GHI Installation Guide

GPFS / HPSS Interface
Release 2.4

July 2015

Portions of this work were produced by Lawrence Livermore National Security, LLC, Lawrence Livermore National Laboratory (LLNL) under Contract No. DE-AC52-07NA27344 with the U.S. Department of Energy (DOE); by the University of California, Lawrence Berkeley National Laboratory (LBNL) under Contract No. DE-AC02-05CH11231 with DOE; by Los Alamos National Security, LLC, Los Alamos National Laboratory (LANL) under Contract No. DE-AC52-06NA25396 with DOE; by Sandia Corporation, Sandia National Laboratories (SNL) under Contract No. DE-AC04-94AL85000 with DOE; and by UT-Battelle, Oak Ridge National Laboratory (ORNL) under Contract No. DE-AC05-00OR22725 with DOE. The U.S. Government has certain reserved rights under its prime contracts with the Laboratories.

DISCLAIMER

Portions of this software were sponsored by an agency of the United States Government. Neither the United States, DOE, The Regents of the University of California, Los Alamos National Security, LLC, Lawrence Livermore National Security, LLC, Sandia Corporation, UT-Battelle, nor any of their employees, makes any warranty, express or implied, or assumes any liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights.

Printed in the United States of America.

Table of Contents

1. GHI Installation......1

1.1. Prepare for Installation......2

1.1.1. Check for Prerequisites......2
1.1.2. Check for incompatible software......3
1.1.3. Create DB2 Accounts for GHI Access to HPSS......3
1.1.4. Prepare the Installation Target Directory......4

1.2. Install Prerequisite Software......5

1.2.1. Install GPFS......5
1.2.2. Create HPSS Users and Groups......5
1.2.3. Install MIT Kerberos (If Using Kerberos Authentication)......5
1.2.4. Install DB2......6
1.2.5. DB2 Server Configuration......6
1.2.6. DB2 Client Configuration......8
1.2.6.1. Install DB2 Client......8
1.2.6.2. Set Up DB2 Permanent License......9

1.3. Install and Build the HPSS Client Source Code......10

1.4. Define GHI Environment Variables......10

1.5. Place GHI Software Into The Build Directory......10

1.6. Build GHI Source Code......11

1.7. Install HSI/GHI-HTAR......12

1.8. Post Installation Procedures......12

2. GHI CONFIGURATION......14

2.1. Configuration Of An Upgrade Installation Of GHI......14

2.1.1. Initial Configuration......14
2.1.2. DB2 Conversion......19
2.1.3. Final Configuration......20

2.2. Configuration of A New Installation of GHI......20

2.3. GHI Manual Page Setup......24

2.4. ILM Policies for HSM......25

2.4.1. migrate.policy......25
2.4.2. reset_incomplete_migration.policy and reset_incomplete_migration.ksh......25
2.4.3. recall.policy......25

2.5. GHI Backup Configuration......25

2.5.1. backup_migration.policy......26
2.5.2. backup_metadata.policy......26
2.5.3. backup_error.policy......26

2.6. GHI Startup and Threshold Limits Configurations......26

2.7. DB2 Backup Configuration......26

2.8. HSIGWD / GHI-HTAR......27

2.8.1. HSIGWD......27
2.8.2. GHI-HTAR......29

2.9. GHI Backups 30
Appendix A - Glossary of Terms and Acronyms 31
Appendix B - References..37
Appendix C - Developer Acknowledgments 38

LIST OF TABLES

Table 1 - Installation Package Sizes and Disk Requirements 2

Preface

About This Document

The GPFS/HPSS Interface (GHI) Installation Guide is for use at system installation time. It outlines the steps required to install and configure a GHI system.

Chapter 1 instructions of a GHI software installation.

Chapter 2 instructions of a GHI configuration.

Conventions Used in This Document

Example commands that should be typed as shown at a command line are proceeded by a not-to-be-typed percent sign ('%') and be presented in a boldface courier font:

```
% sample command
```

Example command output and example contents of ASCII files are presented in non-bolded courier font:

```
% cat sample_file
sample file line 1
sample file line 2
```

When a file listing or command output has been shortenned for display, an elipsis will be used to indicate missing lines:

```
% sample command
some command output
   . . .
more command output
```

Example interactive command input and output is presented in a courier font, with input to be supplied at the indicated point in command execution shown in bold:

```
% sample command
command output
prompt-1> some input
more command output
prompt-2> more input
still more command output
```

User input, whether on the command line or to an interactive command prompt, that's dynamic will be shown in italics:

```
% sample command /pathname
command output
prompt-1> XYZ <calculated value>
```

In the above, “sample command” and “XYZ” would be typed exactly as shown, while “/pathname” would be replaced by the appropriate pathname and “<calculated value>” replaced with the result of some specified calculation.

Any text preceded by a pound sign (‘#’) should be considered comment lines:

```
    # This is a comment
```

NOTE: Prior to the installation of The GPSS/HPSS Interface (GHI), a series of planning meetings should have been conducted to discuss the project’s scope and objectives, site mission, timelines, expectations, criteria for success, and project constraints. The “Planning Process” is not a subject of this Installation Guide and is done during the Proposal and/or System Engineering phase of the project. Refer to the *GHI Management Guide* for planning considerations. The planning process for GHI, as well as HPSS, should be done carefully (and prior to deployment) to ensure that the resulting system satisfies the site’s requirements and operates in an efficient manner.

1. GHI INSTALLATION

This chapter provides instructions and supporting information for installing the GHI prerequisite software, the GHI software, and DB2 software from the HPSS distribution media, and performing the GHI infrastructure configuration.

If doing an upgrade installation of GHI, do one of the following on the current GPFS cluster manager to create a file or data listing to be used to validate default configuration values present after running ***ghiupdate*** (see section 2.1.2):

- For current system version equal "2.2.x":

  ```
  % cp /var/hpss/ghi/config/templates/ghi.conf.template
  <copy_of_template>
  % mkdir <copy_of_config>
  % cp /var/hpss/ghi/etc/* <copy_of_config>
  ```

  <copy_of_template> and <copy_of_config> are any desired pathnames.

- For current system version equal "2.3.x":

  ```
  % ghilsfsdefaults
  % ghilscluster
  % ghilsfs           (for each configured file system)
  % ghilsiom          (for each configured file system)
  ```

Then, stop GHI prior to performing any of the processing steps outlined below. Refer to the *GHI Management Guide* for the running version of GHI for how to stop it.

The steps required to install a GHI system are listed below. Each step is discussed in more detail in the section referenced.

For a new installation, all steps listed in the remainder of this chapter need to be performed. For an upgrade installation, the first three sections should not be required, but they should be read and understood to ensure that the proper set-up is in place from the current configuration and what steps need to be performed if a variance is noted.

1. Prepare for Installation (Section 1.1).
2. Install Prerequisite Software (Section Error: Reference source not found).
3. Install and Build HPSS Client Application Programming Interface (API) Code (Section 1.3).
4. Define GHI Environmental Variables (Section 2.3).
5. Fetch GHI Software From Media (Section 1.4).
6. Build GHI Source Code (Section 1.5).
7. Install and Build GHI-HTAR and HSIGWD (Section 1.6)

8. Post Installation Procedures (Section 1.7).

All steps required to install and setup a GHI system require sufficient privileges, except where noted otherwise,. Privileges are normally acquired by logging into the system as *root* or another privleged user, or by issuing a “su -“ command.

The following distribution media is provided (possibly only ***GHI*** if doing an upgrade):

DB2: Obtain the HPSS software from your IBM HPSS customer support representative.

HPSS Mover/Client API: Obtain the HPSS software from your IBM HPSS customer support representative. The software is avaiable in both tar files and image files.

GHI: Obtain the GHI software from your IBM HPSS customer support representative. The software is available in the tar file format. This software includes the HSIGWD/GHI-HTAR package needed for aggregating files during migration. Only GHI-HTAR is compatible with GHI, is provided by IBM, and only the provided version should be used with GHI.

The GHI software package names and sizes for the supported platforms are as follows:

Platform	**GHI Package Name**	**Package Size**	**Space Requirement s**	**Package Description**
AIX/LINUX	ghi2.4.tar.gz	< 1 MB	15 MB	All GHI Components
AIX/LINUX	hsi.5.0.0.1g.tar.gz (part of ghi.2.4.tar.gz)	< 3 MB	<26 MB	HSIGWD and GHI-HTAR

Table 1 - Installation Package Sizes and Disk Requirements

1.1. Prepare for Installation

The following sections discuss the steps that need to be taken in order to prepare the system for GHI installation.

1.1.1. Check for Prerequisites

- Operating system: RHEL 6.4 on x86_64, or AIX 6100-07-05

  Linux / x86_64:

```
% uname -p
x86_64
% cat /etc/redhat-release
Red Hat Enterprise Linux Server release 6.4 (Santiago)
```

  AIX:

```
% oslevel -s
```

```
6100-07-05
```

- GPFS: 3.5 PTF16

  NOTE: Perform these steps if upgrading only.

  ```
  % mmfsadm <<<"dump version"|grep Build
  Build branch "3.5.0.16 ".
  ```

  or

  ```
  % mmfsadm

  Enter commands (type "help" or "?" for help):

  mmfsadm> dump version
     . . .
  Build branch "3.5.0.136 ".
     . . .
  mmfsadm> quit
  ```

- HPSS: 7.4.2

  Contact your HPSS Support representative to verify HPSS version.

- DB2: 10.5

  Contact your HPSS Support representative to verify DB2 version.

1.1.2. Check for incompatible software

GHI cannot run on the same GPFS cluster as any other HSM application which makes use of the DMAPI. This includes TSM. If another HSM managed file system is required it must be run on a separate cluster and remotely mounted on the GHI managed cluster. This incompatibility is due to GHI not always receiving DMAPI mount events when competing HSM applications are running.

1.1.3. Create DB2 Accounts for GHI Access to HPSS

The HPSS Core Server must have a UNIX user ID (uid) for ***hpssdmg*** and ***hpssdb***, and group ID (gid) for ***hpsssrvr*** and ***hpssdb***. If these do not exist, you must create them.

If the ***hpssdmg*** account needed to be created, the following also needs to be done on the HPSS Core Server to create the corresponding server account for ***hpssdmg*** in the HPSS ACL table:

```
# /opt/hpss/bin/hpss_server_acl

hsa> acl -n "Core Server"
1) PVL Mount Notification Interface (v1) 007ff347-e533-1cc6-b22d-
02608c2cedf4
```

```
2) Client Interface (v1) 32ba9692-4667-11d6-aa3a-0004ac49692b
3) Account Validation Interface (v1) 647f22a8-a1e9-11d3-a739-000001341966
4) Realtime Monitor Interface (v1) 80c9a256-2f13-11d3-a0c8-000001341966
Select an interface
Choose an item by number (RET to cancel):
> 3
hsa> show
perms - type - ID (name) - realm ID (realm)
===========================================
rw-c-d- - user - 303 (hpssssm) - 80960 (RS37.LOC.GOV)
rw-c-dt - user - 307 (hpssmps) - 80960 (RS37.LOC.GOV)
r--c--- - user - 308 (hpssfs)  - 80960 (RS37.LOC.GOV)
r--c--- - user - 314 (hpssftp) - 80960 (RS37.LOC.GOV)
------t - any_other
hsa> add user hpssdmg rwc
hsa> show
perms - type - ID (name) - realm ID (realm)
===========================================
rw-c-d- - user - 303 (hpssssm) - 80960 (RS37.LOC.GOV)
rw-c-dt - user - 307 (hpssmps) - 80960 (RS37.LOC.GOV)
r--c--- - user - 308 (hpssfs)  - 80960 (RS37.LOC.GOV)
rw-c--- - user - 309 (hpssdmg) - 80960 (RS37.LOC.GOV)
r--c--- - user - 314 (hpssftp) - 80960 (RS37.LOC.GOV)
------t - any_other
hsa> quit
```

If GHI will be configured for UNIX authentication, a UNIX User ID of ***hpssdmg*** and ***hpssdb***, and a Group ID of ***hpsssrvr*** are required on each node on which GHI will execute. All authentication/authorization is done using the ***hpssdmg*** principal. The numeric IDs must match those on the HPSS Core Server; which may be obtained from the "/etc/passwd" file on your HPSS Core Server. The $HOME directory for ***hpssdb*** must be set to "/var/hpss/hpssdb". The following shows these accounts configured (user/group IDs may differ on your system):

```
% grep hpss /etc/passwd
hpssdb:x:316:301::/var/hpss/hpssdb:/bin/bash
hpssdmg:x:311:300::/usr/local/hpssdmg:/bin/bash
% grep hpss /etc/group
hpsssrvr:x:300:
hpssdb:x:301:
```

1.1.4. Prepare the Installation Target Directory

The GHI software is installed in the "/opt/hpss/src/ghi" directory. Before installing the GHI software, make sure that the installation target directory either does not exist or is not being used. If it exists, recursively delete its contents. (You may wish to save the contents elsewhere prior to deletion.) If it does not exist, it will be created during the installation process.

The disk where the installation target directory resides must have enough space to hold all the GHI packages to be installed on this node and the to-be-created binaries. 50 megabytes should be sufficient. Additional space is needed for GPFS, and the DB2 and HPSS client code; refer to the documentation for these products to determine their space requirements.

WARNING: Do not use NFS mounted directories for installing or allocating space for GHI related components. Installing on NFS is problematic and the errors can be difficult to diagnose.

1.2. Install Prerequisite Software

This section provides an overview of how to install the prerequisite software to prepare for the upcoming GHI configuration. Verify that the correct software versions are obtained as described in the release notes for the version of GHI you are installing.

1.2.1. Install GPFS

Contact your IBM GPFS customer support representative to obtain and install GPFS. GPFS may use either *ssh* or *rsh* for comunications between nodes in the cluster; if using *ssh*, be certain the additional configuration gets done to allow password-less command execution (steps covered in the GPFS documentation).

Make sure that threshold processing is enabled for the cluster. To do this, issue the following command:

```
% mmchconfig enablelowspaceevents=yes
```

1.2.2. Create HPSS Users and Groups

GHI needs three users and two groups created on the GHI nodes in order for proper operation. The user and group ID numbers created must match the corresponding user and group ID numbers on the HPSS core server. The user and group ID numbers can be found in the /etc/passwd file on the core server. Home directories are the default HPSS install values.

User: hpss
Group: hpss
Home directory: /opt/hpss

User: hpssdb
Group: hpss
Home directory: /var/hpss/hpssdb

User: hpssdmg
Group: hpsssrvr
Home directory: /home/hpssdmg

1.2.3. Install MIT Kerberos (If Using Kerberos Authentication)

The capability to use MIT Kerberos authentication is provided in HPSS. However, IBM HPSS Service Agreements do not provide service or support for problem determination and resolution for MIT Kerberos. Kerberos maintenance/support must be customer-provided.

Kerberos is included in the Linux Operating System and should therefore not need to be installed.

1.2.4. Install DB2

GHI uses two DB2 tables in the HPSS Core Server database for each managed file system to manage data needed for maintaining of backups. The first table contains information on each backup of the GPFS metadata. GHI inserts an entry each time a backup is initiated, and updates it upon termination (whether success or failure). The second table contains entries which are made whenever a backed-up file is deleted from GPFS. The table is consulted whenever a GHI backup is deleted from HPSS to determine if the HPSS copies of GPFS files associated with the backed-up GPFS metadata can also be deleted or if the HPSS copy is referenced by other GHI backups.

GHI uses a third DB2 table in the HPSS Core Server database to keep track of all configured GHI file systems which are linked to “this” HPSS. Multiple GHI clusters may make use of the same HPSS, and they will share this third DB2 table.

To accomplish this, a new GHI database is created and configured on the Core Server and DB2 clients are installed on all GHI nodes will be Session nodes.

1.2.5. DB2 Server Configuration

If this is the first GHI installation on this HPSS, **the following steps must be performed on the HPSS Core Server node to add the new database:**

1. Source the database profile.

```
% source /var/hpss/hpssdb/sqllib/db2profile
```

2. Create the database

```
% db2 "create database hghi on /var/hpss/hpssdb"
```

3. Connect to the database and verify the version

```
% db2 connect to hghi

   Database Connection Information

 Database server        = DB2/LINUXX8664 10.5.x
 SQL authorization ID   = HPSS
 Local database alias   = HGHI
```

4. Create the buffer pool

```
% db2 "create bufferpool bp4k size 10000 pagesize 4K"
```

Note. A table entry is 20 bytes + the index. One entry is created in the table for each backup that is taken. In the above example, the bufferpool creates 4K * 10000 pages, it creates a space of 40000K.

5. Create the tablespace

   a. If the current HPSS system is using the original RAW logical volume devices for DB2 tablespace containers, do the following:

```
% db2 "create tablespace ghidb pagesize 4K managed \
      by database using (FILE ‘$file’ 10000) bufferpool
bp4k"
```

   where $file is the directory used to store the tablespace. It is recommended to use “/var/hpss/hpssdb/ghidb”

   b. Create DB2 tablespace containers. (Customize the example DB2 command below for the actual names and number of containers.)

```
% db2 "create large tablespace ghidb pagesize 4K managed by
database \
      using (FILE '/hpss1_tbspc1/dms/hpssdb/hghi/ghidb'
10000, \
      FILE '/hpss1_tbspc2/dms/hpssdb/hghi/ghidb' 10000, \
      FILE '/hpss1_tbspc3/dms/hpssdb/hghi/ghidb' 10000, \
      FILE '/hpss1_tbspc4/dms/hpssdb/hghi/ghidb' 10000) \
      EXTENTSIZE 128 \
      PREFETCHSIZE AUTOMATIC \
      BUFFERPOOL bp4k \
      OVERHEAD 7.500000 \
      TRANSFERRATE 0.060000 \
      AUTORESIZE YES \
      MAXSIZE NONE \
      NO FILE SYSTEM CACHING \
      DROPPED TABLE RECOVERY ON"
```

6. Grant connect to the database

```
% db2 "grant connect on database to user hpss"
% db2 "grant createtab on database to user hpss"
% db2 "grant dbadm on database to user hpss"
```

7. Configure the primary, secondary logs, log archives, log file size, and number of logs similar to the standard of the HPSS databases

```
% db2 "update db cfg for hghi using NEWLOGPATH
/<primrary_log_path>/hghi"

% db2 "update db cfg for hghi using MIRRORLOGPATH
/<secondary_log_path>/hghi"

% db2 "update db cfg for hghi using AUTO_MAINT off"

% db2 "update db cfg for hghi using AUTO_RUNSTATS off"

% db2 "update db cfg for hghi using AUTO_TBL_MAINT off"
% db2 "update db cfg for hghi using LOGARCHMETH1 DISK:/
<primrary_log_archive_path>/hghi/"
```

```
% db2 "update db cfg for hghi using LOGARCHMETH2 DISK:/
<secondary_log_archive_path>/hghi/"
```

```
% db2 "update db cfg for hghi using LOGFILSIZ 25000"
```

```
% db2 "update db cfg for hghi using LOGPRIMARY 10"
```

```
% db2 "update db cfg for hghi using LOGSECOND -1"
```

Use the following table to determine how to set LOGBUFSZ:

Machine Memory	**LOGBUFSZ <Table Value>**
< 16 GB RAM	**4096**
16-64 GB RAM	**8192**
> 64 GB RAM	**16384**

Table 2 – LOGBUFSZ sizing

```
% db2 "update db cfg for hghi using LOGBUFSZ <table value>"
```

```
% db2 "update db cfg for hghi using DFT_QUERYOPT 2"
```

8. Disconnect from the database

```
% db2 disconnect all
```

1.2.6. DB2 Client Configuration

Perform the following steps on each potential GPFS quorum node (all nodes which include “quorum” in the “Designation” column of the output of a GPFS *mmlscluster* command):

- Install the DB2 Client.
- Set up DB2 permanent license.

A DB2 client does not need to be installed on any other GHI nodes in the cluster. The following sections describe the procedures to perform these steps.

1.2.6.1. Install DB2 Client

Refer to the *DB2 Command Reference* document on how to install the DB2 client.

1.2.6.2. **Set Up DB2 Permanent License**

This section describes the procedure to create the DB2 permanent license and to specify the number of processors licensed for the root subsystem machine.

To create a permanent DB2 license, issue the following commands:

```
% cd /opt/ibm/db2/V10.5/adm    (on AIX, substitute 'IBM' for 'ibm')
% ./db2licm -a <path name to DB2 generic license file>
```

The generic DB2 license file ("*/db2/license/db2ese.lic") can be found on the DB2 Installation CD or image. It can also be obtained by contacting your HPSS customer support representative.

To update the license with the appropriate number of processors, issue the following command:

```
% ./db2licm -n db2ese <number of processors>
```

Refer to the *DB2 Command Reference* document for more information on how to use the ***db2licm*** utility to manage the DB2 license.

The following steps are to be performed on each of the nodes that have a DB2 client installed to gain access to the new GHI database:

1. Create an instance.

   ```
   % /opt/ibm/db2/V10.5/instance/db2icrt -a SERVER \
         -s client -u hpssdb hpssdb    (on AIX, substitute 'IBM' for 'ibm')
   ```

2. If, for some reason, the $HOME directory for ***hpssdb*** is not set to "/var/hpss/hpssdb", issue the following command:

   ```
   % ln -s <hpssdb_home> /var/hpss/hpssdb
   ```

   Where: <hpssdb_home> is the assigned $HOME directory for ***hpssdb***.

3. Source the database profile (don't miss the 'dot'!)

   ```
   % source /var/hpss/hpssdb/sqllib/db2profile
   ```

4. Catalog the database.

   ```
   % db2 catalog tcpip node $NODE remote $HPSS_CORE \
         server $PORT
   % db2 catalog db hghi as hghi at node $NODE
   ```

   Where:

   $NODE = unique name; recommend using short host name

   $HPSS_CORE = HPSS core server

   $PORT = DB2 port defined in /etc/services on the HPSS core server

5. Recycle the HPSS Core Server to allow DB2 to pick-up the changes:

```
% rc.hpss stop
% rc.hpss start
```

1.3. Install and Build the HPSS Client Source Code

Follow the instruction in the *HPSS Install Guide* section “Install and Build HPSS” to either construct and build the HPSS Client source tree and install the HPSS Client, or to install the HPSS RPMs.

Once the installation of HPSS is complete, secure copy the mm.keytab from the core sever to the client. Execute this command on the HPSS Client:

```
% scp <user>@<HPSS core>:/var/hpss/etc/mm.keytab /var/hpss/etc/
```

1.4. Place GHI Software Into The Build Directory

The GHI tar file must be extracted into the “/opt/hpss/src/ghi” directory. To do this, issue the following commands:

```
% mkdir -p /opt/hpss/src/ghi
% cd /opt/hpss/src/ghi
% tar -xzvf <path to "ghi2.4.tar.gz">
```

Skip the ‘mkdir’ if the directory already exists (which will be the case if doing an upgrade installation over the current installation, but be sure to clean out the directory prior to issuing the “tar -x...”).

If this is to be a heterogeneous cluster (mixed hardware type and/or operating system, e.g., RHEL and AIX, or RHEL x86 and RHEL ppc) repeat the above on one node for each hardware/OS combination. Make note of the node(s) used: they will be used as parameters to the ***ghicrcluster*** or ***ghiupdate*** command (section 2).

The node (or at least one of the nodes if the cluster is heterogeneous) on which the GHI source is to be compiled in the next step (section 1.5) must either be the GPFS cluster manager or able to become it via the GPFS *mmchmgr* command.

1.5. Build GHI Source Code

The steps involved in building the GHI source code depend on whether this is a new installation or an upgrade of GHI. They should be done on each node on which the GHI source code was placed in the previous step (section 1.4) and from within the build directory (“/opt/hpss/src/ghi”).

- For a new installation of GHI, issue the following command:

```
% make
```

- For an upgrade of GHI, issue the following commands:

```
% make clean
% make clobber
% make
```

The display of “info” or “warning” messages during execution of the above command(s) is acceptable and should not be an indication that the compiled code will not execute properly. Any “error” messages must be addressed.

1.6. **Install HSI/GHI-HTAR**

The HSI package must be installed on each of the GHI nodes that are to be configured as I/O Manager nodes. If all nodes are homogenous, the package can be installed on a single node and the executables distributed to the remainder of the I/O Manager nodes. In a homegenous environment, when HSI is compiled for Kerberos make sure that all the GHI machines have the same version of Kerberos installed. Distributing the binaries without installing Kerberos in this case would cause runtime errors. In a heterogenous environment, the HSI package must be installed on one of each type of I/O Manager node and distributed to others of the same type.

The HSI package must also be installed on the HPSS Core Server node in conjunction with the installation and configuration of the HSI Gateway Daemon. Typical configuration is to install the HSI package in the “/var/hpss/hsi” directory, but it can be placed in any desired directory.

GHI-HTAR source code that is compatible with GHI is provided by IBM, and only the provided version should be used with GHI.

Configuration of GHI-HTAR is covered in section 2.9.

1.7. **Post Installation Procedures**

After the GHI software has been installed,

verify the following directories have been created:

- /opt/hpss/src/ghi
- /opt/hpss/src/ghi/man
- /opt/hpss/bin
- /opt/hpss/lib

- /var/hpss/ghi
- /var/hpss/ghi/policy
- /var/hpss/ghi/config
- /var/hpss/ghi/config/templates

If this is a new GHI installation, enable DMAPI processing for each to-be GHI-managed file system:

> Unmount the file system. Then, enter the following command to enable DMAPI processing:
>
> ```
> % mmchfs <file system> -z yes
> ```

Note: When DMAPI is enabled, the GHI Mount Daemon is required to mount the file system.

2. GHI CONFIGURATION

The following sections describe how to configure GHI after it has been installed on one node in the cluster (or one of each type of node in a heterogeneous cluster). Begining with GHI version 2.3, configuration is performed with command-line tools -- the various configuration files are no longer directly edited to effect changes. Also, distribution of needed files is performed automatically.

The steps required to configure GHI depend upon whether this is a new or an upgrade installation of GHI. If doing an upgrade installation, follow the procedures outlined in section 2.1. If configurating a new installation, follow the procedures outlined in section 2.2.

For a new installation, all steps listed in section 2.4 through the remainder of this document also need to be performed. For an upgrade installation, with the exception of handling incompletely-migrated files detailed in section 2.5.2, and HTAR-related processing detailed in section 2.9, the remaining steps may not apply. But, the remainder of this document should be read and understood to ensure that the proper set-up is in place from the current configuration and what steps need to be performed if a variance is noted.

2.1. Configuration Of An Upgrade Installation Of GHI

Configuration of an upgrade installion of GHI requires three steps. The first is to execute ***ghiupdate*** to format the previous configuration information to be compatible with the current version of GHI. After that has been completed, the GHI garbage collection (GC) DB2 tables will need to be made compatible with this release of GHI. Finally, GHI is started and the updated configuration inspected and any necessary changes made.

2.1.1. Initial Configuration

Run ***ghiupdate*** on the current GPFS cluster manager to bring the existing GHI configuration up to compatibility with the newly-installed version of GHI. GPFS must be running to update the GHI configuration. Except for IOMs, which are always running, GHI must not be running. If GHI is running, ***ghiupdate*** will report an error and abort. If the current GPFS cluster manager is not the node (or one of the nodes) on which the steps outlined in section 1.5 were performed, issue a GPFS *mmchmgr* command to set the cluster manager to one of these nodes.

ghiupdate is invoked as follows:

```
% ghiupdate [-bdTv] <OS/HW_list>
```

The options “[-bdTv]” are optional. <OS/HW_list> is one to three pairs of parameters, used to define the sources for distributing executable (binary) files to effect the update.

Each pair consists of a node type and a node name. The node type is one of "--aix" for AIX, "--x86_64" for Linux running on x86/64-bit hardware, and "--ppc_64" for Linux on a PowerPC platform. These are the machines on which the source code was built in section 1.4. Node names are the node on which the applicable source code was compiled. A node type and name may be used only once, i.e., the same node cannot serve as binary source for both AIX and Linux/PPC. Node names must the Admin node as listed by a GPFS '*mmlscluster'* command.

The four options which may be specified are:

-T – invoke ***ghiupdate*** in "test mode", meaning that all processing steps which do not alter any system or GHI files or restart any running processes will be attempted and the results reported.

-b – keep backup copies of all processed GHI and system files, which are normally discarded when the update completes successfully.

-d – delete all references to nodes not known to GPFS, i.e., nodes not shown as an Admin node as listed by a GPFS *mmlscluster* command.. If this option is not specified, any reference to an unknown node will be treated as a fatal error.

-v – provide a step-by-step (verbose) listing of processing steps as they are executed.

Any combination of options may be specified, e.g., "-bdTv", will have ***ghiupdate*** output verbosely, deleting unknown nodes from the configuration, and not deleting its backup copies of GHI and system files -- all in test mode.

Given the above, the following would be a valid ***ghiupdate*** command:

```
% ghiupdate -dv --aix newyork-p1 --ppc_64 chicago-p1
```

It will produce verbose output while effecting an update to the current release level of GHI, using node "newyork-p1" as the source for AIX code, and "chicago-p1" as the source for Linux/PPC code (and report error if there exists in the GHI cluster a node with some other OS/hardware combination). Any nodes listed in the to-be-updated GHI configuration which are not currently known to GPFS as Admin nodes will not appear in the updated configuration.

It is highly suggested to run ***ghiupdate*** twice, the first time using option '-T' to run in test mode. When runing in test mode, ***ghiupdate*** will go thru all the processing steps required to effect the update -- except for putting the "finished product" into production. The output will begin with:

```
*** TEST-ONLY ***
  Commands will be generated but not executed:
```

and will subsequently indicate application of whatever updates are required to the GHI configuration to bring it to compatibility, but the updates are not actually applied. If the test mode runs to a successful completion, "Done." will be displayed, after which

ghiupdate can be re-executed without the '-T'. If the first run results in an error, address the indicated errors and try another run with '-T'.

ghiupdate includes code to restore the cluster to its pre-existing configuration in the event of an error. (The back-out steps will also be shown if running in verbose mode.) However, if it is desired to have an additional level of backup in place, the following files and directories should be backed up prior to executing ***ghiupdate***:

• /var/hpss/ghi/config/	on GPFS cluster manager
• /var/hpss/ghi/etc/	on GPFS cluster manager
• /var/hpss/ghi/policy/	on GPFS cluster manager
• /etc/inittab	on every node with an IOM
• /etc/services	on every node with an IOM
• /etc/inetd.conf	on every AIX node with an IOM
• /etc/xinetd.d/	on every Linux node with an IOM

Part of ***ghiupdate***'s processing is to distribute the contents of $HPSS_BIN (nominally "/opt/hpss/bin") to all GHI nodes. It does this on each node by first issuing a "rm -f" for "$HPSS_BIN/ghi_io*" to avoid getting a "Text file busy" error if an IOM ("/opt/hpss/bin/ghi_iom") happens to be running on that node. Then, the command to accomplish the copy operation is issued. These two commands are not "backed-up", so any error-correction will need to be done by hand. (Both commands are displayed when running in verbose mode.)

ghiupdate will provide a running commentary on its processing steps, emding with "Done." if all goes without error, as in the following example, which shows verbose output:

```
% ghiupdate -v --aix newyork-p3a

*** UPDATE GHI CONFIGURATION ***
  -- CONFIGURATION CHANGES MAY BE MADE ***
OK to proceed with the above? (y/n) y
Determining system type of newyork-p3a.clearlake.ibm.com
  /usr/bin/ssh -f newyork-p3a.clearlake.ibm.com uname -a
2>/dev/null
GPFS cluster manager is newyork-p3a
Checking to see if GHI is up...
  ps -ef | grep ghi_pm | grep -v grep
Validating GHI nodes file...
  GHI node 'newyork-p3a' being renamed to 'newyork-
p3a.clearlake.ibm.com'
  GHI node 'newyork-p4a' being renamed to 'newyork-
p4a.clearlake.ibm.com'
Distributing updated GHI node list to all GHI nodes...
  Backing-up newyork-
p3a.clearlake.ibm.com:/var/hpss/ghi/etc/ghinode.conf
    /usr/bin/ssh -f newyork-p3a.clearlake.ibm.com cp
```

```
/var/hpss/ghi/etc/ghinode.conf /var/hpss/ghi/tmp/ghinode.conf
  /usr/bin/scp -B /var/hpss/ghi/tmp/ghinode.conf_upd newyork-
p3a.clearlake.ibm.com:/var/hpss/ghi/etc/ghinode.conf
     . . .
  Delete newyork-p4a.clearlake.ibm.com --
/var/hpss/ghi/tmp/ghi_nybfs2.conf
Done.
```

For comparison, here’s the same example without the ‘-v’ option:

```
% ghiupdate --aix newyork-p3a

*** UPDATE GHI CONFIGURATION ***
  -- CONFIGURATION CHANGES MAY BE MADE ***
OK to proceed with the above? (y/n) y
Checking to see if GHI is up...
Validating GHI nodes file...
Distributing updated GHI node list to all GHI nodes...
     . . .
Done.
```

In this final example, ***ghiupdate*** is executed while GPFS is not running:

```
% ghiupdate -v --x86_64 miami
Can't find GPFS cluster manager!
*** No changes have been made ***
```

Notice that it did not finish with “Done.”

An issue which may arise and which will require human intervention to resolve is evidenced by ***ghiupdate*** displaying the following message:

```
Can't resolve '?XXX' data to create FS entry ‘FS_name’ field
‘field_name’
```

This occurs when ***ghiupdate*** has finished reading the existing configuration for the indicated *FS_name* and did not find data needed for building the default configuration, where ‘?XXX’ is ‘?UID’, ‘?FS_Name’, ‘?Mount_Point’, ‘?SD_Port’, or ‘?ED_Port’. The ‘?XXX’ is a referrence to a value in the existing configuration. For example, the default configuration contains:

```
IOM Monitor Output Path = ?Mount_Point/scratch/mon/mon_iom.out
```

This means that if ***ghiupdate*** needs to create an FS entry for the “IOM Monitor Output Path”, the actual pathname is generated by using the text to the right of the ‘=’ after substituting ‘?Mount_point’ with the actual mount point for the FS. If the actual mount point happenned to be “/gpfs/ghi/disk2”, the pathname to be configured for the FS would be “/gpfs/ghi/disk2/scratch/mon/mon_iom.out”. If the mount point entry is missing from the existing configuration, ***ghiupdate*** cannot fill in the actual mount point. It will display a message and terminate. The required human intervention is to edit the existing configuration file, "/var/hpss/ghi/etc/ghi.conf", to add the missing "Mount Point" entry in the stanza for *FS_name*. Then, re-run ***ghiupdate***.

A second situation which might arise which will require operator intervention is if multiple GHI clusters are linked to the same HPSS and file systems on separate clusters, e.g., "ghi_fs1" and "ghi_fs2", share the same mount point, e.g., "/ghi". There has always been a restriction that GHI file systems which are linked to the same HPSS cannot be identically-named, biut the restriction against identically-named mount points is new for GHI 2.4. If ***ghiupdate*** is being executed on the second (or later) cluster to connect to "this" HPSS, it may find that the mount point of a file system on "this" GHI cluster is already configured to another GHI cluster, and abort with the following message to the user:

```
MP '<mount_point>' already configured on cluster <cluster_name>
```

where, '<mount_point>' is the offending mount point and <cluster_name> is where the mount point exists, e.g., "/ghi" and "prod_cluster". The mount point needs to be changed before the upgrade can be re-tried. To do this, un-mount the FS on all nodes in the cluster, and then change its mount point in both the GPFS and GHI configurations, as shown in the following example to change the mount point of file system "ghi_fs" from "/ghi" to "/ghi1":

```
% mmunmount /ghi –a
% mmchfs ghi_fs –T /ghi1
% vi /var/hpss/ghi/etc/ghi.conf
   <look for the line "ghi_fs = {">
   <Directly under it should be "Mount Point = /ghi">
   <Change "/ghi" to "/ghi1" so that the line reads "Mount Point
= /ghi1">
   <Save the file>
```

GPFS will delete the old mount point ("/ghi") and create the new mount point ("/ghi1"), either immediately or when the FS ("ghi_fs") is next mounted. Users will need to be advised of the new mount point and make necessary changes in their GPFS policies, scripts, procedures, etc. The GHI system policies in "/var/hpss/ghi/policy/<*FS_name*>" (e.g., "/var/hpss/ghi/policy/ghi_fs") will also need to be modified to use the new mount point. Be sure the updates are made on all GHI nodes for GHI system policies and where required for user policies, scripts, procedures, etc.

Also, the directory which holds the file containing the default configuration values, which is "/var/hpss/ghi/config/templates//", may contain no longer needed files. While these extra files have no effect on system operation, they have been a source of questions as to why they don't get updated. "/var/hpss/ghi/config/templates/" is required to contain just one file: "ghi.conf.template". Any other files in this directory may be safely deleted from any and all GHI nodes if desired.

2.1.2. DB2 Conversion

Once ***ghiupdate*** has completed sucessfully, the GHI garbage collection (GC) DB2 tables for each file system will need to be made to be compatible with this release of GHI. It is highy recomended that a backup of the DB2 tables for the filesystems be taken prior to starting the conversion. No special procedures are required for the backup. Whatever procedure a site uses to take a DB2 backup ay be used.

Do the following procedure to effect the conversion. This needs to be done for each GHI-managed file system. Where a DB2 command shows "GC_<FS>", the <FS> is to be replaced with the name of the FS, e.g., "GC_ghifs1" if the name of the FS is "ghifs1". If a DB2 ALTER, REORG, or DELETE command completes successfully, the following will be displayed:

```
DB20000I  The SQL command completed successfully.
```

- Connect to DB2 on the core server. (This need not be done for each file system since the connection should remain open.)

```
% source /var/hpss/hpssdb/sqllib/db2profile
% db2 connect to hghi
```

- Add constraints to ensure that SOID / ORDINAL pairs are unique.

```
% db2 ALTER TABLE GC_<FS> ALTER COLUMN SOID SET NOT NULL \
  ALTER COLUMN ORDINAL SET NOT NULL
% db2 REORG TABLE GC_<FS>
% db2 "ALTER TABLE GC_<FS> ADD PRIMARY KEY ( SOID , ORDINAL )"
```

- If the above "ADD PRIMARY KEY" command doesn't complete successfully, it indicates that there exists at least one SOID / ORDINAL combination which is not unique, and this will need to be corrected. Do so with the following command and then execute the "ADD PRIMARY KEY" command a second time.

```
% db2 "DELETE FROM ( SELECT ROWNUMBER() OVER \
  (PARTITION BY SOID , ORDINAL) AS RN FROM GC_<FS> ) AS A WHERE
RN > 1"
```

  The above command need not be done if the "ADD PRIMARY KEY" command completes successfully.

- Add two new columns to the GC table.

```
% db2 ALTER TABLE GC_<FS> ADD STATE INTEGER not null default 2 \
  ADD COUNT BIGINT not null default -1
```

2.1.3. Final Configuration

Once the DB2 conversion has completed sucessfully, start GHI with the command:

```
% ghistartup -g
```

Then, run the ***ghilsfsdefaults*** command and compare its output to the contents of <copy_of_template> or to the ***ghilsfsdefaults*** data listing created prior to shutting down the previous version of GHI (section 1). (The format will differ between the ***ghilsfsdefaults*** output and the contents of <copy_of_template>, but matching-up the actual configuration values should not be difficult.) If desired, run ***ghichfsdefaults*** to set a default value(s) to the value it had in the previous version of GHI. Note that default values might have been changed from earlier versions of GHI, so it is not necessarily an indication that ***ghichfsdefaults*** needs to be executed just because a difference in default values was found. Default values may also have been added or deleted.

Run the ***ghilscluster*** command, and the ***ghilsfs*** and ***ghilsiom*** commands for each configured file system and compare the output to the files in <copy_of_config> or to the previous version ***ghilscluster***, ***ghilsfs***, and ***ghilsiom*** output. Run the appropriate ***ghichcluster***, ***ghichfs***, ***ghiaddiom***, ***ghichiom***, or ***ghideliom*** command(s) to make any changes to configuration values which may be necessary. Any ***ghilsfs*** value which was at the defaut in the priveous version of GHI and now shows to not be at the default probably needs to be updated to reflect the new default value.

Note that, beginning with GHI 2.4, there are no more I/O Agents (IOAs).

All of the GHI commands discussed in this section are fully-documented in the GHI *Management Guide*.

2.2. Configuration of A New Installation of GHI

Configuring a new installation of GHI requires three steps:

1. Define the overall cluster configuration. This includes the nodes which will be known to GHI (not necessarily all nodes known to GPFS), and logging parameters. This is accomplished via command ***ghicrcluster***. ***ghicrcluster*** must be run on the node used to perform the steps outlined in section 1.4 (or one of the nodes on which source code was compiled if the cluster is a heterogeneous cluster). GPFS must be running.
2. Start GHI and define each GPFS filesystem to be known to GHI. FS configuration data includes HPSS aggregation and class-of-service, HPSS data and backup locations, as well as operating parameters for the GHI ED, SD and IOM(s) for each FS. This is accomplished via command ***ghiaddfs*** for each FS to be configured, which may be issued from any node in the cluster. Filesystems to be defined must not be mounted in GPFS when the ***ghiaddfs*** command is issued. ***ghiaddfs*** will supply default values for most of these parameters, which may then be altered as necesary with ***ghichfs***.
3. Lastly, for each filesystem, the associated IOMs are defined. Per-IOM configuration includes the node, communication parameters, and expected transfer rate. Each IOM, for each FS, is configured via command ***ghiaddiom***, which may be issued from any node in the cluster.

All of the GHI commands discussed in this section are fully-documented in the GHI *Management Guide*.

Step (1) must be completed for at least all nodes which may be GHI session nodes, i.e., for all nodes which include “manager” in the “Designation” column of the output of a GPFS *mmlscluster* command. GHI is then started, and steps (2) and (3) perfomed while GHI is running, and can be performed iteratively. Nodes can be added to the configuration at any time via command ***ghiaddnode*** (which also may be issued from any node in the cluster).

Begin step (1) by determining the nodes to be included in the initial configuration. They can either be specified on the ***ghicrcluster*** comand line or listed in a file to be specified with the ***ghicrcluster*** comand. If nodes are to be listed in a file, they are separated by white-space, i.e., blanks, tabs, or newlines, e.g.:

```
node1.here.com node2.here.com node3.here.com
bigbox
bluebox
```

Node names need not be fully-qualified. But, they must be Admin node names as listed by a GPFS ‘mmlscluster’ command. Be sure to include any node which may be a GHI session node. These nodes are those for which the “Designation” column of ‘mmlscluster’ output contains “manager”. To create the configuration, execute the appropriate form of the ***ghicrcluster*** command, either specifying the nodes directly on the command line:

```
% ghicrcluster -v --<OS/HW_list> node1.here.com node2.here.com …
bigbox bluebox
```

or using a node-list file named “node_list”:

```
% ghicrcluster -vN --<OS/HW_list> node_list
```

The ‘-v’ enables verbose output, which provides a step-by-step commentary of processing. Its use is highly recommended because it is very helpfull in tracking down execution errors. All the GHI configuration commands take an option ‘-v’. When executed, ***ghicrcluster*** will create operational versions of the various configuration files and push these, along with all the compiled GHI source code (based on “<OS/HW_list>”), to each node specified on the command line or listed in the nodes file.

<OS/HW_list> is one to three pairs of parameters, used to define the sources for distributing executable (binary) files. Each pair consists of a node type and a node name. The node type is one of “--aix” for AIX, “--x86_64” for Linux running on x86/64-bit hardware, and “--ppc_64” for Linux on a PowerPC platform. These are the machines on which the source code was built in section 1.4. Node names are the node on which the applicable source code was compiled. A node type and name may be used only once, i.e., the same node cannot serve as binary source for both AIX and Linux/PPC, nor may, for example, two AIX nodes be named. Node names must the Admin node as listed by a GPFS ‘mmlscluster’ command.

An example of a ***ghicrcluster*** command to create a six-node cluster; the x86/64 binaries will come from node “ss01”:

```
% ghicrcluster -v –x86_64 ss01 ss01 ss02 io01 io02 io03 io04
```

If all proceeds without error, “Done.” will be displayed. Otherwise, the following will be displayed:

```
Please correct problems which prevented completion of
processing.
Then re-run with '-r' option.
```

Correct the indicated errors and then re-run using the following command:

```
% ghicrcluster -r [<OS/HW_list>]
```

The ‘-r’ can also be ‘-vr’ to enable verbose output. When ‘-r’ is specified, ***ghicrcluster*** will attempt to complete configuration of the previously specified nodes. Repeat with ‘-r’ until “Done.” is displayed. <OS/HW_list> is optional, and defaults to whatever was specified in the original ***ghicrcluster*** command. If specified, it replaces the original spec per the discussion a couple paragraphs back.

Start GHI with the command:

```
% ghistartup -g
```

With GHI running, perform steps (2) and (3) for each filesystem and associated IOM to be configured. Step (2) may be performed for all filesystems and then step (3) for all IOMs, or steps (2) and (3) done in-turn for each filesystem/IOM(s).

For each filesystem, the FS name and mountpoint are to be supplied by the user. The ports to be used by the associated Event and Scheduler Daemons (ED and SD) may also be user-supplied, or most probably, left to their default values. Each FS is configured via the ***ghiaddfs*** command:

```
ghiaddfs [-v] <FS_Name> [-c "# <comment>"] <Mount_Point>
[<SD_Port> <ED_Port>]
```

<FS_Name> and <Mount_Point> must be the same as the GPFS configuration for the FS.

The ‘-v’ option is highly-reccomended in case any execution errors need to be resolved, for example, to configure file system “ghi_fs1”, mounted at “/ghi/fs1”:

```
% ghiaddfs -v ghi_fs1 /ghi/fs1
```

The default SD end ED ports are 80x0 for the SD and 80x1 for the ED, where ‘x’ is the order in which filesystems were configured, i.e., 8010 and 8011 for the first-configured FS, 8020 and 8021 for the second-configured FS, and so on. The actual configured port numbers will be the first available ports starting with the default. If more than 9 filesystems are to be configured, then the port numbers will need to be explicitly specified for filesystem number 10+ because port numbers are limited to 65535. Unless more than 9 filesystems are to be configured, letting ***ghiaddfs*** pick the ports is recommended.

Here is an example in which the ports to be used for the SD and ED are explicitly specified:

```
% ghiaddfs -v ghi_fs1 -c "# For highly-volatile data" /ghi/fs1
10101 10102
```

If all proceeds without error, “Done.” will be displayed. Otherwise, ***ghiaddfs*** will attempt to back-out any changes made before encountering the error.

Beginning with version 2.4, GHI introduced the capability to configure a GHI-managed file system to be read-only with respect to HPSS. The preceding discussion of ***ghiaddfs*** applied to creation of a normal, full-access file system. Please refer to the *GHI Management Guide* for a complete discussion of GHI read-only file systems and how and when they should be configured.

Once an FS (normal full-access or read-only) has been configured via ***ghiaddfs***, its associated IOM(s) can be configured. This is done for each IOM to be configured via command ***ghiaddiom*** as follows:

```
ghiaddiom [-v] <FS_Name> [-c "# <comment>"] <IOM_Node>[:<Port>] \
          <Active_On_Session_Node> <Est_XFER_Rate>
```

The ‘-v’ option is highly-reccomended in case any execution errors need to be resolved.

<FS_Name> is the name of the filesystem for which the IOM is being configured.

<IOM_Node> is the node on which the IOM will run, optionally on the associated “:<Port>”.

<Active_On_Session_Node> is either ‘true’ or ‘false’, and indicates whether or not the IOM is to be activated if <IOM_Node> also happens to be the current GHI session node.

<Est_XFER_Rate> is the expected throughput, in bytes/second, expressed as an integer, optionally followed by units of KB, MB, or GB.

The default port selected for an IOM is 80x2, where ‘x’ is the order in which filesystems were configured, i.e., 8012 for the first-configured FS, 8022 for the second-configured FS, and so on. The actual configured port numbers will be the first available ports starting with the default. If more than 9 filesystems are to be configured, then the port numbers will need to be explicitly specified for filesystem number 10+ because port numbers are limited to 65535.

For example, to configure an IOM for FS “ghi_fs1” to run on node “io4”:

```
% ghiaddiom -v ghi_fs1 -c "# Fast-E NIC" io1 true 65MB
```

In addition to updating GHI-specific configuration data, ***ghiaddiom*** will also update system configuration files “/etc/inittab”, “/etc/services”, and either “/etc/inetd.conf” (AIX) or “/etc/xinetd.d/” (Linux).

If an error occurs which causes any of these ***ghi**** commands to terminate prior to completing, it will attempt to back-out any changes applied prior to the error. (The back-out steps will also be shown in verbose mode.) Correct the condition causing the error and repeat the command until a "Done." is obtained.

2.3. Define GHI Environment Variables

While most, if not all, GHI environment variables can be used as defined by HPSS, they need to be reviewed to ensure that they are set correctly. Refer to *Section 5.6 - Define HPSS Environment Variables* of the *HPSS 7.4.2 Installation Guide* for information on updating the environment variables. The GHI environment variables each come pre-loaded with a default value which should be as noted below. Custom values are assigned in "/var/hpss/etc/env.conf". The following three GHI environment variables are required:

HPSS_GHI_PATH

Path to GHI non-user run-time directories. Must be set to ***/var/hpss/ghi***.

HPSS_GHI_CONF

GHI configuration filename. Must be set to ***ghi.conf***.

HPSS_GHI_FSCONF

GHI file system IOM config file. Must be set to ***ghi_%s.conf***. This is a template in which the actual file for any given managed FS is obtained by substituting the FS name for the '%s. File(s) exists in "$HPSS_GHI_PATH/etc". For example, given an FS name of 'gpfs1fs', the actual filename would be "/var/hpss/etc/ghi_gpfs1s.conf".

2.4. GHI Manual Page Setup

This section describes the procedure to set up manual pages for GHI utilities to work with the *man* command.

Perform the following steps on all nodes desired:

1. If the man pages are to be made available on a node on which the GHI source has not been installed, perform the following:

```
% mkdir -p /opt/hpss/src/ghi/man
% cd /opt/hpss/src/ghi/man
% scp <node>:`pwd`/* .       ### don't miss the dot!
```

   <node> is any of the nodes on which the GHI source code was installed. The *scp* command copies files from "there" to "here". (Any site-approved method beside *scp* may be used to place "/opt/hpss/src/ghi/man" on these other nodes.)

2. Create a symbolic link for the GHI manual catalog by issuing the following

commands:

```
% cd /opt/hpss/src/ghi
% ln -s man cat7
```

3. Edit the “/etc/environment” file so that users have the GHI manual pages in their MANPATH by adding the following line to the end of the file:

```
MANPATH=${MANPATH}:/opt/hpss/src/ghi
```

After this is done, users who subsequently login to the system are able to view GHI manual pages using the *man* command. For example, to view the ***ghi_ls*** manual page:

```
% man ghi_ls
```

2.5. ILM Policies for HSM

GHI makes use of GPFS *Information Lifecycle Management* (ILM) policies. A policy is a plain-text file that describes files and directories to either be included in or excluded from some processing. GHI provides template versions which you may use as a starting point to configure custom policies. These templates can all be found in the “/var/hpss/ghi/policy” directory.

2.5.1. migrate.policy

This file is an example of a GPFS ILM policy that you can use as a starting point to create custom migration policies. This file can be placed in any directory in the system. The policy should have separate rules for aggregates and non-aggregates. The script that gets invoked from the policy engine, ***ghi_migrate***, requires a “-a” option to process aggregates.

2.5.2. reset_incomplete_migration.policy and reset_incomplete_migration.ksh

These template files constitute an example of executing a GPFS ILM policy that you can use as a starting point to create custom policies to reset files for which migration was started but never completed. Such files will show as “[incompletely-migrated]” when listed with “ghi_ls -h”. They are “migrated enough” such that GPFS will not select them to be re-migrated, and the migration-reset process will result in their being set back to “un-migrated” so that GPFS will select them in the next applicable migration policy run. These files can be placed in any directory in the system.

2.5.3. recall.policy

This file is an example of a GPFS ILM policy that you can use as a starting point to create custom recall policies. This file can be placed in any directory in the system.

2.6. GHI Backup Configuration

The backup process requires the following three GPFS ILM policy files:
“backup_migration.policy”**,** “backup_metadata.policy” **,** and “backup_error.policy”.
They must be copied from the “/var/hpss/ghi/policy” directory to
“/var/hpss/ghi/policy/<*file system*>” and modified to be file system specific.

2.6.1. backup_migration.policy

This file is an example of a GPFS backup migration policy rules file that you can use as a starting point to create your custom policy. The migration policy migrates all the unmanaged files to be backed up into HPSS. The policy file should be updated to reflect the migration rules to be used for this file system. The policy should be able to select every file that has not been migrated to HPSS – and exclude any file which should not be migrated.

2.6.2. backup_metadata.policy

The policy is used by the GPFS SOBAR ‘mmimgbackup’ command to provide the GPFS file system namespace and file metadata to GHI to be placed to HPSS.

WARNING: **The contents of this policy should not be changed.**

2.6.3. backup_error.policy

The policy generates a list of files which did not make it into the backup.

WARNING: **The contents of this policy should not be changed.**

2.7. GHI Startup and Threshold Limits Configurations

File “threshold.policy” is a template GPFS ILM policy which must be copied from the “/var/hpss/ghi/policy” directory to “/var/hpss/ghi/policy/<*file system*>” and modified to be file system specific. The script that gets invoked from the policy engine, ***ghi_migrate***, requires a “-p” option to punch holes in the file system.

To apply a threshold policy to a file system, issue the following command:

```
% mmchpolicy <file system> <full path of threshold.policy>
```

2.8. DB2 Backup Configuration

This section provides some guidance for the administrator to identify the site’s backup process for the GHI database table.

At this point, a cron job must be setup to backup DB2. The script “db2_fullbackup.ksh” is used to backup the metadata. The script will:

- Make one full online backup for the hghi database
- Run as hpssdb

Add a *cron* job to the root’s “crontab” to run the “db2_fullbackup.ksh” script once a day at a predetermined time. For example below at 03:00 each day:

```
# crontab -e
```

Add the following line. This example assumes that backup is to be run at 3:00 AM each day. Modify the time interval as appropriate.

```
      * 3 * * * /hpss_src/scripts/db2_fullbackup.ksh >
/db2_backup/db2info/backup_$(date+%Y%m%d%H%M) 2>&1
```

The file “/db2_backup/db2info/backup_<*timestamp*>” contains the log of each backup execution. After the initial configuration, regular backups of the metadata will be performed automatically as dictated by the cron job that is described above.

The HPSS Administrator must examine the log file to determine that each backup was successful. If the backup is not successful, resolve the underlying problem then follow the site’s backup policy to rerun the DB2 backup.

The database backup images in “/db2_backup” and the transaction logs archived in the directory “/db2_logarchive1” then must be saved into a safe storage area.

Backup the initial GPFS cluster configuration:

```
% /opt/hpss/bin/ghi_backup_cluster
```

2.9. HSIGWD / GHI-HTAR

The following sections describe how to configure the GHI specifics for the aggregation interface. GHI-HTAR provides its own independent set of authentication mechanisms, which need not be the same as those used by GHI. GHI-HTAR authenticates to HPSS via the HSI Gateway Daemon, which runs as a trusted HPSS server.

GHI-HTAR must be installed on all GHI nodes on which an IOM may execute, and also on the HPSS Core Server node in conjunction with the installation and configuration of the HSI Gateway Daemon. Typical configuration is to install the HSI package in the “/var/hpss/hsi” directory, but it can be placed in any desired directory. This location will be refered to as “<install path>” in the discussions which follow.

2.9.1. HSIGWD

The HSIGWD is required to be configured on the HPSS Core Server. To configure the HSIGWD, perform the following steps:

1. Unpack the distribution package.
2. Run the Configure script.
3. Run the Compile script to build the desired components.
4. Execute the following command:

   ```
   % cp <install path>/5.0.0.1g/ndapi/ndserver/hpss_hsigwd.5.0.0.1g /opt/hpss/bin/hpss_hsigwd
   ```

5. Edit "/etc/services" and add at the end:

   ```
   % hsigwd 1217/tcp
   ```

6. On AIX platforms,
   - ***a.*** Edit "/etc/inetd.conf" and add at the end:

     ```
     hsigwd stream tcp nowait root /opt/hpss/bin/hpss_hsigwd hpss_hsigwd
     ```

   - ***b.*** Execute the command:

     ```
     % refresh -s inetd
     ```

7. On Linux platforms,
   - ***a.*** Execute the command:

     ```
     % cp <install path>/5.0.0.1g/misc/templates/xinetd.d /etc/xinetd.d/hsigwd.
     ```

   - ***b.*** Execute the command**,** where pid is the ***xinetd*** process ID:

     ```
     % kill –HUP pid
     ```

8. Append the HSI stanza in "<*install path*>/5.0.0.1g/misc/templates/HPSS.conf.template" to "/var/hpss/etc/HPSS.conf"**.**
9. Create the "/var/hpss/etc/cos" file by running "<*install path*>/5.0.0.1g/hsi/templates/make_cos.pl"**.** The COS configuration file generated is used to coordinate the storage location for the GHI-HTAR data files.
10. Create the "/var/hpss/ndapi" directory, used to contain the logfiles created by the HSIGWD.
11. For UNIX authentication:
    - ***a.*** Run "hpss_unix_keytab" to create the "hpss.htar.keytab" file to contain the hpssdmg ID.

b. Set permissions to 0x600 and owner to root.

c. Copy the keytab file to each of the GHI nodes.

12. For Kerberos authentication, exectute:

```
% cp
<install path>/5.0.0.1g/ndapi/snderver/hpss/hsigwd_kchild.5
.0.0.1g /opt/hpss/bin \
    /hsigwd_kchild
```

2.9.2. GHI-HTAR

GHI-HTAR executables must reside on each of the IOM nodes. To configure GHI-HTAR, perform the following steps:

1. Unpack the distribution package in "/var/hpss/hsi".

2. Run the "Configure" script. Be sure to set

NDAPI_SERVER_HOST=<*HPSS Core Server*>

and

HTAR_GPFS_HPSS_SUPPORT=on

All other options can be left to the suggested defaults, subject to site policy

3. If the "Configure" script does not initiate the compilation process, run the "Compile" script to build the desired components. The suggested default settings should be used.

The display of "info" or "warning" messages during execution of the compilation process is acceptable and should not be an indication that the compiled code will not execute properly. Any "error" messages must be addressed.

4. Execute:

```
% cp <install path>/5.0.0.1g/htar/src/htar
/var/hpss/hsi/bin/htar
```

5.

```
% cp <install path>/hsi/5.0.0.1g/htar/wrapper/htar.aix.blue
/var/hpss/hsi/bin/htar.ksh
```

The htar.ksh wrapper script must reside in "/opt/hpss/bin". The script must be modified to provide correct values for the following information:

HTAR_BASE_PATH: Location of HSI executable. This should be set to the location of the GHI executables (typically "/opt/hpss/bin").

TMPDIR: Location of the temporary files. The amount of space required is based on the size of an aggregate, plus temporary files created for the data files. It is recommended to set this variable to "<*mount_point*>/scratch/.ghi/<hostname>".

REALM_NAME**:** Realm name for location of HSIGWD. Set this variable when using Kerberos authentication.

HPSS_KEYTAB_PATH**:** Location of keytab. Set this variable when using Unix authentication.

HPSS_HOSTNAME**:** Interface to be used for the control path.

6. Copy "/var/hpss/hsi/bin/htar" and "/opt/hpss/hsi/bin/htar.ksh" file to each of the GHI I/O Manager nodes. Edit ***HPSS_HOSTNAME*** on each IOM node to reflect the control path for that node

2.10. **GHI Backups**

GHI backups of the file system require that the policy files be placed in the "/var/hpss/ghi/policy/<*file_system*>" directory. The template files should also be placed in that directory. The policy and template files should be modifed based on site specific requirements.

APPENDIX A - GLOSSARY OF TERMS AND ACRONYMS

Each of the following may have separate meanings in HPSS and GHI, and will be so noted if so. If not noted, the same meaning applies to both.

ACL HPSS - Access Control List.

AIX Advanced Interactive Executive. An operating system provided on many IBM machines.

Alarm HPSS - A log record message type used to log high-level error conditions.

ANSI American National Standards Institute.

API Application Program Interface.

Archive HPSS - One or more interconnected storage systems of the same architecture.

Attribute When referring to a managed object, an attribute is one discrete piece of information, or set of related information, within that object.

Class of Service HPSS - A set of storage system characteristics used to group files with similar logical characteristics and performance requirements together. A Class of Service is supported by an underlying hierarchy of storage classes.

CM GHI - Configuration Manager.

co-managed GHI - File data resides in both GPFS and HPSS.

Configuration The process of initializing or modifying various parameters affecting the behavior of a server or infrastructure service.

COS HPSS - Class of Service.

Core Server HPSS - An HPSS server which manages the namespace and storage for an HPSS system. The Core Server manages the Name Space in which files are defined, the attributes of the files, and the storage media on which the files are stored. The Core Server is the central server of an HPSS system. Each storage sub-system uses exactly one Core Server.

Daemon A UNIX program that runs continuously in the background.

DB2 A relational database system, a product of IBM Corporation, used by HPSS and GHI to store and manage HPSS and GHI metadata.

Debug HPSS - A log record message type used to log lower-level error conditions.

GHI - A log record message type used to log lower-level processing steps/conditions.

Delog HPSS - The process of extraction, formatting, and outputting HPSS central log records.

Directory An object that can contain files, symbolic links, hard links, and other directories.

Dismount HPSS - An operation in which a cartridge is either physically or logically removed from a device, rendering it unreadable and unwritable. In the case of tape cartridges, a dismount operation is a physical operation. In the case of a fixed disk unit, a dismount is a logical operation.

DMAPI GHI - Data Management Application Programming Interface.

DNS Domain Name Service.

DOE Department of Energy.

Drive HPSS - A physical piece of hardware capable of reading and/or writing mounted cartridges. The terms device and drive are often used interchangeably.

DRP Disaster/Recovery Plan

ED GHI - Event Daemon.

Event GHI - A log record message type used to log informational messages (e.g., subsystem starting, subsystem terminating).

Export An operation in which a cartridge and its associated storage space are removed from the HPSS system Physical Volume Library. It may or may not include an eject, which is the removal of the cartridge from its Physical Volume Repository.

File An object than can be written to, read from, or both, with attributes including access permissions and type, as defined by POSIX (P1003.1-1990). HPSS supports only regular files.

file family HPSS - An attribute of an HPSS file that is used to group a set of files on a common set of tape virtual volumes.

fileset A collection of related files that are organized into a single easily managed unit. A fileset is a disjoint directory tree that can be mounted in some other directory tree to make it accessible to users.

fileset ID A 64-bit number that uniquely identifies a GPS fileset.

fileset name A name that uniquely identifies a fileset.

file system ID A 64-bit number that uniquely identifies a GPS file system.

GB Gigabyte (2^{30}).

GPFS General Parallel File System.

GHI GPFS/HPSS Interface.

GHI-HTAR Specially modified GHI-specific version of the HTAR program.

GSS Generic Security Service.

Hierarchy HPSS - See Storage Hierarchy.

HPSS High Performance Storage System.

HSI Hierarchical Storage Interface.

HSIGWD HSI Gateway Daemon.

HTAR HPSS tar program – a utility to aggregate a set of files directly into HPSS without first writing to local storage, and to randomly retrieve individual member files via creation of a separate index file.

IBM International Business Machines Corporation.

ID Identifier.

I/O	Input/Output.
IOA	GHI - I/O Agent.
IOM	GHI - I/O Manager.
IP	Internet Protocol.
junction	HPSS - A mount point for an HPSS fileset.
KB	Kilobyte (210).
LAN	Local Area Network.
LANL	Los Alamos National Laboratory.
LD	GHI - Log Daemon.
LLNL	Lawrence Livermore National Laboratory.
MB	Megabyte.
MD	GHI - Mount Daemon.
metadata	Control information about the data stored under HPSS, such as location, access times, permissions, and storage policies. Most HPSS metafile contents are stored in a DB2 relational database.
migrate	HPSS - To copy file data from a level in the file's hierarchy onto the next lower level in the hierarchy. GHI - To copy file data from a GPFS file into HPSS.
mount	HPSS - An operation in which a cartridge is either physically or logically made readable and/or writable on a drive. In the case of tape cartridges, a mount operation is a physical operation. In the case of a fixed disk unit, a mount is a logical operation.
mount point	HPSS - A place where a fileset is mounted in the XFS and/or HPSS namespaces. GHI - A place where a filesystem or fileset is mounted in the GPFS namespaces.

Mover HPSS - An HPSS server that provides control of storage devices and data transfers within HPSS.

Name Service HPSS - The portion of the Core Server that providesa mapping between names and machine oriented identifiers. In addition, the Name Service performs access verification and provides the Portable Operating System Interface (POSIX).

name space HPSS - The set of name-object pairs managed by the HPSS Core Server.

NLS National Language Support.

NSL National Storage Laboratory.

Object See Managed Object.

OSF Open Software Foundation.

PB Petabyte (2^{50}).

PM GHI - Process Manager.

POSIX Portable Operating System Interface (for computer environments).

RPC Remote Procedure Call.

SCSI Small Computer Systems Interface.

SD GHI - Scheduler Daemon.

SNL Sandia National Laboratories.

SSA Serial Storage Architecture.

storage class An HPSS object used to group storage media together to provide storage for HPSS data with specific characteristics. The characteristics are both physical and logical.

storage hierarchy An ordered collection of storage classes. The hierarchy consists of a fixed number of storage levels numbered from level 1 to the number of levels in the hierarchy, with the maximum level being limited to 5 by HPSS. Each level is associated with a specific storage class. Migration and stage commands result in data being copied between different storage levels in the hierarchy. Each Class of Service has an associated hierarchy.

storage subsystem A portion of the HPSS namespace that is managed by an independent Core Server and (optionally) Migration/Purge Server.

TB Terabyte (2^{40}).

TCP/IP Transmission Control Protocol/Internet Protocol.

Transaction A programming construct that enables multiple data operations to possess the following properties:
All operations commit or abort/roll-back together such that they form a single unit of work.
All data modified as part of the same transaction are guaranteed to maintain a consistent state whether the transaction is aborted or committed.
Data modified from one transaction are isolated from other transactions until the transaction is either committed or aborted.
Once the transaction commits, all changes to data are guaranteed to be permanent.

APPENDIX B - REFERENCES

- ***HPSS Installation Guide***
- ***HPSS Management Guide***
- ***HPSS User’s Guide***
- ***HPSS Conversion Guide***
- ***GPFS Data Management API Guide***
- ***GPFS Administration and Programming Reference***
- ***GPFS Advanced Administration***

APPENDIX C - DEVELOPER ACKNOWLEDGMENTS

The GHI feature of HPSS was developed by IBM Global Business Services – Federal. HPSS is jointly owned and developed by the HPSS Collaboration consisting of Lawrence Berkeley National Laboratory, Lawrence Livermore National Laboratory, Los Alamos National Laboratory, Oak Ridge National Laboratory, Sandia National Laboratories, and IBM.

We would like to acknowledge IBM Almaden Research Center for the software development collaboration between GPFS and HPSS to implement the GPFS/HPSS Interface.

We would like to acknowledge NERSC, the National Energy Research Scientific Computing Center, for their help with initial design and development.

We would like to acknowledge SDSC, San Diego Supercomputing Center, for their help with the initial requirement review.

We would also like to acknowledge HLRS, High Performance Computing Center of the University of Stuttgart, and NCSA, National Center for Computing Applications, for providing testbeds for the initial GHI release.