GUFI User's Guide

GUFI Developers

September 11, 2023

Contents

1 License

This file is part of GUFI, which is part of MarFS, which is released
under the BSD license.

Redistribution and use in source and binary forms, with or without modification,
are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this
list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice,
this list of conditions and the following disclaimer in the documentation and/or
other materials provided with the distribution.

3. Neither the name of the copyright holder nor the names of its contributors
may be used to endorse or promote products derived from this software without
specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY EXPRESS 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 THE COPYRIGHT HOLDER OR 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.

From Los Alamos National Security, LLC:
LA-CC-15-039

2 Introduction

Over the years, the amount of data we store and use has grown exponentially to the point that petabytes of storage is not uncommon. What used to be a simple task of accessing and sorting through information has been compounded into an arduous task with the size and scale of super-computing data centers. Being able to query data effectively, while also taking into account user permissions becomes paramount into accomplishing daily tasks. This is what the Grand Unified File Index (GUFI) tool aims to accomplish.

This process of efficiently accessing data is accomplished by recreating the tree structure via indexing. Each directory contains an SQL database file that stores the metadata of the files as well as summary information for that directory and optionally summary information for the entire tree below that directory.

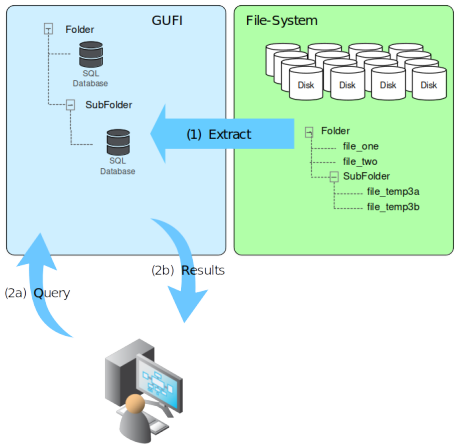

Figure 1: Layout and user interaction with GUFI

3 Environment

Users interact with GUFI using `gufi_find`, `gufi_ls`, `gufi_stat`, and `gufi_stats`. These are wrapper scripts that access the actual implementations of these files.

There should be a file readable (but not necessarily modifiable) by all GUFI users called `/etc/GUFI/config`. This file specifies where the GUFI server is, as the actual GUFI trees are not expected to be locally available.

4 gufi_find

`gufi_find` is a wrapper script for `gufi_query` that attempts to recreate a large portion of GNU `find(1)`'s functionality.

One major difference between them is how arguments are parsed. In `find(1)`, expression order matters. In `gufi_find`, expression order does not matter.

4.1 Flags

The following `find(1)` test expression options are replicated. See `find(1)` for details.

> `-amin`, `-atime`, `-cmin`, `-ctime`, `-empty`, `-executable`, `-false`, `-fprint`, `-gid`, `-group`, `-help`, `-iname`, `-inum`, `-links`, `-lname`, `-maxdepth`, `-mindepth`, `-mmin`, `-mtime`, `-name`, `-newer`, `-path`, `-printf`, `-readable`, `-samefile`, `-size`, `-true`, `-type`, `-uid`, `-user`, and `-writable`

The action expressions `printf` and `fprint` are also available. `printf` writes to `stdout`, and `fprint` writes to per-thread output text files. All format specifiers other than `ACFTYZ` have been implemented.

Additionally, a few GUFI extensions have been added:

Option	Description

--size% num num	Extends the `size` option to search for sizes in the range [size + min%, size + max%].
	e.g.
	`-size 100c --size% 10 20` → `[110, 120]`
	`-size 100c --size% -10 20` → `[90, 120]`

--num-results num	Limit the number of results printed
--smallest	Output by size, ascending.
--largest	Output by size, descending.
--in-memory-name name	Change the name of the tables used to store intermediate results when aggregating. Generally not used.

5 gufi_ls

`gufi_ls` is the equivalent of the POSIX command ls applied to a GUFI tree. As with `gufi_find`, there are a multitude of options available listed below

5.1 Flags

Flags	Functionality
--help	displays help menu
-v, --version	show program's version number and exit
-a, --all	do not ignore entries ending with .
-A, --almost-all	do not list implied . and ..
--block-size <block_size>	with -l, scale sizes by block_size when printing them
-B, --ignore-backups	do not list implied entries ending with
-G, --no-group	in a long listing, don't print group names
-i, --inode	print the index number of each file
-l	used a long listing format
-r, --reverse	reverse order while sorting
-R, --recursive	list sub-directories recursively
-s, --size	print the allocated size of each file in blocks
-S	sort by file size, largest first
--time-style <TIME_STYLE>	time/date format with -l
-t	sort by modification time, newest first
-U	do not sort; list entries in directory order

Table 1: gufi_ls Flags and Functionality

Additionally, several GUFI extensions have been added:

Option	Description
--delim <c	delimiter separating output columns
--in-memory-name <name>	Name of in-memory database when -R is used
--aggregate-name <name>	Name of final database when -R is used
--nlink-width <chars>	Width of nlink column
--size-width <chars>	Width of size column
--user-width <chars>	Width of user column
--group-width <chars>	Width of group column
--skip-file <filename>	Name of file containing directory basenames to skip

6 gufi_stat

gufi_stat is analogous to stat(1) in file status mode.

6.1 Flags

Option	Description
-c FORMAT, --format FORMAT	use the specified FORMAT instead of the default; output a newline after each use of FORMAT
-t, --terse	print the information in terse form
--help	display this help and exit
--version	output version information and exit

7 gufi_stats

`gufi_stats` is used to analyze a tree and retrieve statistics from an index. `gufi_stat` does not have an analogous standard utility.

7.1 Flags

`gufi_stats` is called with at minimum one positional argument specifying the statistic desired. Each statistic has a default version, Most also have a more in-depth version that is specified by their category: recursive, cumulative, or other.

Optional Flags	Functionality
--help	displays help menu
--version, -v	display program's version number and exits
--recursive, -r	run command recursively
--cumulative, -c	return cumulative values
--order <order>	sort output (if applicable)
--delim <c>	delimeter separating output columns
--num-results <n>	first n results
--uid <u>, --user <u>	restrict to user
--in-memory-name <name>	Name of intermediate database
--aggregate-name <name>	Name of final database
--skip-file <filename>	Name of file containing directory basenames to skip

Table 2: `gufi_stats` Flags and Functionality

7.1.1 Recursive

These statistics will run a computation on a single directory. Specifying `-r` or `--recursive` will recursively descend the starting directory and return each subdirectory's statistic.

Statistic	Description
depth	Get the depths of the provided directory relative to the root directory
filesize	Get the size the files in the immediate directory
filecount	Get the number of files in the immediate directory
linkcount	Get the number of links in the immediate directory
dircount	Get the number of directories in the immediate directory
leaf-dirs	Get the leaf directories immediately under the current directory
leaf-depth	Get the depth of the leaf directories immediately under the current directory
leaf-files	Get number of files in the leaf directories immediately under the current directory
leaf-links	Get number of links in the leaf directories immediately under the current directory

7.1.2 Cumulative

These statistics will run a computation on an entire subtree. By default, the results will be grouped together by UID. Specifying `-c` or `--cumulative` will combine all of the results into a single sum.

Statistic	Description
total-filesize	Get the total size taken up by the entire directory
total-filecount	Get the total number of files under the directory
total-linkcount	Get the total number of links under the directory
total-dircount	Get the total number of directories under the directory
total-leaf-files	Get total number of files in the leaf directories under this directory
total-leaf-links	Get total number of links in the leaf directories under this directory
files-per-level	Get counts of how many files are in each level of the tree
links-per-level	Get counts of how many links are in each level of the tree
dirs-per-level	Get counts of how many dirs are in each level of the tree
average-leaf-files	Get average number of leaf files under the provided directory
average-leaf-links	Get average number of leaf links under the provided directory

7.1.3 Other

Statistics under the "other" category do not have extra flags associated with them.

Statistic	Description
median-leaf-files	Get median number of leaf files under the provided directory
duplicate-names	Find files with matching names and sizes