with the reverse byte order from the byte order of the extents, pageaudit will return a success code, but incorrectly report repository read/write errors.

Bug 29670 - Specification of remote replicate tranlogs may inadvertently succeed

Product: GemStone/S

Versions: 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.2, 6.0.1, 6.0

Platform All

Fixed In: n/a

Specification of remote extents, remote replicate extents, remote tranlog directories, and remote replicate tranlog directories is not supported in releases 6.0 and later. In general, attempts to specify a unsupported remote file will fail on stone startup, with an appropriate error message written to the stone's log. However, specification of remote replicate tranlog directories may succeed with a warning message. Manually ensure your system configuation file does not specify remote replicate tranlog directories.

Bug 29627 - Nonblocking GCI function may SIGSEGV on Linux

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.4, 6.1.3, 6.1.2, 6.1.1

Platform Linux

A GCI non blocking function, called from a user action that is
loaded into a gem with an rpc application, may SIGSEGV on Linux.

Bug 29617 - restoreReclaimPages primitive errors reported incorrectly

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.x, 5.1.5.1

The call to the method SystemRepository>>restoreReclaimPages
ultimately invokes a primitive. If the primitive call fails, for
example if the repository is not in restore mode, or there was
not a transaction log previously restored in that session, the
code that handles the primitive failure is incorrect and will
report the wrong error, such as rtErrArgOutOfRange.

Bug 29578 - #transactionless mode incorrect from System(c)>>cacheStatistics:

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.4, 6.1.3, 6.1.2, 6.1.1, 6.1

Platform All

In transactionless mode, the method

```
  ((System cacheStatistics: System myCacheProcessSlot) at: 95)
```

returns an incorrect value, 4294967295, instead of -1. The value
is correct when viewed in statmonitor data.

Bug 29520 - Stone will not start with null primary/valid replicate tranlog directories

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.4, 6.1.3, 6.1.2, 6.1.1

Impact: Informational

If the tranlog directories are set to /dev/null, but there are
valid directories set for replicate tranlogs, the stone will not
start.

Workaround:

Do not attempt to use replicate tranlogs without primary tranlogs

Bug 29485 - Language file compilation does not handle % as documented

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.4, 6.1.3, 6.1.2, 6.1.1, 6.1

Platform All

When using the GemStone error message compiler in the process of
translating error messages, the %% is documented, in the SAG pG-7,
to print the % character.

This does not work correctly; after compiling, when the message
is invoked, the resulting error message will not include the %,
and the character following the %% in the source string will be
missing.

Workaround:

Instead of using %% for a percent sign, use \%\%

Bug 29242 - VSD may fail to start with Tcl 8.3 installed

Product: GemStone/S

Versions: 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.4, 6.1.3, 6.1.2, 6.1.1, 6.1

Platform Unix

If Tcl 8.3 is installed on your machine, VSD may fail to start with
the following output:

```
  Application initialization failed: Can't find a usable Tix Init.tcl
  in the following directories:
```

(Followed by a list of directories.)

Workaround:

Create a symbolic link under $GEMSTONE that points to the correct Tix
libraries shipped under $GEMSTONE/lib, e.g.:

```
% ln -s $GEMSTONE/lib/tix8.1 $GEMSTONE/library
```

Bug 29095 - In tranlog full condition, sessions may timeout and be killed

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.x, 5.1.5.1, 5.1.5

Platform All

During a tranlog full condition, most sessions will eventually hang waiting for a response from the stone to an abort or commit. While the sessions are waiting, the stone determines that it has not contacted the stone for more than the STN_GEM_TIMEOUT, and terminates the session.

Workaround:

Do not allow the tranlog space to become full. If STN_GEM_TIMEOUT is set to 0, the bug will not have an effect.

Bug 28821 - Statically linked custom gems broken on Linux

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.4, 6.1.3, 6.1.2, 6.1.1, 6.1

Platform Linux

Statically linked custom gems do not work correctly on Linux and may encounter a SIGSEGV error when run. This may occur even though the compile and link steps report no error.

Workaround:

No workaround

Bug 28789 - Inherited NRS value may cause remote SPC startup problems

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.x, 5.1.5.1, 5.1.5, 5.1.4

Platform All

When starting a remote shared page cache, some network resource string (NRS) values used on the remote host are inherited from the stone. This can be problematic if the #dir or #log NRS modifiers are specified in the stone's environment because these directives specify directory paths which may not exist on the remote host. If this occurs, the remote shared page cache may not start and the error message provided may not indicate the problem.

Workaround:

To avoid this bug, ensure the #dir and #log directives are not specified in the stone's environment. Alternatively, you can create the directories specified in the #dir and #log NSR strings on the remote host.

Bug 28728 - Some system-level statistics are incorrect on Windows

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.1.5, 6.1.4, 6.1.3, 6.1.2, 6.1.1, 6.1

Platform Windows

On Windows, the following statistics collected by statmonitor on "NtSystem" may erroneously always be zero:
    TotalInterrupts
    TotalInterruptTime
    TotalPrivilegedTime
    TotalUserTime
    TotalAPCBypasses
    TotalDPCBypasses

On Windows, the following statistics collected by statmonitor on "NtMemory" and for all "processes" may have values smaller than they should be on systems with more than 4G of memory. This is because only the lower 32 bits of the value are recorded, so any of the upper 32 bits are lost.
    AvailableBytes
    CacheBytes
    CacheBytesPeak
    CommittedBytes
    CommittedBytesInUse
    PageFileBytes
    PageFileBytesPeak
    PoolNonpagedBytes
    PoolPagedBytes
    PoolPagedResidentBytes
    PrivateBytes
    SystemCacheResidentBytes
    SystemCodeResidentBytes
    SystemCodeTotalBytes

SystemDriverResidentBytes
SystemDriverTotalBytes
TotalPoolNonpagedBytes
TotalPoolPagedBytes
ResidentBytes
VirtualBytes
VirtualBytesPeak
WorkingSet
WorkingSetPeak

Bug 28546 - Gems with static user action code die with SIGILL

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.4, 6.1.3, 6.1.2, 6.1.1, 6.1

Platform AIX

Gem executables statically liked with user action code (as described in the GemBuilder for C manual, Appendix C) die with SIGILL, illegal instruction.

Workaround:

Use dynamically linked user action libraries, as described in the GemBuilder for C manual, Chapter 3.

Bug 28330 - Gems can crash with page cache fault during stone shutdown

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.x, 5.1.5.1, 5.1.5, 5.1.4

Platform All

Under certain conditions, when the stone is shutdown immediately with other sessions running (stopstone -I), there is a small possibility that the currently running sessions may fail with a page cache fault, followed by the expected #gsErrStnShutdown error (4057).

Since the stone is being shutdown anyway, this page cache fault is benign and will not corrupt the DB. The error messages in the log will indicate a "Bitmap page fault: leaf page <X> kind <K>.".

Workaround:

You can safely ignore this error message.

Bug 28208 - #rtErrLostOtHandlingFailed while in a transaction

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.x, 5.1.5.1, 5.1.5, 5.1.4, earlier versions

Platform All

Under extreamly rare conditions, a gem that has just entered a transaction will receive a LostOt signal sent from the stone while it was still outside of a transaction. The gem cannot properly handle the LostOt signal under these conditions and will fail with a #rtErrLostOtHandlingFailed error (GemStone error 2380).

If this error occurs at the end of a MarkForCollection, it is usually benign. You can check statmonitor/VSD data and GcGem logs to determine if the MFC completed successfully or not.

Workaround:

None. You should treat this as a normal LostOt event, and in GBS, logout and log back in to resynchronize the GBS caches.

Bug 28103 - Tranlog full condition may cause stone to crash

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.2, 6.0.1, 6.0

Platform All

Under some specific timing conditions, a tranlog full condition can cause the stone to crash with a fatal stone error. While normally the system is designed to pause when tranlogs are full, to allow space to be cleared or new tranlog directories to be added, there
are some points in the processing of a commit or other transaction log entry where the code can not safely pause. To preserve tranlog integrity, the stone shuts down. Since the tranlogs are full, it is not possible to shut down cleanly.

Assuming tranlog space is made available, you can safely restart the stone and allow it to recover. There is no specific risk associated with this bug, however, there are other bugs related to tranlog full conditions, some of which do cause corruption; see bugnotes for 30034 and 29024. Other bugs associated with tranlog full conditions are 28057, 29095, and 28082.

Workaround:

None. Avoid tranlog full conditions; a number of problems have been reported under this circumstance.

Bug 27908 - Set-valued indexes are known to have problems

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.x, 5.1.5.1, 5.1.5, 5.1.4

Avoid creating set-valued indexes (indexes with a * in the path). Set valued indexes are known in practice to be less reliable than standard indexes. In addition, due to the nature of these indexes, there may be excessive volume of indexing infrastructure created, and the same functionality will often have similar performance if the code is refactored to use standard indexes.

GemStone may choose to deprecate this functionality at some future time.

Workaround:

If you refer to section 5.6 in the 6.0 GS Programming Guide (page 5-28 in the 6.x version, 5-31 in the 5.x version), there are two examples of using set-valued indexes. Taking the first one, finding all employees with children under age 18:

```
  AllEmployees select:  { :emp  |  emp.children.*.age <= 18 }
```

This is equivalent to the code:

```
 | results |
 results := AllEmployees class new.
 AllEmployees do: [ :anEmployee |
     anEmployee.children do: [ :aChild |
         (aChild.age <= 18) ifTrue: [results add: anEmployee]]].
 results
```

The set-valued query (on emp.children.*.age) on can be used without creating a set valued index.

Note that, due to the semantics of select over set-valued indexes, the resulting collection may contain multiple instances of the same Employee (if, for example, the Employee has more than one child under 18). The resulting collection will contain one element for each occasion the predicate evaluates to true, which can mean the resulting collection is larger than the original collection.

While using the set-valued query, you might expect that creating an index of the form:

```
  AllEmployees createEqualityIndexOn: 'children.*.age'
```

would make the query faster, in actual practice this may or may not be the case, and excessive overhead objects may be generated.

If you wish to use indexes for this query, avoiding set-valued indexes, you can refactor by putting all of the children objects into a separate collection, called AllChildren. The index then goes on AllChildren. You would need to modify code for adding/deleting children to a particular employee so that this collection is also kept in synch. You also need to add a back pointer from the children object to point to the owning parent object.

Here is how that query might be written:

```
(AllChildren select: { :child  |  child.age <= 18 })
   collect: [ :child  |  child parent ]
```

or perhaps more efficiently:

```
 | result |
 result := AllChildren speciesForSelect new.
 (AllChildren select: { :child | child.age <= 18 }) do: [:child |
   result add: child parent ].
 result
```

Bug 27660 - continueTransaction after password change fails

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.x, 5.1.5.1, 5.1.5, 5.1.4

Platform All

Using continueTransaction after changing your password will result
in a read authorization failure. For example:

```
topaz 1> run
 System myUserProfile oldPassword: 'hello' newPassword: 'goodbye'.
 System continueTransaction.
%
-----------------------------------------------------
GemStone: Error         Nonfatal
An attempt was made to read the object with ID 143689 in segment
aSegment, Number 5 in Repository SystemRepository
Owner SystemUser write
World none with insufficient authorization.
Error Category: [GemStone] Number: 2115 Arg Count: 2
Arg 1: 143689
Arg 2: a Segment
 itsRepository   a Repository
 itsOwner        an UserProfile
 groupsRead      nil
 groupsWrite     nil
 ownerAuthorization 2
 worldAuthorization 0
 spare1          nil
topaz 1>
```

Workaround:

Use commitTransaction after changing the password.

Bug 27482 - Cannot run pageaudit from root directory on Windows

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.1.5, 6.1.x, 6.0.1, 6.0

Platform Windows

On Windows, attempting to run a pageaudit from a root directory (for example, c:\) results in an error stating that the directory name is invalid.

Workaround:

Run pageaudit from a subdirectory.

Bug 27338 - Missing tranlogs are not detected during restore

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.x, 5.1.5.1, 5.1.5, 5.1.4, 5.1.3.1, 5.1.3

Platform All

When executing "SystemRepository restoreFromCurrentLogs" to restore tranlogs from the directory specified by the configuration parameter STN_TRAN_LOG_DIRECTORIES, if a tranlog is missing from the middle of the sequence, the system will not detect this and restore only up to the missing tranlog. If the system is then put into use via "SystemRepository commitRestore", it will create a "fork in time" which will result in lost data.

For example, imagine you are restoring a system that includes tranlogs T1, T2, T3, T4, T5, and T6. Say that T3 is missing. After committing the restore, the system will only contain data up to T2. Now assume that new transactions are then committed which will go into tranlogs T7 and T8 before the mistake is discovered. At this point, your only options are:

1. Continue with the system as-is, loosing data in T3 through T6.
2. Assuming T3 is recovered, you can restore the backup and replay tranlogs T1 through T6, loosing data in T7 and T8. Since T7 and T8 were generated from a different "context" (T1 through T2) they cannot be replayed in a context that includes T3 through T6.

Workaround:

Make sure during tranlog restore that the tranlog directory contains all tranlogs needed. Also, you should review the stone log to determine which tranlogs were actually loaded and confirm that this is correct before committing the restore and using the system.

Bug 27279 - Limit on number of netldi spawned process on Windows

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.1.5, 6.1.4, 6.1.2, 6.1.1, 6.1, 6.0

Platform Windows

On Windows 2000 and Windows XP with a default application desktop heap, there is a limit on the number of processes a netldi can spawn. This has the effect of limiting the number of rpc logins, since the netldi must spawn a gem for each rpc login. Other factors that limit the number of gems is if the netldi process is owned by LocalSystem, or the gemnetobject script is used to spawn gems.

After this limit has been reached, logins will fail with GemStone error 4042 and a message similar to this:

```
 GemStone: Error        Fatal
 Unable to create a GemStone session.
```

```
  Netldi 'netldi60' on host 'donkey' reports the request 'gemnetobject'
failed: Your netldi request timed out after 30 seconds.
  If a gem or page server log file was created for this request it will
contain information on why this request failed.
  Otherwise check the netldi log 'c:/GemStone60/product/log/netldi60.log'
for information.
  Error Category: [GemStone] Number: 4042 Arg Count: 0
```

There may also be a popup dialog on the server machine stating "The application failed to initialize properly (0xc0000142). Click ok to terminate the application".

Note that GemStone changes in version 6.0.1 avoiding this problem. However, due to side effects, the changes were backed out in GemStone/S 6.1.

Workaround:

Limited desktop heap is a known cause of this problem (there are other factors). See

```
   http://support.microsoft.com/default.aspx?scid=kb;en-us;Q126962
```

This can be fixed by increasing the size of the desktop heap. This requires using regedit and rebooting. Contact your system administrators or backup your registry and use extreme caution in editing your registry as you can make your machine unusable.

The value of the registry setting

```
    HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\Session Manager\SubSystems\Windows
```

is a long string that contains the substring "SharedSection", which may look similar to

```
    SharedSection=1024,3072,512
```

the final parameter (in this example 512), if provided, specifies the desktop heap assigned to non-interactive sessions. Increasing this to a larger value, such as 1024, should provide enough heap to allow a much greater number of logins.

Bug 27268 - Interrupted recovery can lead to page audit errors

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.2, 6.0.1, 6.0

Platform All

If you kill the stone during system recovery (while tranlogs are being replayed to restore transactions committed since the last checkpoint), and then restart the stone again and allow it to sucessfully recover, the resulting system may generate the following error messages during page audits:

```
   The number of free pages (1234) does not agree with audit results (1212).
   Audit and Repository disagree on Page #12345
      Audit indicates that it is _not_ allocated.
      Found pagekind = 4
```

```
  ...
```

This indicates that there are pages in the repository which are not currently used, but which are not properly listed on the free page list. These pages are effectively no longer available to the system and waste space, but otherwise cause no harm.

To clear the condition, you can either:

1. Restore from a backup to a clean extent and replay tranlogs.

2. Generate a new backup from the corrupted system, then restore the backup to a clean extent (required if STN_TRAN_FULL_LOGGING = FALSE).

Workaround:

No workaround

Bug 27227 - Topaz output suppressed while executing from .topazini

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.x, 5.1.5.1, 5.1.5, 5.1.4, 5.1.3

Platform All

If you execute commands, such as login, within the .topazini file, output is not displayed. This includes login error messages.

Workaround:

In the .topazini file, add the following:

```
  output push /dev/tty
  <commands>
  output pop
```

Bug 27201 - On Windows, topaz -l doesn't clean up after forced termination

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.1.5, 6.1.x, 6.0.x, 5.1.5.1, 5.1.5, 5.1.4

Platform Windows

If a linked topaz session is forcibly terminated, either by another session or by doing System>>shutDown, it will fail to properly release its shared memory resources. This can prevent stone startup or future logins from that invocation of topaz.

Workaround:

Exit topaz after a session has been terminated.

Bug 27091 - Some statmonitor statistics not updated for remote shared page cache

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.2, 6.0.1, 6.0

Platform All

The following gem statistics are not updated if the gem is
running on a remote shared page cache:
* LostOtsSent
* SigAbortCount
* SigAbortsSent

Bug 27089 - OS network drives can confuse GemStone local/remote status

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.2, 6.0.1, 6.0

Platform All

Impact: Critical

As of 6.0, GemStone no longer supports hosting extents, replicates, or tranlogs on disk drives which are remote from the local machine running the stone process. Running with remote drives significantly increases the complexity of the system and reduces performance over a system using local drives.

GemStone normally detects remote drives, and will disallow any operation attempting to use a remote drive for an extent, replicate, or tranlog. While GemStone currently does a good job in detecting remote drives, unanticipated configurations and future enhancements from OS vendors in the handling of network drives may potentially confuse GemStone into thinking that a remote drive is instead local, and allow the system to run with remote extents, replicates, and/or tranlogs. Such configurations have not been QA tested by GemStone Systems, and are unsupported. Running with such a configuration could result in system crashes and corrupted data.

Workaround:

Be careful when configuring your system and make sure that all disk drives used for extents, replicates, and tranlogs *really* are local to the machine running the stone process.

Bug 27082 - Repository >> restoreStatusNextFileId not aware of final fileId

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.2, 6.0.1, 6.0

Impact: Informational

When the stone is in restore mode, the method
  Repository>>restoreStatusNextFileId

should report the ID number of the next transaction log file to be restored. However, after all the transaction logs have been restored, it will erroneously return the next ID in the sequence, although a transaction log file with this ID does not exist.

Workaround:

Do not rely on this method to determine when all transaction logs have been restored.

Bug 27065 - Recovery fails with unused extents

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.2, 6.0.1, 6.0

Platform All

When extents are added programmatically, the configuration file is updated immediately, but the change is not recorded until the next checkpoint. If there is an unexpected shutdown occurs before the next checkpoint, the stone recovery may fail with the error message:

File already exists; you must delete it before recovery can succeed.

Workaround:

Removed the recently added extent file/s. As long as the transaction logs are available, any data in these extents will be restored by the recovery process.

Bug 26396 - copydbf of a compressed gzip file fails

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.2, 6.0.1, 6.0

Platform All

Copydbf of a compressed gzip file fails with the following error message:

```
  unix> copydbf extent0.dbf.gz  extent0.dbf

  GemStone is unable to open the file extent0.dbf.gz.
  No disk I/O error information is available.
```

Workaround:

Uncompress the file using gzip first.

Bug 26117 - GsFile class>>contentsAndTypesOfDirectory:onClient: reports files as directories

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.x, 5.1.5.1, 5.1.5, 5.1.4

Platform All

The return value of GsFile class>>contentsAndTypesOfDirectory:onClient:
is an array of values with the contents of the directory, and,
for each, a boolean indicating if this item is a file. Files
in the directory that have restricted permissions are reported
false, which incorrectly implies that they are directories.

Bug 25456 - Compressed file names without .gz appended

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.x, 5.1.5.1, 5.1.5, 5.1.4

If the argument to:
  fullBackupCompressedTo:
  continueFullBackupCompressedTo:
  entire destination specification of 'copydbf -C'

has a string length of 3 or fewer characters, the '.gz' is not appended
to the compressed file that is created.

Also note that the arguments to these methods must be file system files,
not directories. While directories may work in some cases, it is not a
tested or supported use. One anomality is that the '.gz' is not appended
to the created file.

Bug 25124 - fullBackupCompressedTo: fails for raw character partitions

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.2, 6.0.1, 6.0

Platform All

"SystemRepository fullBackupCompressedTo:" will fail with the following
error message when applied to a raw character partition:

```
  topaz> run
  SystemRepository fullBackupCompressedTo: '/dev/rchardevice'
  %
  -----------------------------------------------------
  GemStone: Error        Nonfatal
  An attempt to write to file '/dev/rchardevice' failed because
    '[rdbfio.gzclose !TCP@myhost#dbf!/dev/rchardevice errno=22,
    EINVAL, Invalid ... .This error terminates the backup and the backup
    file is not usable.
  Error Category: [GemStone] Number: 2305 Arg Count: 2
  Arg 1: /dev/rchardevice
  Arg 2: [rdbfio.gzclose !TCP@myhost#dbf!/dev/rchardevice errno=22,
    EINVAL, Invalid argument (programmer error)]
```

After hitting this error, the system will still think that a backup
is in progress. You will need to execute:

```
  SystemRepository abortFullBackup
```

to clear the confusion.

Workaround:

Use the equivalent block device name for the chosen raw partition.

Bug 24961 - Increasing #mfcGcPageBufSize can have adverse effects

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.x, 5.1.5.1, 5.1.5

Platform All

Impact: Informational

The release notes for GS/S 5.1.5 mention that increasing the size of the markForCollection mark/sweep buffer can improve MFC performance. This is done by executing:

```
   (UserGlobals at: #mfcGcPageBufSize put: <NewBufferSize>)
```

in the session performing the markForCollection. However, in environments where the MFC session is subject to sigAborts, increasing this parameter to too high a value can instead result in poor sigAbort response times and degraded MFC performance.

If you perform markForCollection while other sessions are busy committing transactions, and the MFC session is therefore subject to periodic sigAborts, then excessively increasing the size of the mark/sweep buffer can result in the following problems:

1. Degraded sigAbort response time for the MFC session. While gathering a batch of pages for mark/sweep processing, the session cannot respond to a sigAbort. If the value of #mfcGcPageBufSize is too high, the delay for processing the sigAbort can result in the system suffering from a CR backlog, or in GemStone versions prior to 6.0.1, the MFC session failing with a lostOTRoot error.
2. Poor MFC performance. After gathering a batch of pages, the next step processes each page, marking and sweeping the objects on that page. This step *is* sensitive to sigAborts, and will cancel all further processing of subsequent pages in this batch upon receipt of a sigAbort. The remaining pages will have to be re-gathered in the next batch. If the sigAbort was received during the page gathering phase, then *all* the work for this batch is abandoned and the pages will need to be re-gathered in the next batch. This can result in the MFC session appearing to hang (the cache statistic ProgressCount does not change for an extended period of time) for as long as sigAborts keep interrupting the session.

Unfortunately, there is no easy way to determine the maximum safe value for #mfcGcPageBufSize, as it will depend on your site

configuration and transaction loads. We suggest you start with
a value of 500 (the default setting is 320), and gradually increase
it over subsequent MFC's, using VSD/statmonitor to monitor sigAbort
behavior (SigAbortCount and AbortCount) and MFC performance
(ProgressCount).

Bug 24015 - AllSymbols collection is not scalable

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.x, 5.1.5.1, 5.1.5, 5.1.4

Platform All

Customers with AllSymbols dictionaries larger than half a million objects
may, under certain circumstances, experience performance problems. Symptoms
can include:

* occasional extremely long login times shortly after the Stone has been
restarted,

or

* the GcGem can periodically consume a large proportion of CPU cycles
and space in the shared page cache

(or both).

Workaround:

You can avoid these problems with some combination of the following five
strategies:

* Minimize the use of symbols in your application. Symbols are appropriate
when:

  -- You need the canonical behavior.
  -- Use of symbols can significantly avoid redundant Strings.

* Avoid creating temporary symbols. GemStone does not automatically remove
symbols from AllSymbols when they are no longer referenced by any other
object. If you create a symbol, use it once, and discard it, the symbol
will continue to use space in AllSymbols until you explicitly remove it.

* Remove unused symbols.
  1. Make sure that the unused symbol is really not used, or you could
create a noncanonical symbol that is equal, but not identical, to one in
AllSymbols, thus causing symbol lookup failures.

  2. Execute:
      AllSymbols remove: #myUnusedSymbol
  (replacing myUnusedSymbol with the actual symbol name).

* Minimize the number of possible collision buckets used by AllSymbols.
Fewer, larger collision buckets may increase the cost of individual symbol

lookups, but significantly decrease the cost of the GcGem background operation and improve total system performance. A good rule of thumb is to allow up to an average of 1000 symbols per collision bucket. The following code rebuilds AllSymbols if necessary according to this guideline.

```
 | numSymbols numCollisionBuckets defaultSize |
defaultSize := 1010.
numSymbols := AllSymbols size.
numCollisionBuckets := ( AllSymbols size // 1000 ) max: defaultSize.
((AllSymbols basicSize // 2 - 2) ~= numCollisionBuckets ) ifTrue: [
   AllSymbols rebuildTable: numCollisionBuckets ].
```

* Recluster AllSymbols periodically to keep AllSymbols and its collision buckets on as few a number of data pages as possible.

Execute:

```
     System clusterAllSymbols
```

Bug 23975 - GCI user actions may have incorrect results following a full backup

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.x, 5.1.5.1, 5.1.5

Platform All

Running GCI user actions in the same session as previously running a full backup may cause the user actions to return incorrect results.

Workaround:

Use a separate session for backing up the repository, than is used for running application code with GCI user actions.

Bug 23746 - Object audit can inappropriately re-enable logins

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.x, 5.1.5.1, 5.1.5, 5.1.4

When an object audit starts, if logins have been disabled but sessions have not yet shut down, then logins are incorrectly reenabled after the audit completes.

For example, if the sequence of statements includes:

```
 System stopOtherSessions.
 SystemRepository objectAudit.
```

and one of the sessions being stopped takes a while to stop, then the object audit will not run in single-user mode and will re-enable logins when it completes.

Bug 23508 - Correction for poor performance using _asIdentityBag on large Bags/Set

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.x, 5.1.5.1, 5.1.5

Platform All

The method #_asIdentityBag is used in many places to convert an unordered collection into an IdentityBag before applying some operation against it. While this is a no-op for IdentityBags and IdentitySets (returning self), it involves doing a copy of elements from the internal dictionary when applied to Bags and Sets.

For very large collections, this adversely affects performance, and can create many temporary objects that must be garbage-collected.

This issue arises in the query operators #select:, #reject:, #detect:, and #detect:ifNone:, although there are other callers of #_asIdentityBag that might also benefit by recoding to avoid the extra copy operation.

Bug 23073 - SharedDependencyList inefficiencies may cause performance issues

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.x, 5.1.5.1, 5.1.5, 5.1.4

Platform All

GemStone uses an internal structure under Globals called SharedDependencyList (SDL), which is a hash dictionary using collision buckets containing dependency lists (DLs). DLs are used to track indexing and other structures that need to be updated whenever a particular object is modified.

The intention of the SDL is to reduce the total number of objects needed to represent this information by allowing objects with the same DL to share a common DL in the SDL, rather than each having its own DL. The actual number of DLs in the SDL depends on the amount of sharing possible, but in general, can be expressed as:

size(SDL) <= min(e, (2**n)-1)

where
  n = total number of indexed collections
  e = total number of elements contained in all indexed collections

There are certain cases where "extra" DLs may be left behind in the SDL. Under extreme conditions, these additional DLs may cause performance problems during index maintenance operations such as index creation and removal, or updating of an index due to a change to an object participating in the index. The following examples provide more information about these cases:

CASE 1

When you add an additional index to a collection, new DLs are generated for those objects participating in the index, but older DLs are left behind. This action repeats as additional indexes are added to a collection. Although this action is unlikely to cause performance issues unless you have hundreds of indexes on a single collection, it can cause the SDL to be larger than expected. If the index is ever deleted, ALL of the DLs are removed, so extra DLs never outlive their associated index collection.

CASE 2

If you have an index that has a String as a final path term, the DL generated also includes a reference count to track how many other objects in indexed collections reference the same String. As these objects add or remove references to the shared String, a new DL is generated for each unique reference count value. If this String is shared extensively, and the number of sharing objects varies widely over time, then a very large number of extra DLs could be generated, and may begin to affect performance. To avoid the possibility of creating these extra DLs, we recommend:

1. Avoid using shared Strings as path terms in indexed collections.
2. Make such Strings invariant. Since invariant objects don't change, the system does not need to track dependency list information.
3. In general, try to make objects in indexing paths invariant whenever possible.

Workaround:

Follow the guidelines suggested above to reduce the likelihood of creating large numbers of additional DLs.

If you find that you already have large numbers of DLs that are significantly affecting your performance, there are steps that can be taken to removed unneeded DLs. Contact GemStone Technical Support for more information.

Bug 22722 - Incomplete implementation of BinaryFloat (C)>> status

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.x, 5.1.5.1, 5.1.5, earlier versions

Platform All

The classmethod status returns an empty array in 5.x versions.
Because of this, the following actions do not perform properly:

Raised exceptions
Rounding mode settings
Enabled traps

Workaround:

No workaround at present. Status will be fully implemented in a future release of GemStone/S.

Bug 22542 - GBS sessions cannot respond to sigAborts when control passes outside Smalltalk

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.x, 5.1.5.1, 5.1.5

Platform All

GemBuilder sessions cannot respond to sigAborts when control passes outside the Smalltalk environment. Control leaves the Smalltalk environment when you execute userActions, socket communications, and other I/O. Unfortunately, not responding to sigAborts opens the possibility of lostOTRoots.

Workaround:

To avoid getting a lotOTRoot when performing file I/O, break the operation into 4K chunks, to ensure control returning to Smalltalk periodically. Another alternative is to work inside a transaction.

Bug 22170 - reclaimAll can fail immediately after markForCollection

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.x, 5.1.5.1, 5.1.5

Platform All

Impact: Informational

If you execute:
    Repository>>reclaimAll

immediately following a markForCollection (for example, in a topaz script) the reclaim can fail by returning false, or by returning true but not actually reclaiming any pages.

This occurs because, after running markForCollection, the following operations must occur:

1. The Gem running markForCollection sends the Stone the list of possibly dead objects and returns. The Stone now holds the list of possibly dead objects -- referred to as the "possible dead set" -- in RAM.

2. Now every Gem currently logged in the system must search the possible dead set for any objects to which it holds references. Then it must commit or abort,at which time it votes to either keep an object in the set, or remove it (if it holds a reference).

        -- NOTE --
    Without commiting or aborting, Gems do not vote and can delay the
process
    indefinitely. The vote cannot be finalized, garbage collection halts
at this point,
    and commit records accumulate.

3. But what about Gems that aren't on the system now, but were when garbage collection started? Their modified objects are in the commit record backlog, in the write sets of each commit record, which the GcGem reads in order to vote on their behalf.

Until these events occur, objects marked by markForCollection are not truly dead, only "possibly dead", and their pages cannot be reclaimed.

If you call stopOtherSessions after the markForCollection, you terminate the GcGem before it can vote on behalf of the logged-out Gems. Therefore, objects in the possible dead set are not promoted to DeadNotReclaimed, and the subsequent reclaimAll has nothing to reclaim.

Workaround:

You can tell when it's time to stop other sessions (if necessary) and run reclaimAll by using statmonitor and Visual Statistics Display (VSD) to watch three cache statistics:

When PossibleDeadSize and GcPossibleDeadSize fall to zero, and DeadNotReclaimedSize goes up, you can run reclaimAll confident that the operation will perform as intended.

Bug 21888 - System timeGmt95 and timeGmt2005 are off by 72 minutes, 48 seconds

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.x, 5.1.5.1, 5.1.5, 5.1.4, 5.1.3.1, 5.1.3

Platform All

The calculation for method "System timeGmt95", which is supposed to return the number of seconds since 1/1/95 00:00:00 GMT, in fact is off by 4368 seconds and really returns the number of seconds since 1/1/95 01:12:48 GMT.

Likewise, the calculation for method "System timeGmt2005", which is supposed to return the number of seconds since 1/1/2005 00:00:00 GMT, is off by 4368 seconds and returns the number of seconds since 1/1/2005 01:12:48 GMT.

Workaround:

If absolute time accuracy is required, add 4368 to the value returned. If the timestamps returned from these methods are only used for sorting data, then no correction is necessary.

Bug 21365 - startstone or login fail with misleading shmget() error

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.x, 5.1.5.1, 5.1.5, 5.1.4

Platform All

When STN_MAX_SESSIONS and keyfile maximum sessions are out of synch, startstone or login fails with the following misleading error message in the shrpcmon log:

```
  <SHRPCMON, shell for GemStone SharedPageCache Monitor>
  <Taking commands from command line.>
   [SpcMon trace]: ... cache creation failed ...
  |   GemStone could not retrieve the IPC identifier associated with the memory |
  |   key 1275488407.  shmget() error = errno=17, EEXIST, The file already exists.
  |                                                                   |
  GemStone could not retrieve the IPC identifier associated with the memory
  key 1275488407.  shmget() error = errno=43, EIDRM, Identifier removed
  (resource such as semaphore or shared memory has been deleted.
  GemStone could not attach to the shared page cache.
  The Shared Page Cache Monitor is shutting down.
  The Shared Page Cache Monitor is shutting down.
```

This can occur when a Gem's configuration file specifies a SHR_PAGE_CACHE_NUM_PROCS less than that of the Stone it is logging in to, or when a keyfile specifies a maximum number of users that is much less than the STN_MAX_SESSIONS of its configuration file.

Workaround:

Set STN_MAX_SESSIONS in the configuration file to reasonably close to (or the same as) the keyfile limit.

Bug 20341 - Temporary object audit errors after restore

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.x, 5.1.5.1, 5.1.5

Platform All

'SystemRepository objectAudit' may report errors of the form:

```
  Object 429737, of class 1541, at 1-based offset 2, references nonexistent
object 1335649
```

after full restore from backups and tranlogs on applications that use indexes on set valued instance variables.

Workaround:

Perform

```
  SystemRepository markForCollection.
```

followed by:

```
  SystemRepository objectAudit.
```

This will eliminate this error for subsequent object audits.

Bug 19449 - DateTime creation on boundaries of Daylight Savings Time

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.x, 5.1.5.1, 5.1.5, 5.1.4, 5.1.3

Platform All

Creation of local DateTimes at the start and end of DST are problematic.

Between the start date and time of DST and one hour later, there is no local time. Attempts to create a local DateTime in this non-existent range do not fail, they return the DateTime one hour earlier.

Between the end date and time of DST and one hour earlier, the local Time is ambiguous; there are two distinct DateTimes that are correct representations of the local time. The current creation mechanism returns the DateTime of the later of the two. Consequentially, there are certain GMT DateTimes (the earlier of the two for an ambiguous local time) that it is not possible to create using the DateTime local time creation methods

Workaround:

Use the DateTime gmt creation methods to avoid the inherent ambiguity of the local time at the DST boundaries.

Bug 19246 - "Audit errors" during object audit and/or page audit

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.x, 5.1.5.1, 5.1.5, 5.1.4, 5.1.3

Platform All

The following two messages may be written to stdout during a page audit or object audit.

If this message appears:

"Audit error - unable to complete disk read\n"

a gemstone error will be displayed:

"Page %ld is bad, referenced in the object table by oop %ld."

and the audit status will be recorded as failing.

If the following message appears:

"Audit error, disk read, mismatch on pageId %ld\n"

And there is no subsequent gemstone error message, then the message can be safely ignored

Bug 18525 - max values for gcGem parameters silently restricted to max SmallInteger values

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.x, 5.1.5.1, 5.1.5, 5.1.4, 5.1.3, 5.1.2

Platform All

All gcGem parameters that are extracted from the SymbolList are limited to SmallIntegers. If one of the values happens to be a LargePositiveInteger, the default value is used, but there is no warning in the log file.

Workaround:

When setting any gcGem parameters, make sure that the value is
less than or equal to SmallInteger|maximumValue

Bug 15521 - Date does not restrict year inputs

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.x, 5.1.5.1, 5.1.5, 5.1.4, 5.1.3.1, 5.1.3

Platform All

Although the supported range for Dates are dates after
December 31, 1901, Date objects are created without errors
for other input year values.

This means that when creating a date without specifying a
input format, which uses the default format, DD/MM/YYYY,
if you provide only a two-digit year, it will treat the 2
digits as the full value of the year, and will create a
date in the first century.

Bug 15520 - Stone hangs when repository is full

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.x, 5.1.5.1, older versions

Platform All

As described in Chapter 4 of the GemStone System Administration
Guide, the Stone attempts to prevent the repository from ever
filling completely. The final action it takes is to send a stop
session to the Gems with the oldest commit records. However, under
certain circumstances, the Gems are waiting for communications
from their clients, and ignore the Stone's request. The repository
then fills completely, causing the Stone to hang for lack of disk
space.

Workaround:

Try to ensure that Gems are not left in a state where they ignore
signals from the Stone. For example, don't leave topaz sessions
sitting at the topaz prompt waiting for instructions. Make sure
topaz is executing some code.

The owner of a Gem process can also logout the Gem manually, or,
if necessary, terminate the Gem manually.

Bug 15350 - Comparison of nil DateTime yields unexpected result

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.x, 5.1.5.1, 5.1.5, 5.1.x

Platform All

A nil DateTime is considered to be less than all valid values of DateTime. This might cause a comparison to yield nil when a DateTime is expected.

Bug 15190 - Network crash may prevent Gems from committing

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.x, 5.1.5.1, 5.1.5, 5.1.x

Platform All

If a Gem is holding the commit token when the network crashes, other Gems are prevented from committing.

Workaround:

Configure the network keepalive option on the Stone machine. Use low values for the settings so that the server machine checks often for a network crash.

Bug 15073 - New instances of UserProfile added to AllUsers, not to the receiver

Product: GemStone/S

Versions: 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.x, 5.1.5.1, 5.1.5, older versions

Platform All

The instance creation methods for UserProfileSet (all methods beginning with addNewUserWithId:) do not add the instance to the receiver, but to the UserProfileSet AllUsers. However, the method UserProfileSet >> add: does add the instance to the receiver.

Also, the method UserProfile class >> newWithUserId:... adds the instance to the UserProfileSet AllUsers, which means that both privilege and authorization are required to create a UserProfile.

Bug 14023 - GemStone does not like a space in its directory name

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.1.5, 6.1.x, 6.0.x, 5.1.5.1, 5.1.5, 5.1.4, 5.1.3, earlier versions

Platform Windows

A space character in the GemStone directory name causes problems in various places where the GEMSTONE environment variable is expanded, with the result that certain programs may not start or execute correctly.

Workaround:

Install GemStone in a directory that does not contain spaces in its path. For example, avoid the directory "Program Files".

Bug 13660 - Login with blank in stone name fails with wrong error

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.x, 5.1.5.1, 5.1.5, earlier versions

Platform All

If you specify the stone name in a way that appends a trailing blank to a name that is otherwise correct, the login attempt fails with an error message that the shared page cache could not be attached. For example, '!@mynode!gemserver51 '. The error message is misleading in suggesting that the cause may lie in the system configuration.

Workaround:

Remove the trailing blank.

Bug 13355 - listInstances results may include dead objects

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.x, 5.1.5.1, older versions

Platform All

The following sequence may cause instances to be committed that shouldn't have been:

```
 markForCollection
 listInstances on Foo
 commit
 objectAudit
```

In this case, markForCollection finds instances of Foo and records them as possibleDead. Because the markForCollection ends its operation with an abort, the session votes on the possibleDead (and doesn't remove any from possibleDead). The listInstances method (operating on the transaction view at the end of the markForCollection) finds those instances of Foo that the mark added to the possible dead and commits them. If the GcGem had completed its sweep of the possibleDeadWsUnion, the commit would have failed, because the object is placed in the exclusiveLock set when the sweep completes.

Workaround:

Avoid committing the result of listInstances or listReferences.

In a single user system, you can do a SystemRepository reclaimAll between the markForCollection and the call to listInstances. This sequence guarantees that the view of the repository doesn't contain any unreclaimed dead, meaning that only "live" instances will be found.

In a multiuser environment, monitor the deadNotReclaimed and wait until this goes to zero.

Bug 13083 - Stack breakpoint at end of method

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.x, 5.1.5.1, 5.1.5, earlier versions

Platform All

It is possible for a stack breakpoint to cause execution to stop at a point in a method that is beyond the last step point in the method. In this case, the topaz debugger displays a step point number that is one larger than the last step point in the method.

Workaround:

No workaround needed.

Bug 12981 - Hash values change on modification of collection elements

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.x, 5.1.5.1, 5.1.5, earlier versions

Platform All

The hash value of a collection depends on the elements contained in that collection. If an element of the collection is changed. or a new element is added, the hash value changes. As a result, if one of these collections is contained in another hash-based object, such as Dictionaries, (Equality) Bags, or (Equality) Sets, and an element in that object is changed or added, you may not be able to find the collection after the modification. For the same reason, if any Dictionary is added to itself as a key, you may not be able to find it afterwards, because its hash value will change as soon as it is added to itself.

Workaround:

No workaround is available.

Bug 12891 - Failed commits can give inconsistent view of data

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.x, 5.1.5.1, 5.1.5, earlier versions

Platform All

Impact: Critical

When a commit fails, your view of the repository is updated to include changes committed by other users since your transaction began. Your own changes are preserved. However, this action can generate a view in which different subnodes of a single multi-part object (such as an NSC) reflect different temporal views of the repository. Queries may return wrong answers, or code may behave strangely.

Workaround:

To avoid any of these possible inconsistencies, you should abort immediately after a failed commit or false continueTransaction. If you want to attempt a recommit after an abort, you must compute all of the necessary information before you begin the commit, or you may end up trying to commit inconsistent data on the next try.

Bug 12819 - IdentityDictionary passivate, activate returns unequal Dictionary

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.x, 5.1.5.1, 5.1.5, 5.1.4

Platform All

Impact: Informational

The Passivation - Activation cycle does not preserve the identity of objects. Because two IdentityDictionaries will not be equal if all their keys are not identical, the result (object) of activating an IdentityDictionary will not be "equal" to the object that was passivated.

Workaround:

No workaround is available.

Bug 12778 - Execution stack overflow error from Dictionary =

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.x, 5.1.5.1, 5.1.5, earlier versions

Platform All

Two dictionaries that are equal and that contain themselves as one of their elements (immediately or through an indirection) will cause infinite recursion when the #= equal message is sent to one dictionary with the other dictionary as the argument.

Workaround:

No workaround is available.

Bug 12729 - fullBackupTo succeeds to unintended location

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.x, 5.1.5.1, 5.1.5, earlier versions

Platform All

When using network resource strings with SystemRepository fullBackupTo, it is possible for the backup to succeed, but to write to an unintended location. Because a network resource string must contain two "!"s, an NRS that contains an incorrectly typed keyword may cause a backup to default to a file whose name includes the NRS string as typed.

Workaround:

Make sure your NRS syntax is correct and valid before starting a backup.

Bug 12685 - continueTransaction misleading in some cases involving rc objects

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.x, 5.1.5.1, 5.1.5, earlier versions

Platform All

Using continueTransaction with instances of reduced-conflict classes can, under certain circumstances, yield spurious commit failures. If you're updating an instance of a reduced-conflict class and you send the message "continueTransaction", the transaction may later fail as a result of an unintended interaction between these two features.

Workaround:

The work-around is to either use reduced-conflict classes or "continueTransaction", but not both.

Bug 11976 - Unclear error message when attempting to use GciUserActionShutdown incorrectly

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.x, 5.1.5.1, older versions

Platform All

Impact: Informational

If you attempt to use a GemBuilder for C function during GciUserActionShutdown, you will receive the following error message:

A GCI operation was requested before the current nonblocking call was completed.

It should be noted that GemBuilder for C should not be called from GciUserActionShutdown (see the GemBuilder for C manual for more information on this function). However, if it is, you will receive the above message.

Workaround:

No workaround.

Bug 11611 - Incorrect unresolved external references cause errors at run-time

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.x, 5.1.5.1, 5.1.5, earlier versions

Platform Unix

When you link a GemStone user-action shared library, be aware that incorrect unresolved external references can cause run-time errors.

Due to the way these libraries are set up and used, you must leave unresolved external references to GemBuilder for C functions in the finished library. These references are resolved when the user-action library is loaded by GemStone. However, if you misspell a GemBuilder for C function call, you may not discover it until run-time, when your process terminates with an unresolved external reference error.

Workaround:

Use a compiler that returns warnings for undeclared functions.

Bug 11247 - The gemsetup.sh and gemsetup.csh scripts set path incorrectly

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.x, 5.1.5.1, 5.1.5, earlier versions

Platform Unix

Gemsetup.sh and gemsetup.csh are both scripts which can be used to select the GemStone version. However, when you use these scripts, they put the GemStone version at the end of the current path. As a result, when you attempt to execute 'startstone', 'startnetldi', or 'topaz', you will get the old version instead of the new version.

Workaround:

Make sure there are no other GemStone installations in your path when you use gemsetup.

Bug 11202 - Do not copy tape as a test for a readable backup file

Product: GemStone/S

Versions: 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.x, 5.1.5.1, 5.1.5, older versions

Platform All

The System Administration Guide recommends the copydbf command as a means to verify that a backup file is readable. The command should work, and it does so for a backup file on disk. However, the command incorrectly returns an error if the backup file is on tape.

Workaround:

Do not use this method to test for readable backup files on tape.

Bug 10849 - waitstone -1 on stone performing restores does not respond

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.x, 5.1.5.1, 5.1.5, earlier versions

Platform All

No message displays to inform the user that waitstone -1 on a stone performing restores does not respond.

Workaround:

No workaround is available.

Bug 10665 - GemStone environment variables work differently for Windows services

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.1.5, 6.1.x, 6.0.x, 5.1.5.1, 5.1.5, 5.1.4, 5.1.3, earlier versions

Platform Windows

The Windows NT/2000/XP services mechanism does not detect changes to system environment variables until the system is rebooted. As a result, environment variables for services work differently than they do for other processes.

Windows services disregard any value of GEMSTONE environment variables. Instead, they define their own GEMSTONE environment variables based on their location on disk.

Workaround:

Use the -v option to explicitly define the GEMSTONE environment variable in the service's context. If the -v option is used to define GEMSTONE, that value will be used instead of the value based on the location. The -v switch can be entered as a startup parameter in the Services option of Control Panel or as part of the command line for a 'create' or 'start' operation.
For example:

stone create -v GEMSTONE_SYS_CONF=$GEMSTONE_SYS_CONF

Bug 10638 - Service may not start when GemStone resides on network drive

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.1.5, 6.1.x, 6.0.x, 5.1.5.1, 5.1.5, 5.1.4, 5.1.3, earlier versions

Platform Windows

A Stone or NetLDI service created to run as the default account LocalSystem may fail to start when the GemStone installation is on another machine accessed by way of a network drive. The symptom is an "Access is denied" error at startup.

In general, the account the service is running as must have "read" access to the GemStone installation directory tree. In addition, the NetLDI's account by default needs "change" access to the GEMSTONE\log directory; the stone's account by default needs "change" access to the GEMSTONE\data directory.

Workaround:

Create the NetLDI using the switch ' /u accountName' where accountName is a user who has network access to the machine where GemStone is installed. This user must also have the Windows NT advanced user right "Log on as a service" on the machine the service will run on.

Although a Stone can be run from a GemStone installation on a network drive, it will not be able to use the default database in GEMSTONE\data. This is because the database extents need to be on a disk that is local to the machine the Stone is running on.

Bug 9537 - Reject queries along path containing * return wrong result

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.x, 5.1.5.1, 5.1.5, 5.1.x, 5.0.x, 4.x

Platform All

When the message reject: is along a path containing a set-valued instance variable, such as reject: {:i|i.a.*.b=5}, the result returned is the set difference between the receiver and a select: using the same predicate. An element in the collection is not returned if any of the elements in the nested nonsequenceable collection (represented by *) satisfies the rejection predicate.

Workaround:

The workaround involves two steps. First, perform a select along the path using the "not" of the operator in the rejection predicate. Then, to the result of that select add all elements in the collection that have a nil anywhere along the path of the reject.

Bug 8425 - Read lock on temporary object prevents a commit from making it permanent

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.x, 5.1.5.1, 5.1.5, earlier versions

Platform All

A read lock on a temporary object prevents a commit from making the locked object permanent because making the object permanent is equivalent to writing the locked object. The session must write lock the object if it wants to lock it and modify it.

Workaround:

Write lock any objects that you want to modify. Alternatively, you can release the read lock on objects you want to make permanent and then commit.

Bug 8199 - BackupLog shows backups that failed with bad arg

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.x, 5.0

Platform All

BackupLog logs the date and time a backup started. However, if the backup method is given a bad argument, the BackupLog claims a backup was started, even though it wasn't.

Workaround:

No workaround is available.

Bug 4390 - String>>at:equals: is inconsistent for arg ''

Product: GemStone/S

Versions: 6.7.2.1, 6.7.2, 6.7.1, 6.7, 6.6.5, 6.6.4, 6.6.3.3, 6.6.3.2, 6.6.3, 6.6.2, 6.6.1, 6.6, 6.5.8, 6.5.7.5, 6.5.7, 6.5.6, 6.5.5, 6.5.4, 6.5.2, 6.5.1, 6.5, 6.3.1, 6.3, 6.2.x, 6.2, 6.1.6, 6.1.5, 6.1.x, 6.0.x, 5.1.5.1, 5.1.5, earlier versions

Platform All

The method String>>at:equals: produces inconsistent answers when the index is 1 and the argument is '' (empty string).

GemConnect

Bug 49546 - GemConnect 2.2.2 on GS/32 6.7.2 on AIX fails

Product: GemConnect

Versions: 2.2.2

Platform GS/32 6.7.2 on AIX only

Impact: Informational

Attempting to load the supplied GemConnect 2.2.2 library on GS/32 6.7.2 on AIX fails with the following error:

```
----------------------------------------------------
GemStone: Error Nonfatal
An attempt to load a user action library failed because: 'GciLoadUserActionLibrary could not load
$GEMSTONE/ualib/liboraapi222-32.so.
  Reason: dlopen($GEMSTONE/ualib/liboraapi222-32.so) failed because rtld: 0712-001 Symbol __gxx_personality_v0 was
referenced from module $GEMSTONE/ualib/liboraapi222-32
.
Error Category: [GemStone] Number: 2171 Arg Count: 1
Arg 1: GciLoadUserActionLibrary could not load $GEMSTONE/ualib/liboraapi222-32
```

Workaround:

Please contact GemTalk Technical Support for a replacement version of liboraapi222-32.so

Bug 49506 - GsRdbChangeNotifObj>>removeAllNotification uses obsolete protected mode

Product: GemConnect

Versions: 2.4

Platform All Platforms

GsRdbChangeNotifObj>>removeAllNotification uses an obsolete mechanism for going into a special "protected mode" using primitive number 901. Executing this method will result in the error:

```
----------------------------------------------------
GemStone: Error Nonfatal
a InternalError occurred (error 2083), Primitive number 901 does not exist in the virtual machine. Error Category: 231169
[GemStone] Number: 2083 Arg Count: 1 Context : 20 exception : 31255553
Arg 1:  901
topaz 1>
```

Workaround:

Contact GemStone Technical Support for a recompiled version of your GemConnect library that addresses this problem.

Bug 49195 - GemConnect library load fail on AIX version 7 and later

Product: GemConnect

Versions: 2.4

Platform AIX

Impact: Informational

On AIX version 7 and greater, you may experience the following error when loading the GemConnect library:

```
Error 2171 , a ImproperOperation occurred (error 2171), user action library load  failed, 'GciLoadUserActionLibrary failed,
dlopen failed, 0509-130 Symbol resolution failed for $ORACLE_HOME/lib/libons.so because:
        0509-136 Symbol CreateIoCompletionPort (number 93) is not exported from
                 dependent module /unix.
        0509-136 Symbol GetQueuedCompletionStatus (number 94) is not exported from
                 dependent module /unix.
        0509-136 Symbol ReadFile (number 95) is not exported from
                 dependent module /unix.
        0509-136 Symbol WriteFile (number 96) is not exported from
                 dependent module /unix.
        0509-022 Cannot load module $GEMSTONE/ualib/liboraapi24-643.so.
        0509-026 System error: Cannot run a file that does not have a valid format.
        0509-192 Examine .loader section symbols with the
                "dump -Tv" command.; Filename: $GEMSTONE/ualib/liboraapi24-643.so' ERROR: UNEXPECTED ERROR
```

Fix is to change the IOCP IBM AIX parameter from “defined” to “available” by executing the following as root:

```
 unix> mkdev -l iocp0
 Response should be: iocp0 Available
```

Bug 49155 - GemConnect 2.4 install on GS/64 3.6 and later fails with define error

Product: GemConnect

Versions: 2.4

Platform With GS64 v3.6

Impact: Critical

Due to a change in how topaz define statements are handled in GS/64 3.6 and later versions, installing GemConnect 2.4 on GS/64 3.6 fails with the following error in the gsoraapifilein24.log:

```
  topaz 1 +>
  topaz 1 +> define ErrorCount 0
  You may not modify that definition with the DEFINE command
  ERROR: WRONG VALUE
  topaz > exec iferr 1 : input pop
  topaz 1> exit
  Logging out session 1.
```

Workaround:

Comment out or delete line 26 from the file $GC/make/gsoraapi.gs which reads:

```
define ErrorCount 0
```

Bug 48599 - GemConnect 2.4 on Solaris/SPARC requires SUNW_1.22.7 (Solaris version 11)

Product: GemConnect

Versions: 2.4

Platform Solaris/SPARC

Impact: Informational

GemConnect 2.4 requires Oracle version 12 and on Solaris/SPARC, Oracle 12 requires libc.so version SUNW_1.22.7, available on Solaris version 11. Using an earlier version of Solaris will trigger the following error when attempting to load the liboraapi24-643.so library:

Error 2171 , a ImproperOperation occurred (error 2171), user action library load failed, 'GciLoadUserActionLibrary failed, dlopen failed, ld.so.1: topaz: fatal: libc.so.1: open failed: No such file or directory; Filename: $GEMSTONE/ualib/liboraapi24-643.so' (ImproperOperation)

Note that Solaris/SPARC is no longer a supported platform for GemStone/S 64 Bit.

Workaround:

Only use on Solaris/SPARC version 11 or later.

Bug 48495 - No support for CESU-8 character format

Product: GemConnect

Versions: 2.4, 2.3

Platform All Platforms

Impact: Informational

Oracle databases using "UTF8" as the character format for either NCHAR character fields or NCLOB objects are actually using CESU-8 (Compatibility Encoding Scheme for UTF-16: 8-Bit -- for details see https://www.unicode.org/reports/tr26/tr26-4.html). True UTF8 encoding is provided via the AL32UTF8 format.

GemConnect does not currently support CESU-8, and the use of the #UTF8 setting for ncharSet or nclobSet in this configuration will result in typeConversionErrors.

Bug 48479 - Uncontinuable error during GemConnect user actions

Product: GemConnect

Versions: 2.4, 2.3, 2.2.3, 2.2.2, 2.2.1, 2.2

Platform All Platforms

When using GemConnect in a multi-threaded environment where one or more of the other threads are performing socket operations (for example, while using Seaside), there can be rare failures of threads performing GemConnect user actions with an UncontinuableError (error 6011). There are a number of different detailed error messages, but these will usually mention: "would cross frame of C primitive, user action, or FFI call"

For example:

```
-----------------------------------------------------
GemStone: Error        Nonfatal
a UncontinuableError occurred (error 6011), Execution cannot be
continued, 'return from on:do or rescue block would cross frame of C primitive,
user action, or FFI call'
```

These rare errors occur when a socket operation timeout is signaled, and the GemStone VM happens to be executing GemConnect user action code that has made a call-back to GemStone smalltalk. This is most likely to happen during an #openCursor:* call, but could occur during any GemConnect user action.

Workaround:

Repeat the operation. This error is quite rare, and the chance that it would happen twice in quick succession is extremely remote.

Bug 48434 - Various NCLOB handling errors

Product: GemConnect

Versions: 2.3

Platform All Platforms

Fixed In: 2.4

GemConnect 2.3 has a number of related issues in the handling of NCLOB data, including:

* When reading a NCLOB using "aClob readAll: recommendedClass", results would sometimes not be reduced properly (for example, a NCLOB containing only ASCII characters could be returned as a QuadByteString instead of a String.

* Incorrect triggering of "Insufficient room in target buffer for conversion" error when room still available.

* Occasional triggering of the error:

```
ERROR 26 , a GsRdbError occurred (error 26), reason:internalError,
a GsOracleConnection, a UndefinedObject, a UndefinedObject,
Unexpected Oracle error, anArray (GsRdbError)
```

masking another Oracle error, making diagnosis difficult.

Bug 47879 - BINARY_FLOAT / BINARY_DOUBLE not supported

Product: GemConnect

Versions: 2.3, 2.2.3, 2.2.2, 2.2.1, 2.2, 2.1.1, 2.1, 2.0.1, 2.0

Platform All Platforms

Fixed In: 2.4

There is no support for Oracle data types BINARY_FLOAT and BINARY_DOUBLE. Attempts to read fields with these data types will return nil.

Workaround:

For reading, you can use the SQL"cast" function to cast the field as a float.

For example, if MyTable contains binary_float field BF and binary_double field BD, you can use the following statement in the openCursor:* call:

select cast(BF as float), cast(BD as float) from MyTable

Bug 47650 - Reading of CLOBS > 65K as LONG/LONGRAW can cause problems

Product: GemConnect

Versions: 2.4, 2.3

Platform All Platforms

Impact: Informational

This is related to bug 44784 (Reading/Writing BLOBS/CLOBS > 64K using default LONG/LONGRAW configuration will silently truncate).

On some Oracle and GsOracleConnection parameter configurations, attempts to read CLOBS > 64K using the default LONG/LONGRAW configuration may instead do one of:

1. Hang the gem.

2. Trigger an Oracle ORA-03106: fatal two-task communication protocol error.

3. Silently truncate the CLOB to 65536 bytes, and generate the following warning to the log file:

[Info] ORA-24345: A Truncation or null fetch error occurred

Note that the default behavior is still to silently truncate the CLOB to 65536 bytes, which may not be desirable.

Workaround:

Read the column as GsOracleCLOBs (set lobLimit > 0) and perform #readAll on the contents.

Bug 47649 - Reading large CLOBS using #readAll can fail with "insufficient room in target buffer"

Product: GemConnect

Versions: 2.3

Platform All Platforms

Fixed In: 2.4

Impact: Critical

When reading a large GsOracleCLOB object with a size greater than 65532 bytes using #readAll method, the read will fail with the error:

a GsRdbError occurred (error 25), reason:typeConversionError, Cannot convert GS Object or Oracle column data, connection: aGsOracleConnection stream: nil details: #(nil #'GsLobOp/ReadAll' 3 nil 'Insufficient room in target buffer for conversion on CHR8/STR conversion')

Workaround:

Use GsOracleCLOB>>readAll: String

Bug 47644 - Reading BLOBS/CLOBS requires SQL "SELECT ... FOR UPDATE"

Product: GemConnect

Versions: 2.3, 2.2.3, 2.2.2, 2.2, 2.1.1, 2.1, 2.0

Platform All Platforms

Fixed In: 2.4

When a GsOracleConnection is configured to return GsOracleBLOB/CLOB objects (lobLimit > 0), the SQL SELECT statement used in the #openCursorOn: call to retrieve them has to include the suffix "FOR UPDATE", even if the application won't be writing the BLOB/CLOB. For example:

stream := myConnection openCursorOn: 'select clob_column from mytable for update'.

Failure to do so will result in the Oracle error:

ORA-22292: Cannot open a LOB in read-write mode without a transaction.

Workaround:

Include "FOR UPDATE" in the SQL SELECT statement used for retrieving the BLOB/CLOB.

Bug 47213 - Customer builds of liboraapi library on GS/64 3.4 and later on AIX are broken

Product: GemConnect

Versions: 2.3

Platform AIX only

Fixed In: 2.4

Customers who build their own extended versions of the liboraapi library using the supplied Makefile643 will have problems where the new customer-supplied user actions cannot be found. When the user action is executed, the following error is generated:

Error 2358 , a ImproperOperation occurred (error 2358), Attempt to call a user action that is not registered with this virtual machine, user action name: #'CustomerUserAction' (ImproperOperation)

Workaround:

In the file Makefile643, replace line:

LDFLAGS = -G -Wl,-bbigtoc -q64 -e GciUserActionLibraryMain

with:

LDFLAGS = -G -q64 -e GciUserActionLibraryMain

Bug 46977 - Migration of GemConnect across multiple GS versions can trigger ImproperOperation

Product: GemConnect

Versions: 2.3

Platform All Platforms

Fixed In: 2.4

Re-installing GemConnect on a stone that has been upgraded from an earlier GS release may trigger the following error at the end of the installation script or when attempting to load the liboraapi library:

```
 -----------------------------------------------------
 GemStone: Error Nonfatal
 a ImproperOperation occurred (error 2142), Cannot execute method, 'method needs recompile,
 GsRdbConnection class >> _installConnManager , oop XXXXXXXXX'
```

Workaround:

Fix by removing class GsOracleConnection from the Globals dictionary and then re-installing GemConnect.

```
Login as SystemUser
```

```
run
Globals removeKey: #GsOracleConnection
%
commit
```

Bug 46826 - Do not copy the GemConnect install version of liboraapi23-643.so to $GEMSTONE/ualib

Product: GemConnect

Versions: 2.4, 2.3

Platform All Platforms

Step 10 on page 13 of the GemConnect 2.3 install guide instructs the user to copy the liboraapi23-643.so library from the GemConnect installation to $GEMSTONE/ualib. This instruction was correct when GemConnect 2.3 was first released and none of the currently released versions of GemStone had a $GEMSTONE/ualib/liboraapi23-643.so library available. But since then, each subsequent GemStone release has included an updated version of $GEMSTONE/ualib/liboraapi23-643.so that supercedes the original version included with GemConnect.

If you *do* copy over the $GEMCONNECT/ualib/liboraapi23-643.so library to a recent $GEMSTONE/ualib directory, attempts to load that library may fail. The error will vary according to the version of GemStone and the platform. As an example, the following error was generated on Linux running GS/64 3.3.3:

ERROR 2171 , a ImproperOperation occurred (error 2171), user action library load failed, 'GciLoadUserActionLibrary could not find GciUserActionLibraryMain'' in /apps/gemstone/product/ualib/liboraapi23-643.so
Reason: dlsym(GciUserActionLibMain) failed because /apps/gemstone/product/ualib/liboraapi23-643.so: undefined symbol: GciUserActionLibMain' (ImproperOperation)

Workaround:

Restore the version of $GEMSTONE/ualib/liboraapi23-643.so that was delivered with the $GEMSTONE product.

Bug 46749 - Using Linux CAP_SYS_RESOURCE causes loading GemConnect liboraapi library to hang

Product: GemConnect

Versions: 2.4, 2.3

Platform Linux only

Impact: Informational

When a topaz/gem process on Linux has the CAP_SYS_RESOURCE privilege set as per the GS/64 Linux Installation Guide, attempts to load the GemConnect liboraapi library will hang. Attempts to view the C stack via either gdb or kill -USR1 for diagnosis will fail with the error:

ptrace: Operation not permitted.

The use of the CAP_SYS_RESOURCE privilege causes the Linux dynamic library loader to ignore the $LD_LIBRARY_PATH used by Oracle to locate the Oracle libraries, resulting in the hang.

Workaround:

Copy the Oracle libraries to a directory in the system search path, such as /usr/lib.

Bug 45862 - GemConnect requires Oracle libclntsh.so.10.1 library

Product: GemConnect

Versions: 2.4, 2.3, 2.2.3, 2.2.2, 2.2.1, 2.2, 2.1

Platform All Platforms

GemConnect currently requires that the Oracle libclntsh.so.10.1 library be available in the $LD_LIBRARY_PATH. Otherwise a library load failure will occur when the GemConnect liboraapi library is loaded. This will cause problems using newer versions of Oracle (11 and 12). GemConnect *will* work on Oracle 11 and 12, but needs a symbolic link from the available libclntsh.so.X.1 library to libclntsh.so.10.1.

Workaround:

Establish a symbolic link in the Oracle library directory to the available libclntsh.so library:

For example, for Oracle 12:

```
unix>  ln -s libclntsh.so.12.1 libclntsh.so.10.1
```

Bug 45253 - GemConnect library load problems on Darwin

Product: GemConnect

Versions: 2.4, 2.3

Platform Darwin

Due to a bug in the 10.2.0.X versions of the Oracle libraries on Darwin, attempting to load the GemConnect liboraapi23-643.dylib library on GemStone/64 versions 3.2.13 and later will fail with a SIGSEGV.

Workaround:

Use 11.2 or later versions of the Oracle libraries when running GemConnect on Darwin.

Bug 44784 - Reading/Writing BLOBS/CLOBS > 64K using default LONG/LONGRAW configuration will silently truncate

Product: GemConnect

Versions: 2.4, 2.3

Platform All Platforms

Impact: Informational

By default BLOBS/CLOBS in GemConnect are treated as Oracle LONG/LONGRAW datatypes, and are limited to the default textLimit size of 65532 bytes. In earlier releases, attempting to read/write BLOBS/CLOBS larger than this would trigger a #typeConversionError (except for bug 44775). But in 2.3, GemConnect will read/write the data to/from Oracle but silently truncate the BLOB/CLOB to 65532 bytes.

Workaround:

Use the alternative BLOB/CLOB API by setting the connection lobLimit > 0 to generate GsOracleBLOB/GsOracleCLOB instances and then calling #readAll/#writeAll: to read/write the bytearray/string to/from the Oracle BLOB/CLOB.

Bug 44775 - Writing UTF8 CLOBS > 65K using default LONG/LONGRAW configuration can SIGSEGV

Product: GemConnect

Versions: 2.2.3, 2.2.2, 2.2.1, 2.2, 2.1

Platform All Platforms

Fixed In: 2.3

By default BLOBS/CLOBS in GemConnect are treated as Oracle LONG/LONGRAW datatypes, and are limited to the default textLimit size of 65532 bytes. Attempting to write BLOBS/CLOBS larger than this should trigger a GemConnect #typeConversionError.

But when doing UTF8 conversion on a CLOB (connection charConversion flag set to #UTF8) writing instances of class String that include one or more characters with a value > 127, this will sometimes fail with a SIGSEGV.

Workaround:

1. Check the size of the CLOB before attempting the write.
2. Use the alternative BLOB/CLOB API by setting the connection lobLimit > 0 to generate GsOracleCLOB instances and then calling #writeAll: to write the string to the Oracle CLOB.

Bug 44456 - Operations on Oracle Objects trigger SIGSEGV

Product: GemConnect

Versions: 2.4, 2.3, 2.2.3, 2.2.2, 2.2, 2.1.1, 2.1, 2.0.1, 2.0, 1.1.6, 1.1.5, 1.1.4

Platform All Platforms

Any attempt to access a field in Oracle that contains an Oracle object or object reference may trigger a SIGSEGV and terminate the session.

Workaround:

Avoid operations that reference objects. For example, if table MYTABLE contains columns C1, C2, and C3, and C2 refers to objects, rather than execute "SELECT * FROM MYTABLE" instead explicitly mention the columns C1 and C3 and skip C2: "SELECT C1, C3 FROM MYTABLE".

Bug 43857 - BLOB/CLOB/NCLOB operations in autoCommit mode can trigger Oracle errors

Product: GemConnect

Versions: 2.4, 2.3, 2.2.3, 2.2.2, 2.2.1, 2.2, 2.1.1, 2.1, 2.0.1, 2.0

Platform All Platforms

Oracle has an operational mode called "auto commit" which causes Oracle
to automatically commit work after each operation. Auto commit mode is
established by setting the associated GsOracleParameter instvar autoCommit
to true.

This causes problems with GemConnect operations using BLOBS/CLOBS/NCLOBS,
which do a series of Oracle OCI calls that *must not* have an intervening
commit between them in order to work properly.

Typical error is a GsRdbError readError by Oracle error ORA-01002: fetch
out of sequence.

Appearance from topaz:

ERROR 8 , a GsRdbError occurred (error 8), reason:readError, An error was
encountered while reading relational data, stream: nil (GsRdbError)

Workaround:

Set GsOracleParameter autoCommit to false.

Bug 43822 - Problems with QuadByte-range characters in CLOBS/NCLOBS

Product: GemConnect

Versions: 2.4, 2.3

Platform All Platforms

Currently you cannot read/write QuadByteStrings containing characters with
a codepoint > 16rFFF to/from CLOBS/NCLOBS. Attempting to do so will generate
the error:

```
GemStone: Error Nonfatal
a GsRdbError occurred (error 25), reason:typeConversionError, Cannot convert
GS Object or Oracle column data, connection: aGsOracleConnection stream:
nil details: anArray( nil, #'GsLobOp/ReadAll', 4, nil, 'Source sequence
is illegal/malformed on UTF8/STR conversion') Error Category: 231169 [GemStone]
Number: 25 Arg Count: 0 Context : XXXX exception : XXXXX
```

Workaround:

No workaround.

Bug 43788 - NCLOBS cannot be read as LONG/LONGRAW

Product: GemConnect

Versions: 2.4, 2.3

Platform All Platforms

GemConnect currently allows BLOBS and CLOBS to be read in as LONG/LONGRAW
Oracle datatypes, allowing the use of normal SQL select/insert calls to
access and update them. But the new NCLOB datatype cannot be processed
this way. Attempting to do so will generate the error:

```
ERROR 6 , a GsRdbError occurred (error 6), reason:oracleError, An unexpected
error was encountered during Oracle processing, connection: aGsOracleConnection
stream: aGsRdbWriteStream details: 'Attempt to read/write an NCLOB as a
LONG/LONGRAW' (GsRdbError)
```

Workaround:

Access NCLOBS using the GsOracleLOB readAll / writeAll operators.

Bug 43363 - GemConnect 2.2.2 Library on GS/32 6.6.4 on Solaris fails

Product: GemConnect

Versions: 2.2.2

Platform Solaris

Impact: Critical

The GemConnect 2.2.2 library liboraapi222-32.so as delivered fails to load on the sparc.Solaris version of the new GS/32 6.6.4 release. The full error is:

```
GemStone: Error        Nonfatal
An attempt to load a user action library failed because: 'GciLoadUserActionLibrary could not load
$GEMSTONE/ualib/liboraapi222-32.so.
  Reason: dlopen$GEMSTONEualib/liboraapi222-32.so) failed because ld.so.1: topaz: fatal:
$GEMSTONE/ualib/liboraapi222-32.so: unknown file type Error Category: [GemStone] Number: 2171 Arg Count: 1
Arg 1: GciLoadUserActionLibrary could not load $GEMSTONE/ualib/liboraapi222-32.so.
  Reason: dlopen($GEMSTONE/ualib/liboraapi222-32.so) failed because ld.so.1: topaz: fatal:
$GEMSTONE/ualib/liboraapi222-32.so: unknown file type
```

Contact GemTalk Systems Technical Support for a new library built for this GS version and platform.

Bug 42711 - GemConnect library does not load on GS/S 6.6.3.2 on AIX

Product: GemConnect

Versions: 2.2.2

Platform AIX

Impact: Critical

Attempting to use GemConnect 2.2.2 on GS/S 6.6.3.2 on AIX will fail with an error similar to the following when the system attempts to load the liboraapi222-32.so library:

```
GemStone: Error        Nonfatal
An attempt to load a user action library failed because: 'GciLoadUserActionLibrary
could not load $GEMSTONE/ualib/liboraapi222-32.so.
  Reason: dlopen($GEMSTONE/ualib/liboraapi222-32.so) failed because rtld:
0712-001 Symbol __gxx_personality_v0 was referenced
      from module $GEMSTONE/ualib/liboraapi222-32.so(), but a runtime definition
      of the symbol was not found.'. Error Category: [GemStone] Number:
2171 Arg Count: 1
Arg 1: GciLoadUserActionLibrary could not load ./liboraapi222-32.so.
  Reason: dlopen(./liboraapi222-32.so)
```

Workaround:

Contact GemStone Technical Support who will provide you with a recompiled version of the GemConnect library that will work properly under AIX.

Bug 42701 - Missing AIO support on AIX causes GemConnect load errors

Product: GemConnect

Versions: 2.4, 2.3, 2.2.3, 2.2.2, 2.2.1, 2.2, 2.1.1, 2.1, 2.0.1, 2.0

Platform AIX

Impact: Critical

When AIO is not configured or available on AIX, loading of the GemConnect library will fail with the following error:

```
GemStone: Error Nonfatal
user action library load failed, 'GciLoadUserActionLibrary failed, dlopen
failed, 0509-130 Symbol resolution failed for /usr/lib/threads/libc.a[aio_64.o]
```

because: 0509-136 Symbol kaio_rdwr64 (number 1) is not exported from dependent module /unix. 0509-136 Symbol listio64 (number 2) is not exported from dependent module /unix. 0509-136 Symbol acancel64 (number 3) is not exported from dependent module /unix. 0509-136 Symbol iosuspend64 (number 4) is not exported from dependent module /unix. 0509-136 Symbol... Error Category: [GemStone] Number: 2171 Arg Count: 1
Arg 1: GciLoadUserActionLibrary failed, dlopen failed, 0509-130 Symbol resolution failed for /usr/lib/thre

Workaround:

Configure/install and enable AIO support on the platform.

Bug 42458 - Beta support for Kerberos

Product: GemConnect

Versions: 2.4, 2.3, 2.2.3

Platform All Platforms

Customers interested in evaluating a beta version of Kerberos support for GemConnect may contact GemTalk Technical Support for a special release.

Bug 42060 - Using BLOB/CLOB on Windows causes problems on commit or connection close

Product: GemConnect

Versions: 2.2.2

Platform Windows

Performing any operations on a BLOB/CLOB in Windows will cause the following error when attempting to commit or close the associated connection:

ORA-22297: warning: Open LOBs exist at transaction commit time

Workaround:

There is currently no work-around.

Bug 42055 - Windows GemConnect library won't load on GS/S 6.6 and later

Product: GemConnect

Versions: 2.2.2

Platform Windows only

Impact: Critical

Depending on your site configuration, the GemConnect library oraapi222-32.dll may not load on Windows using GS/S 6.6 and later. Error generated is:

```
-----------------------------------------------------
GemStone: Error        Nonfatal

An attempt to load a user action library failed because:
'GciLoadUserActionLibrary could not load oraapi222-32.
  Reason: LoadLibrary(oraapi222-32) failed with errcode 126'.
```

```
Error Category: [GemStone] Number: 2171 Arg Count: 1
Arg 1: GciLoadUserActionLibrary could not load oraapi222-32.
  Reason: LoadLibrary(oraapi222-32) failed with errcode 126
---------------------------------------------------
```

If you experience this error, contact GemStone Technical Support, providing the version numbers for GS/S, GemConnect, and Oracle used in your configuration. We can provide you with a reconstructed version of oraapi222-32.dll that will work properly.

Workaround:

Contact GemStone Technical Support for a new oraapi22-32.dll library.

Bug 41286 - ProfMonitor sampling bias on GemConnect

Product: GemConnect

Versions: 2.4, 2.3, 2.2.3, 2.2.2, 2.2.1, 2.2, 2.1.1, 2.1, 2.0.1, 2.0

Platform All platforms

Impact: Informational

Because GemConnect uses a user action library, it is subject to server bug 41283.

When the ProfMonitor sample timer fires while user action C code is executing, the sample is not actually taken until control returns back to smalltalk. This is usually when the user action code finishes and control has returned back to the calling smalltalk method. But if the user action code makes a smalltalk callback, the sample will be taken in that smalltalk method. In both cases this can result in a sampling bias toward these methods, leading to higher than expected tally counts and percentages for these methods.

For GemConnect, this will be most noticible for methods:

GsRdbReadStream(class)>>newForConnection:
(called during GsOracleConnect>>openCursorOn:*)

DateTime>>_adjustForLocalOffset
(called during GsRdbReadStream operations returning date/time information)

System(class)>>_processDeferredGciUpdates
(called during GsRdbReadStream scanning operations)

Collection>>remove:*
(called during GsRdbReadStream>>free)

In addition, any methods that are used as accessor or setter methods in a rdbColumnMapping may also exhibit some bias.

Workaround:

No workaround. Be aware of this behavior when using ProfMonitor on code that uses GemConnect.

Bug 40811 - Errors on UTF8 CLOB operations using DoubleByteStrings on Windows

Product: GemConnect

Versions: 2.2.2, 2.2.1, 2.2, 2.1.1, 2.1, 2.0.1, 2.0

Platform Windows

Fixed In: 2.2.3

Impact: Critical

Attempts to read/write DoubleByteStrings into UTF8-based CLOBS on Windows does not correctly byte-swizzle the characters. This can result in #typeConversionErrors. If *only* GemConnect is used to access the Oracle data, the results in GemConnect will be correct, although the data will be stored incorrectly in Oracle. But incorrect results will occur if other external processes read/write the Oracle data.

Bug 40755 - Errors on UTF8 conversion on Linux, Solaris/Intel

Product: GemConnect

Versions: 2.2.1, 2.2, 2.1.1, 2.1, 2.0.1, 2.0

Platform Linux, Solaris/Intel ONLY

Fixed In: 2.2.2

Impact: Critical

Attempts to read/write UTF8 Oracle data on Linux and Solaris/Intel platforms do not correctly byte-swizzle the characters. This can result in incorrect results or #typeConversionErrors.

In the case of CLOBs, if *only* GemConnect is used to access the Oracle data, the results in GemConnect will be correct, although the data will be stored incorrectly in Oracle. But incorrect results will occur if other external processes read/write the Oracle data.

Workaround:

No workaround.

Bug 40717 - Problems reading/writing very large UTF8 CLOBS

Product: GemConnect

Versions: 2.2.1, 2.2, 2.1.1, 2.1

Platform All platforms

Fixed In: 2.2.2

Attempts to read/write very large CLOBS using UTF8 conversion that have a size greater than the setting of the associated GsOracleParameter #textLimit field may result in #typeConversionError or in incorrect data.

Workaround:

Increase setting of GsOracleParameter #textLimit field to a value greater than the maximum size of the post-UTF8 converted CLOB. Note that this will only work up to 64K bytes (the limit for textLimit).

Bug 40711 - Errors during BLOB/CLOB reads can trigger ORA_03127 errors

Product: GemConnect

Versions: 2.2.1, 2.2, 2.1.1, 2.1

Platform All platforms

Fixed In: 2.2.2

If an error occurs during the reading of a very large BLOB/CLOB object
from Oracle, the next following Oracle operation may fail with the error:

ORA-03127: no new operations allowed until the active operation ends

Workaround:

Disconnect and then re-connect to Oracle.

Bug 40633 - SIGSEGV on nil GsOracleParameter fields

Product: GemConnect

Versions: 2.2.1, 2.2, 2.1.1, 2.1, 2.0.1, 2.0, 1.1.6, 1.1.5, 1.1.4, 1.1.3, 1.1.2, 1.1.1, 1.1, 1.0

Platform All platforms

Fixed In: 2.2.2

Leaving any of the following GsOracleParameter instance variables set to
nil will result in a SIGSEGV when attempting to login to Oracle:

1. server
2. username
3. password

Workaround:

Set the necessary instance variables.

Bug 40290 - Oracle Library initialization problems not reported clearly

Product: GemConnect

Versions: 2.2.1, 2.2, 2.1.1, 2.1, 2.0.1, 2.0

Platform All platforms

Fixed In: 2.2.2

Problems with Oracle Library initialization are not reported properly.
The error message returned may not indicate that the problem is with the
loading of Oracle Library components. Possible causes include:

1. $LD_LIBRARY_PATH does not include the $ORACLE_HOME/lib directory.
2. $ORACLE_HOME/lib directory and/or its contents have incorrect permissions.
3. Wrong/incompatible version of $ORACLE_HOME/lib.

Workaround:

Check items listed above in cases where the first Oracle operation fails
with an obscure error message.

Bug 40132 - UTF8 conversion needs $NLS_LANG set to .UTF8

Product: GemConnect

Versions: 2.4, 2.3, 2.2.3, 2.2.2, 2.2.1, 2.2, 2.1.1, 2.1, 2.0.1, 2.0

Platform All platforms

As of Oracle 10, when using GsoracleConnection charConversion/charGS/ncharGS
mode #UTF8, it is now necessary to set the environmental variable $NLS_LANG
to an appropriate character set with suffix *.UTF8

Failure to do this will cause Oracle to *also* perform UTF8 conversion
on a string that has already been converted to UTF8 by GemConnect. If
data is *only* read/written by GemConnect, then you may not even notice
a problem, as the double-conversion preformed during a read will cancel
out the double-conversion performed during the write. But incorrect results
will occur if data is read/written to Oracle outside of GemConnect.

Another side-effect is that the extra conversion could result in increasing
the size of the string stored in Oracle, potentially causing it to overflow
the allocated space for a string that would have otherwise fit. This case
would trigger Oracle error:

ORORA-12899: value too large for column "NAME" (actual: XX, maximum: XX)

For US customers, the correct value for $NLS_LANG would be AMERICAN_AMERICA.UTF8.
Similar values would apply for international customers, according to their
locale.

This requirement is also documented in the GemConnect Programming Guide,
Section 3.1.

Workaround:

Set $NLS_LANG to the appropriate value.

Bug 39964 - Reading large CLOBS with UTF conversion overflows buffer

Product: GemConnect

Versions: 2.2, 2.1.1, 2.1, 2.0

Platform All platforms

Fixed In: 2.2.2

When reading large CLOBS from Oracle and performing UTF8 conversion,
the internal buffer used to hold the data can overflow, causing memory
corruption, usually indicated by SIGSEGV or SIGBUS errors.

The size at which the overflow occurs is dependent on how Oracle is
configured; an approximate value for a typical configuration is around
4000 characters.

Workaround:

No workaround

Bug 39944 - Problem with reads/writes to Oracle TIMESTAMP fields

Product: GemConnect

Versions: 2.2, 2.1.1, 2.1

Platform All platforms

Fixed In: 2.2.2

Impact: Critical

There is a serious problem with GemConnect reads/writes to Oracle TIMESTAMP fields containing a fractional second component (Oracle data fields defined as TIMESTAMP(X), where X > 0). The current implementation incorrectly scales the fractional second component when translating it to milliseconds for the GemStone DateTime object.

When reading the Oracle TIMESTAMP(X) field into GemStone, this can result in the millisecond value being too large, overflowing into the seconds/minutes/hours fields and resulting in the generation of a GemStone DateTime object that is incorrect. In some cases, the excessively large millisecond value will break the GCI call generating the DateTime, resulting in a GS #objErrDoesNotExist error. The possible variance from the correct time is proportional to the scale of the fractional second specification for the TIMESTAMP field:

TIMESTAMP(X)	Worst-case variation
X = ?	from correct time
1	28 hours
2	2.8 hours
3	17 minutes
4	1.7 minutes
5	10 seconds
6	1 second (Note: TIMESTAMP(6) is the Oracle default)
7	0.1 seconds
8	0.01 seconds
9	correct

When writing from GemStone to Oracle, the conversion of the GemStone DateTime millisecond field to the TIMESTAMP fractional seconds field will result in a fractional second value that is too small, but the other fields in the TIMESTAMP will be correct. The worst-case variation will be < 1 second.

Note that if the Oracle TIMESTAMP field is *only* generated by GemStone, and is *only* read back by GemStone, that the DateTime write/read results will be consistent, as the incorrect conversions will cancel each other out. The TIMESTAMP as stored in Oracle will be incorrect, but GemStone will see a consistent DateTime value between write and read. Only the TIMESTAMP fractional second field is incorrect in this case.

Customers affected by this problem should upgrade to the upcoming GemConnect 2.2.1 release, which will specifically address this problem. The release notes will include instructions on how to fix Oracle TIMESTAMP fields that were written by earlier versions of GemConnect.

Workaround:

No workaround

Bug 39531 - CR's (carriage returns) stripped from SQL statements

Product: GemConnect

Versions: 2.4, 2.3, 2.2.3, 2.2.2, 2.2.1, 2.2, 2.1.1, 2.1

Platform All platforms

Impact: Critical

In GemConnect 2.1, we "fixed" a problem with Oracle parsing of stored procedures by scanning SQL strings passed via GemConnect #openCusorOn:* or #execute:* methods, and replacing any carriage returns with spaces before passing them on to Oracle.

Unfortunately, this is the wrong approach if you happen to want those carriage returns as part of the data stored in Oracle. If any strings that you write to Oracle contain carriage returns, you must apply the following workaround to avoid corrupting your data.

Workaround:

The bug was introduced with the addition of a new method called GsOracleConnection>>prepareSqlString:. This method replaces any carriage returns with spaces, and also converts a DoubleByteString to a String if possible.

Replace the method GsOracleConnection>>prepareSqlString: with the following code, which retains the DoubleByteString conversion, but removes the code that replaces the carriage returns with spaces:

```
--------------------------------------------------------

category: 'Temporary 39531 Fix'
method: GsOracleConnection
prepareSqlString: sqlString

" Do some pre-processing of a SQL command string
  to make sure it complies with various Oracle restrictions:

  * No DoubleByteStrings ( try converting to String )
  * [code to replace carriage returns removed]
"

   | str |
   sqlString _validateClass: String.
   str := sqlString asString copy.
   " At this point, we shouldn't still have a DoubleByteString "
   (str isKindOf: DoubleByteString)
     ifTrue: [ str _errorExpectedClass: String ].
   ^ str
%
```

Bug 39389 - QuadByteStrings not supported in GemConnect

Product: GemConnect

Versions: 2.2, 2.1.1, 2.1, 2.0

Platform All platforms

Fixed In: 2.3

QuadByteStrings are not currently supported in GemConnect.

Workaround:

No workaround

Bug 39360 - Table/Column names limited to 120 characters

Product: GemConnect

Versions: 2.1, 2.0.1, 2.0

Platform All platforms

Fixed In: 2.2

Impact: Critical

The length of table and column names used for Oracle tables is limited to 120 characters. Exceeding this limit can cause SIGSEGV errors.

While normally more than adequate, these fields can also be used for the name of a stored procedure, along with it's argument list. For a long argument list with lengthly argument names, this length can exceed 120 characters.

Workaround:

No workaround

Bug 38174 - Problem with object change notification on deletes

Product: GemConnect

Versions: 2.0.1, 2.0, 1.1.5

Platform All platforms

In GS/S 32-bit version 6.3 and later, and on GS/S 64-bit version 2.2.0 and later, changes were made in how object change notification handles deletes from collections. Due to these changes, using GemConnect versions prior to 2.1 on these GS/S versions will cause indexing errors in the #aboutToDelete:index: method.

If the following apply, you should use the updated version of #aboutToDelete:index: listed below:

1. Your GemConnect application uses object change notification.
2. Your application defines the method #aboutToDelete:index:
3. You are using GemConnect versions 1.X (any version), 2.0, or 2.0.1
4. You are using GS/S 32-bit versions 6.3 or later, or
   You are using GS/S 64-bit versions 2.2.0 or later

Workaround:

Add and commit the following code as SystemUser:

```
category:  'Bug 38174 fix'
method: GsRdbChangeNotifObj
deletingIn: aSeqColl startingAt: offset count: count

   | index |

  " for GS/32 6.2.X and earlier, use the following "
  " for GS/64 2.1.X and earlier, use the following "
  " index := offset - (aSeqColl class instSize). "

  " for GS/32 6.3.0 and later, use the following "
  " for GS/64 2.2.0 and later, use the following "
  index := offset.

  aSeqColl aboutToDelete: (aSeqColl copyFrom: index to: (index + count - 1))
        index: index.
%
```

Bug 37906 - Strings silently truncated for Oracle data

Product: GemConnect

Versions: 2.0.1, 2.0

Platform All platforms

Fixed In: 2.1

When writing Strings to an Oracle database, GemConnect will silently truncate the strings to fit the Oracle column if necessary, without raising an error.

Workaround:

No workaround

Bug 37857 - GemConnect 2.0/2.0.1 problems on GS/64 2.2.X AIX/Linux

Product: GemConnect

Versions: 2.0.1, 2.0

Platform AIX and Linux only

Impact: Critical

Due to changes in compiler and linker options, currently released versions of GemConnect will not run with GS/64 versions 2.2.0 and later on AIX and Linux. GemConnect will work fine on Solaris.

Attempting to use GemConnect in these configurations will result in either problems loading the GemConnect userAction library liboraapi20-642.so, or errors related to the inability to find particular GemConnect user actions.

Workaround:

Contact GemStone Technical Support, who can supply you with a special build of GemConnect 2.0.1 that fixes this problem.

Bug 37786 - Oracle error information is not cleared until next error

Product: GemConnect

Versions: 2.0.1, 2.0

Platform All platforms

Fixed In: 2.1

Oracle error information is returned as an argument in appropriate GemConnect errors, and is also available in GsOracleConnection>>messages. This information is kept around until the next Oracle error occurs.

If a non-Oracle related GemConnect error is triggered later, this now obsolete information can still be retrieved from GsOracleConnection>>messages, and can cause some confusion since it's not related with the current error scenario.

Also, some Oracle errors are handled internally by GemConnect and should be invisible to the application. But these will also set GsOracleConnection>>messages, and can cause confusion on a subsequent non-Orcle related GemConnect error.

Workaround:

Ignore GsOracleConnection>>messages when handling a non-Oracle GemConnect error.

Bug 37619 - GemConnect 2.0/2.0.1 doesn't load on AIX GS/64 2.2.2

Product: GemConnect

Versions: 2.0.1, 2.0

Platform AIX

Impact: Critical

The useraction library liboraapi20-642.so for GemConnect versions 2.0 and 2.0.1 will not load on AIX GemStone/S 64 Bit versions 2.2.2 and later. This problem is AIX only - Solaris works fine.

Note that customers cannot simply recompile/relink the GemConnect library, due to other changes in GS/64 2.2.2.

Workaround:

Contact Technical Support for a patch version of GemConnect 2.0.1 that will work with AIX GS/64 2.2.2.

Bug 37609 - Oracle errors when not logged in trigger #gciErrBadSessionId

Product: GemConnect

Versions: 2.0.1, 2.0

Platform All platforms

Fixed In: 2.1

Any GemConnect Oracle errors generated while the session is not yet logged into GemStone (for example, problems loading liboraapi library using a topaz loadua commnad) will generate a #gciErrBadSessionId error, rather than a more helpful Oracle error.

Workaround:

Login to GemStone first, then attempt to load the GemConnect liboraapi user action library. This should expose the underlying Oracle error (typically, a problem finding the $ORACLE/lib directory).

Bug 37550 - New writeStreams don't support synonyms

Product: GemConnect

Versions: 2.0.1, 2.0

Platform All

Fixed In: 2.1

The new GemConnect writeStream feature does not support Oracle synonyms. Only actual table names (prefixed with scemas if necessary) are currently supported.

If attempts to use #open[Insert|Update|Delete]cursorOn:* fail on a given #rdbTableName that works when using #openCursorOn:*, check with your Oracle DBA to determine if the #rdbTableName is actually a synonym.

See bug 37363 for a related writeStream limitation on views.

Workaround:

Use the underlying table(s) referenced by the synonym.

Bug 37543 - GsRdbWriteStreams need schema or get #invalidTableErrors

Product: GemConnect

Versions: 2.4, 2.3, 2.2.3, 2.2.2, 2.2.1, 2.2, 2.1.1, 2.1, 2.0.1, 2.0

Platform All

When generating instances of the new GsRdbWriteStream class via these methods:

```
openInsertCursorOn:*
openUpdateCursorOn:*
openDeleteCursorOn:*
```

and the Oracle schema that contains the table is different from the current schema of the current Oracle user, you will need to prefix the table name with the schema, using standard Oracle schema naming conventions.

For example, if table MYTABLE was created by Oracle user BOB using his default current schema (same as his user name: BOB), then for Oracle user FRED to access it, he will need to use 'BOB.MYTABLE' as the name of the table.

The fully qualified table name can be either passed directly to the open*CursorOn:* method via the tableName: argument, or made part of the table's associated #rdbTableName method definition.

Failure to properly qualify the table name with the schema when needed will result in a GemConnect #invalidTableName error.

Workaround:

No workaround

Bug 37455 - #rdbColumnMapping must follow strict order

Product: GemConnect

Versions: 2.0.1, 2.0, 1.1.5, 1.1.4, 1.1.3, 1.1.2, 1.1.1, 1.0

Platform All platforms

When using #rdbColumnMapping, the order of the column declarations must follow that used in the internal Oracle table layout. Failure to do so will result in a spurious #columnBindingError.

The correct order can be determined from SQL using the "DESCRIBE tablename" command.

Workaround:

Use the SQL command "DESCRIBE tablename" to determine the correct order.

Bug 37363 - New writeStreams don't support views

Product: GemConnect

Versions: 2.0.1, 2.0

Platform All

Fixed In: 2.1

The new GemConnect writeStream feature does not support Oracle views. Only actual table names (prefixed with schemas if necessary) are currently supported.

If attempts to use #open[Insert|Update|Delete]cursorOn:* fail on a given #rdbTableName that works when using #openCursorOn:*, check with your Oracle DBA to determine if the #rdbTableName is actually a view.

See bug 37550 for a related writeStream limitation on synonyms.

Workaround:

Use the underlying table(s) referenced by the view.

Bug 36634 - Specialized Integer queries return Floats

Product: GemConnect

Versions: 2.4, 2.3, 2.2.3, 2.2.2, 2.2, 2.1.1, 2.1, 2.0.1, 2.0

Platform All

Fixed In: n/a

Impact: Informational

Under certain conditions, specialized Oracle SQL queries that are
expected to return Integers will instead return Floats. Since the
data returned from Oracle is ambiguous, GemConnect cannot convert
these values.

For example, the query:

'select count (*) from <tablename>'

(returns number of rows in <tablename>)

will return a Float.

Workaround:

Modify application code to expect a Float and handle it accordingly.

Bug 36402 - GemConnect does not handle 2007 DST changes

Product: GemConnect

Versions: 2.0, 1.1.6, 1.1.5, 1.1.4, 1.1.3, 1.1.1

Platform All

Fixed In: 2.0.1

Impact: Critical

The rules govering the start and end of Daylight Savings Time have
been changed for the US, starting in 2007. DST will start earlier
and end later.

GemConnect shared libraries include internal C code that implements
the old rules. This means that instances of DateTime read into
GemStone from Oracle via GemConnect, for US TimeZones, representing
times in the period from March 11, 2007 to April 1, 2007, will be
incorrect by one hour. A similar problem exists for the period of
change in the end of DST, and the equivalent times in subsequent
years.

GemConnect 2.0 uses GemStone server code for conversion when writing
GemStone DateTimes to Oracle, and earlier versions of GemConnect
required the customer to create a SQL string for writing a DateTime
to Oracle. Therefore, writing DateTimes to Oracle is not affected
by this GemConnect bug. However, due to the related GemStone server
bugs 35810 and 35811 (see
http://support.gemstone.com/gemstone_s/downloads/patches/timezone/index.html),
future DateTimes in US TimeZones during the affected periods,
depending on your application, may be incorrect in Oracle.

Note that Oracle also requires patches to handle the DST change.
Details for installing these patches are available at the Oracle
MetaLink site under 359145.1 and 402742.1.

Bug 35736 - No DoubleByteStrings for SQL statements

Product: GemConnect

Versions: 2.0.1, 2.0

Platform All

Fixed In: 2.1

You cannot use a DoubleByteString as an SQL commandString in GemConnect execute* methods. Doing so will generate GemConnect #invalidSQL errors.

Workaround:

No workaround

Bug 35677 - DoubleByteString characters must be < 0x8000

Product: GemConnect

Versions: 2.2, 2.1.1, 2.1, 2.0.1, 2.0

Platform All

Fixed In: 2.3

Impact: Critical

When using UTF8/UTF16 conversion in GemConnect, all characters used in DoubleByteStrings must have an integer encoding < 0x8000. Failure to do so will result in spurious GemConnect #typeConversionErrors.

Workaround:

No workaround

Bug 35459 - Oracle OCI has problem with CR's in SQL statements

Product: GemConnect

Versions: 2.0.1, 2.0, 1.1.5, 1.1.4, 1.1.3, 1.1.2, 1.1.1, 1.0

Platform All

Fixed In: 2.1

Impact: Informational

The Oracle OCI, which handles the interface from GemConnect to Oracle, has problems dealing with carriage returns (Character cr) used by themselves as line separators. This is not universal -- for example, SELECT statements will be parsed correctly, but statements using BEGIN/END blocks will fail with ORACLE error ORA-06550 and the error message:

```
--------------------------------------------------------------------
PLS-00103: Encountered the symbol "" when expecting one of the following:
  begin function package pragma procedure subtype type use
  <an identifier> <a double-quoted delimited-identifier> form
  current cursor
-------------------------------------------------------------------
```

Workaround:

Reformat the SQL statement string to remove the carriage returns before passing it on to the GemConnect #executeNoResults:/#open*Cursor* method.

Bug 32854 - Problems with customer-built 64-bit liboraapiNN.so library

Product: GemConnect

Versions: 2.4, 2.3, 2.2.3, 2.2.2, 2.2, 2.1.1, 2.1, 2.0.1, 2.0, 1.1.5

Platform GS64, Solaris

When using a customer-built liboraapiNN.so GemConnect library on GemStone64 on Solaris, you must include the directories /usr/lib/sparcv9 and /usr/ucblib/sparcv9 in your $LD_LIBRARY_PATH. Otherwise, when the gem/topaz session attempts to load OS system libraries used by liboraapiNN.so, it will search /usr/lib and /usr/ucblib, finding the 32-bit versions of the OS system libraries instead. This will trigger "wrong ELF class: ELFCLASS32" errors.

Workaround:

Ensure your $LD_LIBRARY_PATH is correct.

Bug 32853 - Customer-built liboraapi library needs to link to correct Oracle libraries

Product: GemConnect

Versions: 2.4, 2.3, 2.2.3, 2.2.2, 2.2.1, 2.2, 2.1.1, 2.1, 2.0.1, 2.0, 1.1.6, 1.1.4, 1.1.3

Platform All Platforms

Impact: Informational

When building a new liboraapi GemConnect library on a system that has both 32-bit and 64-bit Oracle installed, the customer must be careful to configure the building and execution of the liboraapi library so that the correct Oracle 32 or 64 bit libraries are used.

In a dual 32- and 64-bit Oracle installation, the 32-bit libraries are stored in $ORACLE_HOME/lib32, while the 64-bit versions are stored in $ORACLE_HOME/lib. If the liboraapi library is built and/or executed using the wrong Oracle libraries, the topaz/gem session will fail when the session attempts to login to Oracle. For example, on Solaris, GemStone error #rtErrUalibLoadFailed (2171) will be raised with the message:

```
   wrong ELF class: ELFCLASS64
```

while on AIX, the session will crash with a SIGILL exception and core dump.

To avoid this, do the following:

1. While building the liboraapi library, the supplied Makefile must be modified as explained in the comment fields. The relevant portion is:

   ```
   ----------------------------------------------------
   ORALIB = $(ORACLE_HOME)/lib

   # When 64-bit Oracle is installed,
   # use the following definition of ORALIB instead:
   # ORALIB = $(ORACLE_HOME)/lib32
   ----------------------------------------------------
   ```

   In this case, the first ORALIB definition should be commented out and the later definition uncommented, so that it reads as:

   ```
   ----------------------------------------------------
   # ORALIB = $(ORACLE_HOME)/lib

   # When 64-bit Oracle is installed,
   # use the following definition of ORALIB instead:
   ORALIB = $(ORACLE_HOME)/lib32
   ----------------------------------------------------
   ```

2. When executing the customer-generated liboraapi library in GemStone, the $LD_LIBRARY_PATH environmental variable for the user executing the topaz/gem process should include $ORACLE_HOME/lib32 and *not* $ORACLE_HOME/lib.

Bug 32787 - Object change notification does not work in GS/S 64 earlier than 2.1

Product: GemConnect

Versions: 2.1, 2.0.1, 2.0, 1.1.6, 1.1.5

Platform GS64

Impact: Informational

Versions of GS/S 64 earlier than 2.1 do not support object change notification, which is frequently used in GemConnect applications to trigger appropriate Oracle DB updates when repository objects are modified. Applications which use or override the following methods will therefore not function correctly:

```
 Object
  notifyChange
  notifyChange:
  aboutToChange:newValue:

 SequencableCollection
  aboutToInsert:index:
  aboutToDelete:index:

 UnorderedCollection
  aboutToAdd:
  aboutToRemove:
```

```
 GsRdbChangeNotifObj
   removeAllNotification
   invokingBecomeOn:to:
   removing:from:
   modifyingObject:atOffset:to:
   modifyingByteObject:startingAt:withNewValue:
   inserting:into:at:insertSize:
   deletingIn:startingAt:count:
   changingSizeOfObject:to:
   adding:to:
```

Workaround:

Upgrade the GemStone/S 64 server to the most recent version.

Bug 29870 - GsRdbChangeNotifObj>>removeAllNotification is broken

Product: GemConnect

Versions: 2.0.1, 2.0, 1.1.6, 1.1.5, 1.1.4, 1.1.3, 1.1.2, 1.1.1

Platform All

Fixed In: 2.1

The method GsRdbChangeNotifObj>>removeAllNotification calls an incorrect primitive number, and will return an error if you attempt to execute it.

Workaround:

File in the following as SystemUser and commit:

```
category: 'Clean Up'
method: GsRdbChangeNotifObj
removeAllNotification

"Removes this change notification object from all dependency lists. The change
notification object will no longer be notified when objects are modified.

ARGUMENTS:
  none

RETURN VALUE:
  unspecified

ERRORS:
  none
"

  <primitive: 901>
  | entries depList |

  entries := SharedDependencyLists removeEntriesContaining: self.
  1 to: entries do: [  :i |
    depList := entries at: i.
    depList _removeCompletelyPathTerm: self.
    SharedDependencyLists at: depList logging: false.
    ].

  System _disableProtectedMode.
%
```

Bug 26608 - GemConnect and user action libraries

Product: GemConnect

Versions: 2.4, 2.3, 2.2.3, 2.2.2, 2.2.1, 2.2, 2.1.1, 2.1, 2.0.1, 2.0, 1.1.6, 1.1.5, 1.1.4, 1.1.3, 1.1.2, 1.1.1

Platform All

During session logout, GemStone normally clears the user action caches and unloads the associated user action libraries. But when GemConnect is used, the release of the GemConnect user action library causes various problems due to signals sent from Oracle background processes being dispatched to now non-existent signal handlers. To avoid this, we now keep the user action libraries loaded across session logouts when GemConnect is used.

This is normally invisible to the user, but could have side effects on sessions which load other user action libraries along with GemConnect, and repeatedly login/logout and load/reload the libraries. Possible problems could include spurious errors about the libraries already being loaded when the reload is attempted. Also, space management will be less than optimal, as the libraries are not being cleared.

Workaround:

Modify code to ignore "user action library already loaded" errors.

Bug 14315 - Non-mapped instance variables result in errors during SQL update generation

Product: GemConnect

Versions: 2.4, 2.3, 2.2.3, 2.2.2, 2.2.1, 2.2, 2.1.1, 2.1, 2.0.1, 2.0, 1.1.6, 1.1.5, 1.1.4, 1.1.3, 1.1.2, 1.1.1, 1.0

Platform All

If you attempt to generate an SQL update command using the generateSQLUpdate:instVarName:newValue:table:columns:keys: method for an instance variable that is not part of the receiving object's column mapping, you will receive the following error:

```
_____________________________
GemStone: Error  Nonfatal
An indexable object or NSC anArray() was referenced with an index 2 that
was out of range.
Error category: [GemStone] Number: 2003 Arg Count:2
Arg 1: an Array
Arg 2: 2
```

Workaround:

Verify that instance variables are column mapped before generating any SQL update statements for them.

Bug 14209 - Cannot get Oracle stored procedure results back

Product: GemConnect

Versions: 2.2, 2.1.1, 2.1, 2.0.1, 2.0, 1.1.6, 1.1.4, 1.1.3, 1.1.2, 1.1.1, 1.0

Platform All

Fixed In: 2.3

GsOracleConnection only supports the execution of stored procedures through its executeNoResults: method. Since it does not support execution of anything other than select statements in its openCursorOn: and execute: methods (the only methods which actually return results), it is not possible to obtain stored procedure results directly.

See bugnote #14078 for related information.

Workaround:

Build a PL/SQL block to wrap the stored procedure and store the return values in a table. Then query the table for results. This requires two calls to the Oracle database.

Bug 13401 - Unresolved references in relink of GemConnect on Solaris

Product: GemConnect

Versions: 2.4, 2.3, 2.2.3, 2.2.2, 2.2.1, 2.2, 2.1.1, 2.1, 2.0.1, 2.0, 1.1.6, 1.1.5, 1.1.4, 1.1.3, 1.1.2, 1.1.1, 1.0

Platform Solaris

Impact: Informational

On Solaris only, relinking GemConnect will result in many unresolved GemBuilder for C function references. This is expected. On Solaris, the GemStone dynamic user-action library attachment system relies on Solaris to resolve these references at run time.

Workaround:

There is no workaround. While there are link switches that will inhibit the list of unresolved references, these switches also alter the runtime binding behavior of the library so that GemStone will not be able to load multiple user-action libraries.

Bug 12691 - Object audit results in loss of GemConnect connection cache

Product: GemConnect

Versions: 2.2, 2.1.1, 2.1, 2.0.1, 2.0, 1.1.6, 1.1.5, 1.1.4, 1.1.3, 1.1.2, 1.1.1, 1.0

Platform All

GemConnect keeps its connection manager object in the session state. If an object audit is performed, the session state is wiped clean and the connection manager will disappear. This manifests itself as a lost connection cache in GemConnect.

Workaround:

Do not perform an object audit in sessions that are using GemConnect.