


GemStone/S 64 BitTM **Installation Guide**

for Solaris on
x86 _64 Compatible Systems

Version 3.6.2

November 2021



INTELLECTUAL PROPERTY OWNERSHIP

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

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

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

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

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

COPYRIGHTS

This software product, its documentation, and its user interface © 1986-2021 GemTalk Systems LLC. All rights reserved by GemTalk Systems.

PATENTS

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

TRADEMARKS

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

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

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

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

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

Linux is a registered trademark of Linus Torvalds and others.

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

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

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

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

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

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

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

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



Preface

About This Documentation

This document explains how to install GemStone/S 64 Bit version 3.6.2 on a workstation running Solaris on x86 _64 Compatible Systems; and how to upgrade from previous GemStone/S 64 Bit versions.

Terminology Conventions

The term “GemStone” is used to refer to the server products GemStone/S 64 Bit and GemStone/S, and the GemStone family of products; the GemStone Smalltalk programming language; and may also be used to refer to the company.

Technical Support

Support Website

gemtalksystems.com

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

- ▶ **Documentation** for the current and for previous released versions of all GemTalk products, in PDF and HTML.
- ▶ **Product download** for the current and selected recent versions of GemTalk software.
- ▶ **Bugnotes**, identifying performance issues or error conditions that you may encounter when using a GemTalk product.
- ▶ **Supplemental Documentation** and **TechTips**, providing information and instructions that are not in the regular documentation.
- ▶ **Compatibility matrices**, listing supported platforms for GemTalk product versions.

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

Help Requests

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

Website: techsupport.gemtalksystems.com

Email: techsupport@gemtalksystems.com

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

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

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

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

24x7 Emergency Technical Support

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

Training and Consulting

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



Table of Contents

Chapter 1. Installing GemStone/S 64 Bit Version 3.6.2

Review the Installation Procedure	7
Check the System Requirements	8
Configure the Operating System	9
Install the GemStone Server.	11
Set the Environment	12
Set the GemStone Key File.	12
Verify TCP/IP	13
Define the NetLDI Service.	13
Run the Installation Script.	14
Complete GemStone Configuration	16
Change Passwords for Administrative Accounts.	16
Add GemStone User Accounts	17
Users must set environment.	17
Install the default TimeZone	17
Further Configuration and Administration	17

Chapter 2. Upgrading from GemStone/S 64 Bit 3.3.x or later

Keyfiles	19
Upgrade Procedure	20
Prior to Upgrade in existing application.	20
Prepare for Upgrade	20
Perform the Upgrade.	22
Post-upgrade Application Code Modifications	23
Make Backup	23
Configure GCI clients and GBS.	23

Chapter 3. Configuring GBS for GemStone/S 64 Bit

Installing GemStone/S 64 Bit Version 3.6.2

This chapter describes the procedure for installing GemStone/S 64 Bit™ version 3.6.2. We recommend that you set up GemStone initially on a single machine, to ensure that all the pieces work together. Further setup to run a distributed system are described in the *System Administration Guide*. You will need to adjust the installation process to meet your specific needs.

If you are upgrading to this release from a previous version of GemStone/S 64 Bit, follow the instructions in the appropriate later chapter of this Installation Guide. These upgrade instructions will provide details on steps that need to be taken before and after the installation described here.

Review the Installation Procedure

The following list summarizes the steps to install GemStone/S 64 Bit.

- ▶ Check the System Requirements. 8
- ▶ Install the GemStone Server 11
- ▶ Set the Environment 12
- ▶ Set the GemStone Key File 12
- ▶ Verify TCP/IP 13
- ▶ Define the NetLDI Service 13
- ▶ Run the Installation Script 14
- ▶ Change Passwords for Administrative Accounts 16
- ▶ Add GemStone User Accounts 17
- ▶ Install the default TimeZone 17

Check the System Requirements

Before you install GemStone/S 64 Bit, ensure that the following system requirements are satisfied. Additional system resources may be necessary to support large applications.

Platforms

- ▶ x86 64-bit system

SPARC distributions are available for development/debugging, not for production use.

RAM and Swap space

- ▶ While small installations can run on systems with only a few GB of physical RAM, increasing RAM is important for GemStone performance.
- ▶ Total swap space should be at least equal to the amount of RAM. We recommend installing twice as much swap space as RAM.

Due to the way GemStone uses memory, systems with insufficient swap space allocated have a risk of memory errors even if there is available RAM.

Disk space

- ▶ Space for the installed distribution files—you need approximately 800MB for GemStone/S 64 Bit, and additional space for other products.
- ▶ Additional disk space as required for your repository.
- ▶ The repository files should be located on a disk drive that does not contain swap space. Use of multiple disk drives is advisable for servers.

Operating system

- ▶ Solaris 10 on x86

Solaris on SPARC distributions are available for development/debugging, not for production use.

Debugger

A C debugger allows C-level stack traces when a GemStone error occurs, or when using the **pstack** command. While not required for GemStone execution, it is strongly recommended that a debugger be installed, so diagnostic error stacks will be available.

- ▶ Sun Dbx Debugger 8.1 SunOS_i386 2016/06/01

C/C++ Compiler

GemStone requires a C/C++ compiler only if you are developing C or C++ code for user actions or for a C or C++ application; as described in the *GemBuilder for C* manual. This compiler is required only for development work, not for execution.

- ▶ Solaris Studio 12.5
CC: Studio 12.5 Sun C++ 5.14 SunOS_i386 2016/05/31

X Windows

An X Windows server allows you to use GemStone's graphical VSD application on Solaris. Alternatively, GemStone statistical data may be viewed on Windows, by transferring the data files, or by mounting the file system on Windows. X Windows is not required for GemStone execution.

Configure the Operating System

The kernel must be configured to support shared memory and semaphores. See your operating system documentation for further information. These requirements apply both to server nodes and to client nodes (nodes that will run Gem processes).

The following suggested settings in `/etc/system` are typical of those used at GemStone, and most are intentionally set high to provide flexibility. These settings may not be suitable for your hardware configuration and system load.

On Solaris 10 and above, Sun recommends configuring shared memory and semaphores at the project level, rather than system wide. This provides greater flexibility, as well as avoiding the need to reboot after editing `/etc/system`. However, if the parameter settings are present in `/etc/system`, they are used as the default project settings. If you are not partitioning your system using projects, you can continue to configure these settings using `/etc/system`.

1. Shared memory maximum

The default for maximum shared memory segment is not adequate to run GemStone/S 64 Bit. We recommend a value larger than your desired Shared Page Cache size, and not more than 75% of your real memory.

For example, if you have 8192 MB of real memory:

$$8192 \text{ MB} * .75 = 6144 \text{ MB}$$

$$6144 \text{ MB} * 2^{20} = 6442450944 \text{ bytes}$$

The parameter in Solaris 10 and above is `project.max-shm-memory`, which controls project total memory use, not segment size. To configure this, use a command similar to the following, which sets the shared memory maximum to 6GB.

Solaris 10:

```
prctl -n project.max-shm-memory -v 6gb -r -i project "user.root"
```

Solaris 11:

```
projmod -sK 'process.max-shm-memory=(privileged,6G,deny)' default
```

In earlier versions of Solaris, `shmsys:shminfo_shmmax` set the maximum shared memory segment size, rather than limiting total memory use. To set this parameter using legacy-style Solaris configuration, in `/etc/system` include the following:

```
set shmsys:shminfo_shmmax=6442450944
```

2. Semaphores

The maximum number of semaphores per id (per semaphore set) limits the number of GemStone sessions that can log in to a particular Stone and connect to its shared page cache.

On the Stone's node, this parameter must provide **two** semaphores for each user who will log in to that Stone from any node plus an overhead of **four**. In distributed

systems, nodes that have only user sessions (Gems), but no Stones, must provide **two** semaphores for each user session on that node plus an overhead of **one**.

The number of semaphores actually requested for a particular shared page cache depends on the GemStone configuration file read by the process that starts the cache and is $(SHR_PAGE_CACHE_NUM_PROCS * 2) + 1$.

In Solaris 10 and above, the project parameter is `process.max-sem-nsems`. The default is 512 which is adequate for smaller-sized systems. For larger systems, use a command similar to the following, which set the value to 2048:

Solaris 10:

```
prctl -n project.max-sem-nsems -v 2048 -r -i project "user.root"
```

Solaris 11:

```
projmod -sK 'process.max-sem-nsems=(privileged,2048,deny)' default
```

In earlier versions of Solaris, this parameter was named `semsys:seminfo_semmsl` (Note that `semmsl` ends with a lowercase L, not a digit.). To set this, use a command similar to the following, which sets the value to 500:

```
set semsys:seminfo_semmsl=500
```

If you are adjusting settings using `/etc/system`, save the file, and reboot the system using `boot -r`.

The maximum number of semaphore identifiers in the system limits the number of GemStone shared page caches on the node; each shared page cache uses one identifier. This parameter must be large enough to allow for one shared page cache for each Stone running on the node, plus one for each cache running on a remote node.

The project parameter in Solaris 10 and above is `project.max-sem-ids`. The default is 128. This should be adequate for most installations.

In earlier versions of Solaris, the parameter is `semsys:seminfo_semni`.

3. PAM

If you are using UNIX authentication for GemStone logins, or if you run NetLDI as root with `setuid` (i.e. not in guest mode), you must have PAM (Pluggable Authentication Module) configured on the server. You may include a specific GemStone authorization service name, or allow the default "other" authentication definitions to be used.

PAM authentication definitions are in `/etc/pam.conf`. The specific GemStone service names are `gemstone.gem` for user authentication, and `gemstone.netldi` for a NetLDI running with authentication. The libraries that are specified in the stack depend on how you are configuring PAM to perform the authentication. The examples below are for PAM configured to invoke LDAP for authentication.

For example, to allow GemStone UNIX authentication, which uses PAM, to authenticate via LDAP:

```
gemstone.gem      auth requisite      pam_authtok_get.so.1
gemstone.gem      auth required        pam_ldap.so.1
```

For NetLDI authentication, again using LDAP:

```
gemstone.netldi  auth requisite      pam_authtok_get.so.1
gemstone.netldi  auth required        pam_ldap.so.1
```

If the following “other” stack is available in `/etc/pam.conf`, GemStone can use this to authenticate via LDAP:

```
other          auth requisite    pam_authtok_get.so.1
other          auth required    pam_ldap.so.1
```

Consult your System Administrators for more information on how authentication is handled on your system.

4. System clock

The system clock must be set to the correct time. When GemStone opens the repository at startup, it compares the current system time with the recorded checkpoint times; a system time earlier than the last checkpoint time may be taken as an indication of corrupted data and prevent GemStone from starting. The time comparisons use GMT.

5. TCP keepalive option

GemStone processes ordinarily use the `TCP keepalive` option to determine how long they will wait after communications activity ceases unexpectedly. This setting can be useful for reaping stale RPC Gems, but the OS default may not be appropriate for your system. For further information, refer to your operating system documentation.

Install the GemStone Server

Installing GemStone can be done as a regular user, but in order to set up shared security, some portions of the installation should be done when logged in as the root user. Other steps of the installation are done as the unix user who will be the GemStone administrative account.

In addition to the installation directory, these are the portions of the system that are affected by the installation of GemStone:

`/dev/rdisk`

Optional raw partitions for repository extents and transaction logs.

`/etc/services`

Internet services database, for NetLTI name lookup.

`/opt/gemstone`

Default location for server lock files, host name id file, and log files for GemStone network servers (NetLTIs). See the *System Administration Guide* for details.

`/tmp/gemstone`

Pipe file for the Stone repository monitor.

1. Log in as the GemStone administrator to the machine on which you are installing GemStone. This part of the installation should **not** be done as root, to ensure all the files are not owned by root.
2. Determine that adequate swap space is available.:


```
> /usr/sbin/swap -s
```
3. Select the drive on which you will install the GemStone software, and the installation directory on this drive, *InstallDir*. Make this directory the current working directory.

We recommend that you avoid choosing either an NFS-mounted partition or one containing UNIX swap space for the initial installation. Mounted partitions can result in executables running on the wrong machine and in file permission problems. Existence of swap space on the same drive can dramatically slow GemStone disk accesses.

4. GemStone/S 64 Bit is provided as a compressed file with a name similar to `GemStone64Bit3.6.2-i386.Solaris.zip`. Move this distribution file to the directory location in which GemStone will be installed, *InstallDir*.
5. Unzip the distribution file using `unzip`.
6. The *InstallDir* now contains a GemStone directory with a name similar to `GemStone64Bit3.6.2-i386.Solaris`.

In addition to subdirectories, this directory also contains the text file `version.txt`, which identifies this particular product and release of GemStone.

The GemStone server is now installed.

Set the Environment

Perform the following steps to properly configure the operating environment.

1. Set the environment variable GEMSTONE.
 - a. If more than one installation of any GemStone/S product resides on this machine, check for existing GemStone environment variables:

```
> env | grep GEM
```

All GemStone environment variables are displayed.
 - b. If any environment variables exist and are not appropriate for the new installation, you must specifically unset each one. For example:

```
> unset GEMSTONE GEMSTONE_SYS_CONF
```
 - c. Set the environment variable GEMSTONE to the *full pathname* (starting with a slash) of your new GemStone installation directory. For example:

```
> export GEMSTONE=InstallDir/GemStone64Bit3.6.2-i386.Solaris
```

Set the GemStone Key File

To run GemStone, you must have a key file for the correct version of GemStone/S 64 Bit and for the appropriate platform. The keyfile must be located where GemStone can find it on startup:

- ▶ A file specified by the KEYFILE configuration parameter in the configuration file used by the stone. This is not set by default, but may be defined to read a keyfile with any name in any location.
- ▶ `$GEMSTONE/sys/gemstone.key`

Licensed Customer key file

You may use a keyfile from any v3.6.x version with v3.6.2; keyfiles from 3.5.x and earlier are not valid with v3.6.2. If you are upgrading from v3.5.x or earlier, or you have questions

about your keyfile or license limits, email keyfiles@gemtalksystems.com, or contact GemTalk Technical Support.

Installing a keyfile

To specify the location and name of the keyfile using the KEYFILE configuration parameter, edit the configuration file that will be used by the v3.6.2 stone to include the location and name of the keyfile.

You may also put the keyfile in the default location, `$GEMSTONE/sys/gemstone.key`. This requires modifying the write permissions of the `$GEMSTONE/sys` directory; ensure you change this back to not writable, after this update.

Verify TCP/IP

To run GemStone, TCP/IP must be functioning, even if your machine is not connected to a network.

Verify that TCP/IP networking software is functioning (1 is the number 1):

```
> /usr/sbin/ping hostname 1 1
```

where *hostname* is the name of your machine. If **ping** responds with *hostname is alive*, TCP/IP is functioning.

Define the NetLDI Service

The NetLDI service, by default `gs64ldi`, can be defined in your system services database to return the NetLDI listening port. A NetLDI is required for some local and all remote sessions to log into GemStone, and it can be resolved by name or directly by port number. If you are defining NetLDI services by name, the same NetLDI service name and port number must be defined on the remote machines as on the Stone's node.

The following steps can be skipped if you will be using the NetLDI port rather than name. Using the port number requires that this same port be specified when the NetLDI is started up, and when sessions login. The details are described in the *System Administration Guide*.

If you are upgrading from a previous version, you may need to keep the NetLDI for that version running. In this case, select a distinct name and/or port for the NetLDI for GemStone/S 64 Bit 3.6.2.

1. Determine whether the `gs64ldi` service is already defined. How to do this will depend on how your system is set up. The GemStone distribution includes an executable that will allow you to do this:

```
unix> $GEMSTONE/install/getservbyname gs64ldi  
s_name=gs64ldi s_port = 50377 s_proto = tcp
```

If `gs64ldi` is defined, skip the rest of this procedure and continue with the installation at "Run the Installation Script" on page 14.

If it is not defined, continue performing this procedure.

2. Add an entry similar to the following to the system services database:

```
gs64ldi 50377/tcp #GemStone/S 64 Bit 3.6.2
```

Choose a port number that is not being used by another service. The port number should be in the range `49152 <= port <= 65535`, to conform to IANA standards (<http://www.iana.org/assignments/port-numbers>).

3. If several machines will be running GemStone, have the UNIX system administrator update the system services database for each machine. This includes Windows client machines as well as UNIX nodes. Note that the port number must be the same for every machine.

Run the Installation Script

The installation script verifies your environment, creates lock file directories, sets up the extent files, and can be used to configure certain types of security for multi-user systems. It is not required to run the installation script; however, if you are new to GemStone, or installing GemStone on a new server, running the install script is recommended.

For development systems without strict security requirements, it is recommended to run GemStone with the NetLDI in guest mode with captive account, which simplifies debugging. In this case, do **not** accept the default answers to the installation script questions regarding protecting the extent and running NetLDI as root.

1. Log in as root.

You can run the installation script as a non-root user, however, some options are not available as a non-root user. To ensure that the installation is successful, you must install as root. See the *System Administration Guide* for details on setting up GemStone server file security.

2. Invoke the installation script from the `install` subdirectory:

```
cd $GEMSTONE/install
./installgs
```

`installgs` is an interactive script that analyzes your system configuration and makes suggestions to guide you through installing GemStone on your machine. You can run the script multiple times; it will skip completed tasks.

NOTE

You can usually terminate execution of the installation script with Ctrl-C without risk to your files. When it is not safe to do so, the message Please do not interrupt appears on the screen.

During installation, you are asked several questions. The entire installation dialog is not reproduced here, but the main points are addressed. Some questions may not be asked, depending on answers to previous questions.

Whenever you are asked to answer “yes” or “no,” answer with y or n. When the script offers a default answer in square brackets (such as “[y]”), press Enter to accept the default.

Set up directories for server lock files and NetLDI logs?

The default location for server lock files and NetLDI log files is `/opt/gemstone`, although `/usr/gemstone` may be used in legacy installations. A different location can be specified using the environment variable

GEMSTONE_GLOBAL_DIR; however, all Gemstone processes that will interact on this machine must have this environment variable set to the same directory.

If the directory does not exist, the installation script offers to create /opt/gemstone and the subdirectories locks and log, and to set access (770) to these directories.

If you answer no to creating the directories, you must create them (or provide a symbolic link) before starting the server.

Set the owner and group for all the files in the distribution?

If you answer **yes**, the script will prompt you for the owner and group you want to use. Refer to Chapter 1 of the *System Administration Guide* for more information about setting owner and group permissions.

If you answer **no**, the permissions will remain the same as when the files were extracted from the distribution media.

Protect the repository file?

The default gives only the owner read and write access (600) through ordinary UNIX commands. Other users can read and write the repository through a GemStone session. If you choose not to protect the repository, the setuid bit is cleared from all executables, which causes them to run under ownership of the user who invokes them.

If you are running a development system without strict security requirements, we recommend **not** selecting the default.

Default: Set the repository permission to 600, and leave the setuid bit applied.

Allow NetLDI to Run as Root?

The NetLDI permits remote processes to interact with the repository. There are two ways to set up a NetLDI so that it can provide services to all GemStone users: it can run as root, or it can run in guest mode with a captive account.

- ▶ To run NetLDIs as root, accept the default “yes” response. Ownership of the NetLDI executable is changed to root, and the setuid bit is set. Any GemStone user will be able to start a NetLDI process that is accessible to all GemStone users because it will always run as root. For certain services, users will need to authenticate themselves by supplying a password.
- ▶ To run NetLDIs in guest mode with a captive account, answer “no” to the prompt, because those modes are not permitted if the NetLDI runs as root. “Guest mode” means that GemStone users do not have to supply a UNIX password to use NetLDI services. The “captive account” is an account that owns all processes the NetLDI starts; typically, it is the GemStone administrative account that owns the files. You must start the NetLDI while logged in as that account.

If you are running a development system without strict security requirements, we recommend **not** selecting the default.

Default: Change ownership of the netldi executable to root, and set its setuid bit.

Set up an Extent?

GemStone is distributed with a read-only copy of the initial repository in `$GEMSTONE/bin/extent0.dbf`. Before you can start GemStone, this file must be copied to a suitable location and made writable. The script offers to copy the file to its default location of `$GEMSTONE/data`.

If you are a new GemStone user, we recommend that you answer `y`. If you are an existing GemStone user, you might prefer to answer `n`, then copy the extent to a different location yourself, and edit the configuration file to reference this location. For more information, see the *System Administration Guide*.

Default: Place a writable copy of `extent0.dbf` in `$GEMSTONE/data`.

Start a NetLDI?

You can start a NetLDI that runs as root by answering yes to this prompt and the confirmation that follows.

If you want to start the NetLDI in guest mode with a captive account, you must do that after completing the installation. For more information about guest mode with captive account, see the *System Administration Guide*.

Default: Do not start a NetLDI at this time.

Start an Object Server?

As root, you cannot start an object server (Stone), but the script offers to start one as another user.

Default: Do not start an object server at this time.

3. Log out as root

After running `installgs`, log out as user root. Further work is done as the GemStone administrative user.

Complete GemStone Configuration

The following should be done by an administrative user, not as root.

Change Passwords for Administrative Accounts

GemStone comes with a number of built-in System user accounts, which are needed to perform administrative operations (such as adding application user accounts).

- ▶ The **DataCurator** account is used to perform system administration tasks.
- ▶ The **SystemUser** account ordinarily is used only for performing GemStone system upgrades.
- ▶ The **GcUser** account is used by the garbage collection task, which runs automatically as a separate login.

The initial password for these administrative accounts is `swordfish`.

Access to each of these accounts should be restricted; you should always change the passwords for these accounts, to provide basic security for your application.

The chapter entitled “User Accounts and Security” in the *System Administration Guide* tells you how to change the passwords.

Add GemStone User Accounts

For each of the users in your system, you should establish GemStone accounts, which involves creating an individual UserProfile in GemStone.

The chapter entitled “User Accounts and Security” in the *System Administration Guide* provides information on how create accounts for GemStone users, and the options for authentication. This task can be done by executing Smalltalk code, or using GemBuilder for Smalltalk tools. See the *GemBuilder for Smalltalk Users’s Guide* for information on the GUI tools in GemBuilder.

Users must set environment

After GemStone/S 64 Bit 3.6.2 has been installed, you should notify each person who will be using GemStone about the installation, and explain how to setup their environment.

Each user must:

- ▶ Set the environment variable GEMSTONE to the *full pathname* (starting with a slash) of the GemStone/S 64 Bit 3.6.2 directory.
- ▶ update their path to include the \$GEMSTONE/bin directory.
- ▶ Optionally, update the man path (MANPATH variable) to include the \$GEMSTONE/doc directory. GemStone provides man pages for utility functions.

These last two steps can be done using scripts that are part of the GemStone distribution. The directory \$GEMSTONE/bin contains the files `gemsetup.sh` and `gemsetup.csh`, which define the GemStone environment for users by modifying the PATH and MANPATH variables to include \$GEMSTONE/bin and \$GEMSTONE/doc, respectively.

For example:

```
export GEMSTONE=installdir
export PATH=$GEMSTONE/bin:$PATH
export MANPATH=$MANPATH:$GEMSTONE/doc
```

If the user will use GemStone frequently, consider adding these steps to the login shell initialization file.

Install the default TimeZone

GemStone/S 64 Bit is shipped with a default time zone of US Pacific. If you are in another Time Zone, edit the file `installtimezone.txt` in the GemStone upgrade directory, then file it in as SystemUser.

Further Configuration and Administration

This chapter has guided you through installation of GemStone/S 64 Bit 3.6.2, with the objective of getting a simple, default configuration up and running.

The next chapters explain the process of upgrading a previous version of GemStone/S 64 Bit to version 3.6.2; and Chapter 3 provides information on GemBuilder for Smalltalk.

For more information and details on customizing your GemStone object server, Gem client processes, and setting up distributed configurations, see the *System Administration Guide*.

Upgrading from GemStone/S 64 Bit 3.3.x or later

This chapter describes how to upgrade an existing GemStone/S 64 Bit 3.3.x, 3.4.x, 3.5.x, or 3.6.x installation to GemStone/S 64 Bit version 3.6.2.

To upgrade from GemStone/S 64 Bit version 3.2.x and earlier, you must first upgrade to a more recent version, and then upgrade to v3.6.2.

If you are using GemBuilder for Smalltalk (GBS), you also need to upgrade the client libraries that are used by GBS. You may also need to upgrade your version of GBS; versions of GBS earlier than 8.5 or 5.4.6 are not supported with v3.6.2. See Chapter 3 for details.

Keyfiles

You may use a keyfile from 3.6 or any v3.6.x version with v3.6.2. Keyfiles from 3.5.x and earlier are not valid with v3.6.2.

To acquire a keyfile for version 3.6.2, email keyfiles@gemtalksystems.com, or contact GemStone Technical Support, preferably providing your existing keyfile.

Keyfiles also manage access to GemConnect, GemBuilder for Java, and the X509-Secured GemStone feature. If you are using these add-on products, you must use a keyfile with the appropriate permissions.

Upgrade Procedure

The following list summarizes the steps necessary to perform the upgrade to GemStone/S 64 Bit v3.6.2.

- ▶ Prior to Upgrade in existing application. 20
- ▶ Prepare for Upgrade 20
- ▶ Perform the Upgrade 22
- ▶ Post-upgrade Application Code Modifications. 23
- ▶ Make Backup 23
- ▶ Configure GCI clients and GBS. 23

NOTE

The following instructions use the version number 3.6.1 to refer to the version you are upgrading from, and version number 3.6.2 indicate the target version you are upgrading to.

Prior to Upgrade in existing application

1. Check for use of deprecated methods

Verify that your application does not invoke any methods that were deprecated in previous releases, by enabling error or logging on deprecation in your existing repository. Deprecated methods are subject to removal in major releases; finding them before upgrading allows the deprecation messages to provide replacement instructions.

For details on finding deprecated methods, refer to the *Programming Guide for GemStone/S 64 Bit*.

2. File out modifications to GemStone classes

File out any modifications or additions you made to GemStone/S 64 Bit kernel class methods. For more information about fileout, see the *GemStone/S 64 Bit Topaz Programming Environment*.

You will need to carefully compare these changes with GemStone/S 64 Bit 3.6.2 kernel methods, and refer to the *Release Notes* for version 3.6.2 to determine whether your changes are still necessary or appropriate.

CAUTION

Any changes that you have made to the GemStone/S 64 Bit kernel classes will be lost during upgrade; you MUST file these out in order to preserve the changes in version 3.6.2.

Prepare for Upgrade

1. Install and configure GemStone/S 64 Bit 3.6.2

Install GemStone/S 64 Bit 3.6.2 to a new installation directory, separate from the installation directory for version 3.6.1, as described in Chapter 1, starting on page 7.

Configure GemStone/S 64 Bit 3.6.2 the way you expect to use it — that is, with the appropriate extent locations and sizes.

If you copy the configuration files from your previous version to the version 3.6.2, be sure to review any changes in configuration parameters to determine if changes are needed.

You should ensure that adequate space is available for extents, transaction logs, and a backup during the upgrade. You must provide space for the extents and transaction logs for both repositories, the old and the new.

2. Reset SystemUser password

Log in to the version 3.6.1 system as a user with OtherPassword privilege, such as DataCurator, and reset the SystemUser password to 'swordfish':

```
topaz 1> printit
(AllUsers userWithId: #SystemUser) password: 'swordfish' .
System commitTransaction.
%
```

The upgrade script logs in with the SystemUser account and the default password, and resets the password for DataCurator and GcUser.

3. Stop user activity

Log in to the version 3.6.1 system as a user with SessionAccess and SystemControl privileges, such as DataCurator, and halt all user activity on the repository.

```
topaz 1> printit
System stopUserSessions.
%
```

4. Shut down the repository

You may now shut down the Stone. At the UNIX command line:

```
stopstone stone361
```

where *stone361* is the name of the version 3.6.1 stone on this machine. The repository must be cleanly shut down to avoid needing recovery when it is restarted with the new version's executables.

5. Set up the version 3.6.2 environment.

Set the environment variables required for the upgrade. For example:

```
export GEMSTONE=InstallDir362
export PATH=$GEMSTONE/bin:$PATH
export upgradeLogDir=tempDir
```

where *InstallDir362* is the GemStone/S 64 Bit version 3.6.2 installation and *tempDir* is a temporary directory for which you have write permission.

6. Copy extent files

Copy your version 3.6.1 extent files into the location specified by the v3.6.2 configuration file option DBF_EXTENT_NAMES:

- a. Identify the complete set of extent files that are used by your 3.6.1 stone. This can be found by examining the configuration file for the version 3.6.1 repository, looking for the last entry for DBF_EXTENT_NAMES.

- b. The target location is the setting for `DBF_EXTENT_NAMES` in the version 3.6.2 installation. Copy each of these extent files to the target location.

For example:

```
cp InstallDir361/data/extent0.dbf InstallDir362/data
cp InstallDir361/data/extent1.dbf InstallDir362/data
cp InstallDir361/data/extent2.dbf InstallDir362/data
```

Before upgrading, ensure that there are no transaction logs from a previous version of GemStone/S 64 Bit in any of the transaction log locations specified in the configuration file that will be used by version 3.6.2.

Perform the Upgrade

1. Start the Stone

Start the 3.6.2 Stone on the 3.6.1 extents you just copied:

```
startstone stoneName362
```

2. Upgrade image

Ensure you are in a directory to which you have write permission, and run the upgrade script.

The upgrade is performed by the script `upgradeImage`. This script has optional switches to specify the stone name and to set to size of the `GEM_TEMPOBJ_CACHE_SIZE` used for the upgrade process.

```
upgradeImage [-h] [-c cacheSize] [-s stoneName]
```

-h prints this usage information.

-c *cacheSize* sets the size of the `GEM_TEMPOBJ_CACHE_SIZE`; if this is not used, the script will default to use a value of 100000.

-s *stoneName* sets the name of the running stone to upgrade; if this option is not used, the script will default to **gs64stone**.

For example,

```
upgradeImage -s stoneName362
```

The script will prompt you to press the return key to begin.

The script invokes subordinate scripts to complete the upgrade. The upgrade process will take some time. You can examine the progress, if desired, by examining the file `$upgradeLogDir/upgradeImage*.out`.

The script should complete with the message:

```
Upgrade completed. No errors detected.
```

If not, please preserve the Stone log file and the contents of `$upgradeLogDir`. Contact your internal GemStone support person or GemStone Technical Support.

3. Restore System Account passwords

Log in to GemStone/S 64 Bit version 3.6.2 as DataCurator or SystemUser, and change the password for SystemUser, DataCurator, and GcUser to a secure password, such as the passwords used for these accounts in v3.6.1. For example:

```
topaz 1> run
(AllUsers userWithId: 'SystemUser') password: '361Password'.
(AllUsers userWithId: 'GcUser') password: '361Password'.
(AllUsers userWithId: 'DataCurator') password: '361Password'.
System commitTransaction
%
```

where *361Password* is the account password used in version 3.6.1.

Post-upgrade Application Code Modifications

1. Reinstall any other GemStone products that modify kernel classes.

If you use GemConnect or GemBuilder for Java, you must reinstall the appropriate version of these products into your repository at this time.

To install, use the procedure in the *Installation Guide* for that product.

2. File in Kernel class changes

If you have modified any kernel class methods of the previous version or if you have added methods to kernel classes, carefully compare your changes with the changes in version 3.6.2 to see whether your changes are still necessary or appropriate. Carefully review the *Release Notes* for each intervening version, as well as examining code in the image.

If the kernel class changes are still applicable, file in the changes, verify that errorcount is 0, and commit.

Make Backup

1. Make backup

At this point, you should create a full backup of the upgraded repository.

Configure GCI clients and GBS

1. Recompile User Actions

It is recommended to recompile and relink User Action libraries.

2. Configure GBS

If you are using GBS clients, ensure you are running a supported version of GBS and client Smalltalk. You must use GBS version 8.5 or later for VW, or GBS 5.4.6 or later for VA, to connect to a GemStone/S 64 Bit v3.6.2 repository.

Configure GBS to use the version 3.6.2 client libraries. Depending on the GBS version you are upgrading from, the required libraries, library naming conventions, and the process GBS uses to identify the correct library to load may have changed.

See Chapter 3, 'Configuring GBS for GemStone/S 64 Bit' for details. If your GBS clients run on a different platform than your GemStone server, refer to the *Installation Guide* for that platform.

Configuring GBS for GemStone/S 64 Bit

The GemStone/S 64 Bit v3.6.2 server requires a compatible version of GBS; versions of GBS earlier than 8.5 are not supported with GemStone/S 64 Bit v3.6.2 or later.

Versions of GemBuilder for Smalltalk (GBS) that are compatible with GemStone/S 64 Bit v3.6.2 do not support Smalltalk client applications running on Solaris, AIX, or Macintosh.

GemStone/S 64 Bit v3.6.2 supports client Smalltalk/GBS applications running with VisualWorks on Linux and Windows, and with VA Smalltalk running on Windows. For a table of all supported GBS and client Smalltalk platforms, see the *GemStone/S 64 Bit Release Notes* for v3.6.2.

For instructions for updating GBS clients that are running on Linux, see the *GemStone/S 64 Bit Installation Guide* for that platform; for GBS clients running on Windows, see the *GemStone/S 64 Bit Windows Client Installation Guide*.