GlusterFS 1.3 User Guide

August 15, 2007

Vikas Gorur

This is the user manual for GlusterFS 1.3.

Table of Contents

Acknowledgements

GlusterFS continues to be a wonderful and enriching experience for all of us involved. I'd like to thank Anand Babu, who concieved GlusterFS and has lead us throughout with his infectious enthusiasm. The development team consists of Anand Avati, Amar Tumballi, Basavanagowda Kanur, Krishna Srinivas, Raghavendra G, and myself. Our CEO, Hitesh Chellani, ensures we get paid for hacking on GlusterFS.

GlusterFS development would not have been possible at this pace if not for our enthusiastic users. People from around the world have helped us with bug reports, performance numbers, and feature suggestions. A huge thanks to them all.

Matthew Paine - for RPMs & general enthu

Leonardo Rodrigues de Mello - for DEBs

Julian Perez & Adam D'Auria - for multi-server tutorial

Paul England - for HA spec

Brent Nelson - for many bug reports

Vikas Gorur (vikas@zresearch.com)
Z Research

1 Introduction

GlusterFS is a distributed filesystem. It works at the file level, not block level.

A network filesystem is one which allows us to access remote files. A distributed filesystem is one that stores data on multiple machines and makes them all appear to be a part of the same filesystem.

Need for distributed filesystems

- Scalability: A distributed filesystem allows us to store more data than what can be stored on a single machine.
- Redundancy: We might want to replicate crucial data on to several machines.
- Uniform access: One can mount a remote volume (for example your home directory) from any machine and access the same data.

1.1 Contacting us

You can reach us through the mailing list **gluster-devel** (gluster-devel@nongnu.org).

You can also find many of the developers on IRC, on the `#gluster` channel on Freenode (`irc.freenode.net`).

For commercial support, you can contact Z Research at:

Z Research Inc.,
3194 Winding Vista Common
Fremont, CA 94539
USA.
Phone: +1-510-5346801
Toll free: +18888136309

You can also email us at support@zresearch.com.

2 Installation and Invocation

2.1 Pre requisites

Before installing GlusterFS make sure you have the following components installed.

2.1.1 FUSE

You'll need FUSE version 2.6.0 or higher to use GlusterFS. You can omit installing FUSE if you want to build *only* the server. Note that you won't be able to mount a GlusterFS filesystem on a machine that does not have FUSE installed.

FUSE can be downloaded from: `http://fuse.sourceforge.net/`

2.1.2 libibverbs (optional)

This is only needed if you want GlusterFS to use InfiniBand as the interconnect mechanism between server and client. You can get it from:

`http://www.openfabrics.org/downloads.htm`.

2.1.3 Bison and Flex

These should be already installed on most Linux systems. We recommend using GNU Bison and Flex.

2.2 Getting GlusterFS

There are many ways to get hold of GlusterFS. For a production deployment, the recommended method is to download the latest release tarball. Release tarballs are available at: `http://gluster.org/download.php`.

If you want the bleeding edge development source, you can get them from the GNU Arch[1] repository. First you must install GNU Arch itself. Then register the GlusterFS archive by doing:

```
$ tla register-archive http://arch.sv.gnu.org/archives/gluster
```

Now you can check out the source itself:

```
$ tla get -A gluster@sv.gnu.org glusterfs--mainline--2.5
```

If you are on an RPM based system, you can also try RPMs contributed by Matthew Paine (`matt@mattsoftware.com`), for CentOS 5, available at:

`http://www.mattsoftware.com/msw_repo/centos/5/`

Leonardo Rodrigues de Mello (`l@lmello.eu.org`) has created Ubuntu (Etch) packages of GlusterFS. They are available at:

`http://guialivre.governoeletronico.gov.br/guiaonline/downloads/pacotes-cluster/dists/etch/glusterfs/`

2.3 Building

You can skip this section if you're installing from RPMs or DEBs.

GlusterFS uses the Autotools mechanism to build. As such, the procedure is straight-forward. First, change into the GlusterFS source directory.

[1] `http://www.gnu.org/software/gnu-arch/`

```
$ cd glusterfs--1.3
```

If you checked out the source from the Arch repository, you'll need to run `./autogen.sh` first. Note that you'll need to have Autoconf and Automake installed for this.

Run `configure`.

```
$ ./configure
```

The configure script accepts the following options:

`--disable-ibverbs`
: Disable the InfiniBand transport mechanism.

`--disable-fuse-client`
: Disable the FUSE client.

`--disable-server`
: Disable building of the GlusterFS server.

2.4 Running GlusterFS

2.4.1 Server

2.4.2 Client

2.5 A Tutorial Introduction

3 Concepts

3.1 Filesystems in Userspace

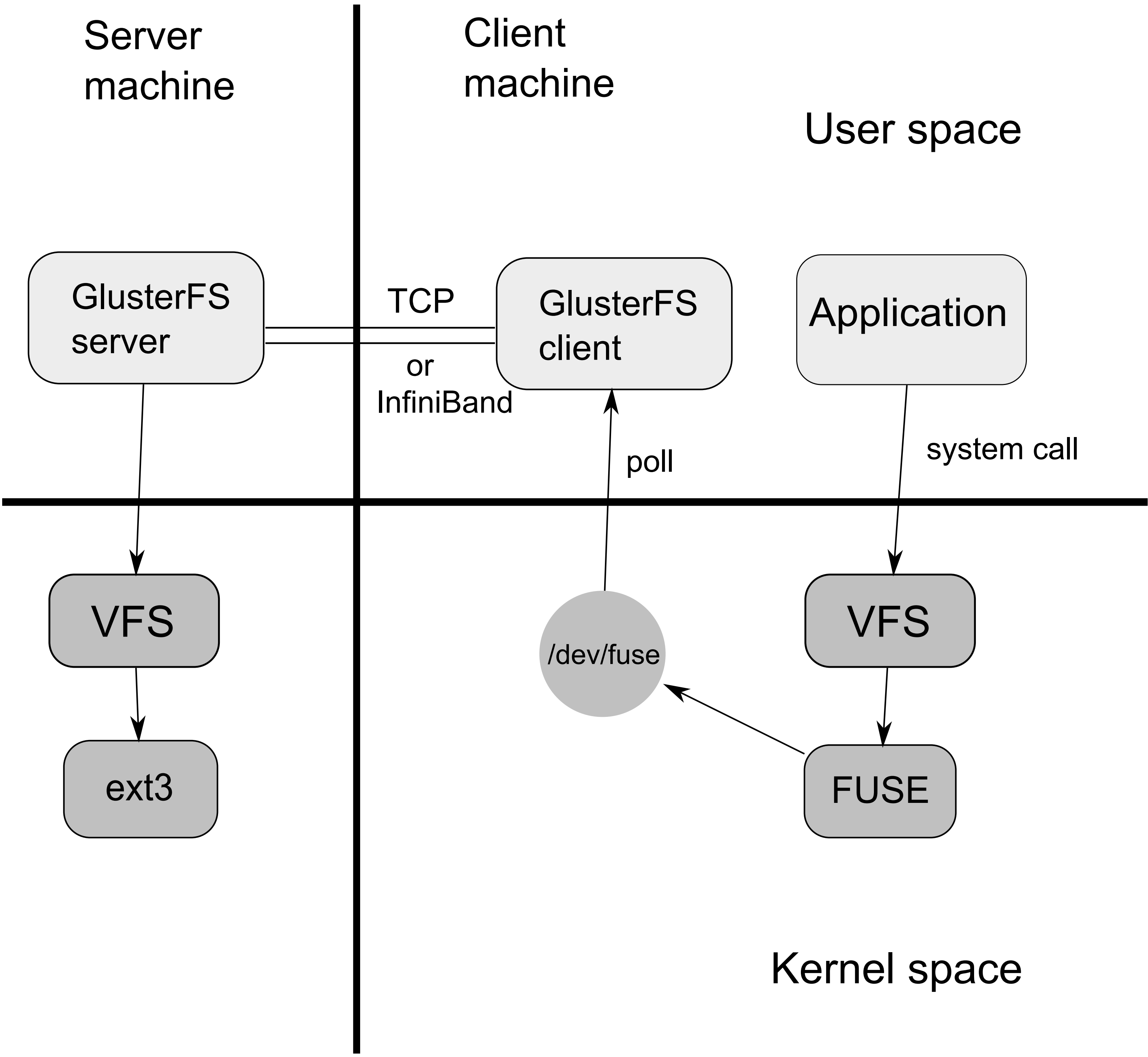

Fig 1. Control flow in GlusterFS

A filesystem is usually implemented in the kernel. Kernel development is much harder than userspace development. FUSE is a kernel module/library that allows us to write a filesystem completely in userspace.

FUSE consists of a kernel module which interacts with the userspace implementation using a device file `/dev/fuse`. When a process makes a syscall on a FUSE filesystem, VFS hands the request to the FUSE module, which writes the request to `/dev/fuse`. The userspace implementation polls `/dev/fuse`, and when a request arrives, processes it and writes the result back to `/dev/fuse`. The kernel then reads from the device file and returns the result to the user process.

application -> kernel -> fuse -> /dev/fuse -> user process -> server -> underlying filesystem

3.2 Translator

3.3 Volume specification file

4 Translators

4.1 Storage Translators

Amazon S3 support is planned.

4.1.1 POSIX

```
type storage/posix
```

reuses POSIX compatible underlying filesystem.

directory <path>
: j

inode-lru-limit <n> (1000)
: k

4.2 Client and Server Translators

4.2.1 Transport modules

three types of transports

non-blocking-connect [no|off|on|yes] (on)
remote-port <n> (6996)
remote-host <hostname> *

infiniband h/w gives api called verbs. lowest level of s/w access. (= ib-verbs). highest performance. ib-verbs reliable connection-oriented channel transfer. (Mellanox notes).

options:

ib-verbs-work-request-send-count <n> (64)
: foo

ib-verbs-work-request-recv-count <n> (64)
: asf

ib-verbs-work-request-send-size <size> (128KB)
: asdf

ib-verbs-work-request-recv-size <size> (128KB)
: adsf

ib-verbs-port <n> (1)
: iuio

ib-verbs-mtu [256|512|1024|2048|4096] (2048)
ib-verbs-device-name <device-name> (first device in the list)
: oaisdf

"impedance matching" is necessary.

ib-sdp. kernel implements socket interface for ib hardware. SDP is over ib-verbs.

ib-verbs is preferred over ib-sdp.

4.2.2 Client

```
type procotol/client
```

client protocol.

```
transport-type [tcp,ib-sdp,ib-verbs] (tcp/client)
remote-subvolume <volume_name> *
inode-lru-limit <n> (1000)
transport-timeout <n> (120- seconds)
```

4.2.3 Server

```
type protocol/server
```

```
client-volume-filename <path> (<CONFDIR>/glusterfs-client.vol)
transport-type [tcp,ib-verbs,ib-sdp] (tcp/server)
```

4.3 Clustering Translators

4.3.1 Unify

```
type cluster/unify
```

unify unifies its subvolumes. it has children, and will do stuff on them.

scheduler is used for creates. rr, random nufa - prefers local. otherwise does rr alu - adaptive least usage. Various criteria. order of preference. entry & exit threshold.

4.3.1.1 ALU

ALU stands for "Adaptive Least Usage". It is the most advanced scheduler available in GlusterFS. It balances the load across volumes, taking several factors in account. It adapts itself to changing I/O patterns, according to its configuration. When properly configured, it can eliminate the need for regular tuning of the filesystem to keep volume load nicely balanced.

The ALU scheduler is composed of multiple least-usage sub-schedulers. Each sub-scheduler keeps track of a certain type of load, for each of the subvolumes, getting the actual statistics from the subvolumes themselves. The sub-schedulers are these:

disk-usage - the used and free disk space on the volume

read-usage - the amount of reading done from this volume

write-usage - the amount of writing done to this volume

open-files-usage - the number of files currently opened from this volume

disk-speed-usage - the speed at which the disks are spinning. This is a constant value and therefore not very useful.

The ALU scheduler needs to know which of these sub-schedulers to use, and in which order to evaluate them. This is done through the "option alu.order" configuration directive.

Each sub-scheduler needs to know two things: when to kick in (the entry-threshold), and how long to stay in control (the exit-threshold). For example: when unifying three disks of 100GB, keeping an exact balance of disk-usage is not necesary. Instead, there could be a 1GB margin, which can be used to nicely balance other factors, such as read-usage. The disk-usage scheduler can be told to kick in only when a certain threshold of discrepancy is passed, such as 1GB. When it assumes control under this condition, it will write all subsequent data to the least-used volume. If it is doing so, it is unwise to stop right after the values are below the entry-threshold again, since that would make it very likely that the situation will occur again very soon. Such a situation would cause the ALU to spend most of its time disk-usage scheduling, which is unfair to the other sub-schedulers. The exit-threshold therefore defines the amount of data that needs to be written to the least-used disk, before control is relinquished again.

In addition to the sub-schedulers, the ALU scheduler also has "limits" options. These can stop the creation of new files on a volume once values drop below a certain threshold. For example, setting "option alu.limits.min-free-disk 5GB" will stop the scheduling of files to volumes that have less than 5GB of free disk space, leaving the files on that disk some room to grow.

The actual values you assign to the thresholds for sub-schedulers and limits depend on your situation. If you have fast-growing files, you'll want to stop file-creation on a disk much earlier than when hardly any of your files are growing. If you care less about disk-usage balance than about read-usage balance, you'll want a bigger disk-usage scheduler entry-threshold and a smaller read-usage scheduler entry-threshold.

For thresholds defining a size, values specifying "KB", "MB" and "GB" are allowed. For example: "option alu.limits.min-free-disk 5GB".

```
alu.order <order> *
("disk-usage:write-usage:read-usage:open-files-usage:disk-speed")
alu.disk-usage.entry-threshold <size> (1GB)
alu.disk-usage.exit-threshold <size> (512MB)
alu.write-usage.entry-threshold <%> (25)
alu.write-usage.exit-threshold <%> (5)
alu.read-usage.entry-threshold <%> (25)
alu.read-usage.exit-threshold <%> (5)
alu.open-files-usage.entry-threshold <n> (1000)
alu.open-files-usage.exit-threshold <n> (100)
alu.limits.min-free-disk <%>
alu.limits.max-open-files <n>
```

4.3.1.2 Round Robin (RR)

Round-Robin (RR) scheduler creates files in a round-robin fashion. Each client will have its own round-robin loop. When your files are mostly similar in size and I/O access pattern, this scheduler is a good choice. RR scheduler now checks for free disk size of the server before scheduling, so you can get to know when to add another server brick. The default value of min-free-disk is 5% and is checked every 10seconds (by default) if there is any create call happening.

```
rr.limits.min-free-disk <%> (5)
rr.refresh-interval <t> (10 seconds)
```

```
random.limits.min-free-disk <%> (5)
random.refresh-interval <t> (10 seconds)
```

4.3.1.3 NUFA

Non-Uniform Filesystem Scheduler similar to NUMA (http://en.wikipedia.org/wiki/Non-Uniform_Memory_Access) memory design. It is mainly used in HPC environments where you are required to run the filesystem server and client within the same cluster. Under such environment, NUFA scheduler gives the local system more priority for file creation over other nodes.

```
nufa.limits.min-free-disk <%> (5)
nufa.refresh-interval <t> (10 seconds)
nufa.local-volume-name <volume>
```

Namespace volume needed because: - persistent inode numbers. - file exists even when node is down. namespace files are simply touched. on every lookup it is checked.

Self heal: two rules: - dir structure should be consistent. - file should exist on only one node.

```
namespace <volume> *
self-heal [on|off] (on)
inode-lru-limit <n> (1000)
```

4.3.2 Automatic File Replication (AFR)

```
type cluster/afr
```

Replication is via *pattern:n*. Extended attributes needed for self heal functionality. Version number and ctime is stored in the attributes.

All of this not recommended: If you increase n, new file will be created. If you decrease n, nothing happens.

If you change subvolume order, it asserts that the first n *available* (nodes which are up) subvolumes have the file.

If a file is missing on a node, the latest version available will be written there. Missing directories are created during lookup.

self heal happens on open.

subvolume list must be same on all clients. Recommended configuration is to have exact same spec. Use -s.

```
debug [on|off] (off)
self-heal [on|off] (on)
replicate <pattern> (*:1)
lock-node <child_volume> (first_child)
inode-lru-limit <n> (1000)
```

4.3.3 Stripe

```
type cluster/stripe
```

uses extended attrs to store info.

```
inode-lru-limit <n> (1000)
block-size <pattern> (*:0 no striping)
```

4.4 Performance Translators

4.4.1 Read Ahead

```
type performance/read-ahead
```

read-ahead pre-fetches a sequence of blocks in advance based on its predictions. When your application is busy crunching the data it has read, glusterfs can pre-read the next batch of data in advance and keep it ready. That way consecutive reads are faster. Additionally it also behaves as a read-aggregator, i.e smaller I/O read operations are combined into fewer larger read operations internally to reduce network and disk load. page-size describes the block size and page-count describes amount of blocks to pre-fetch.

This translator is well utilized when used with IB-verbs transport. With FastEthernet and GigE interface, without read-ahead, one can achieve link max.

all reads are broken into page counts. page+n are read. if the read is not consecutive read ahead is stopped. page-count is per file.

```
page-size <n> (256KB)
page-count <n> (2)
force-atime-update [on|off|yes|no] (off|no)
```

4.4.2 Write Behind

```
type performance/write-behind
```

In general write operations are slower than read. The write-behind translator improves write performance significantly over read by using "aggregated background write" technique. That is, multiple smaller write operations are aggregated into fewer larger write operations and written in background (non-blocking). aggregate-size determines the block size till which write data should be aggregated. Depending upon your interconnect, RAM size and work load profile you should tune this value. Default of 128KB works well for most users. Increasing or decreasing this value beyond certain range will bring down your performance. You should always benchmark with an increasing range of aggregate-size and analyze the results to choose an optimum value.

flush behind

```
aggregate-size <n> (0)
flush-behind [on|yes|off|no] (off|no)
```

4.4.3 IO Threads

```
type performance/io-threads
```

AIO add asynchronous (background) read and write functionality. By loading this translator, you can utilize the server idle blocked time to handle new incoming requests. CPU, memory or network is not utilized when the server is blocked on read or write call while DMA'ing disk. This translator makes best use of all the resources under load and improves concurrent I/O performance.

NOTE: io-threads translator is useful when used over unify, or just below server protocol in server side. Its not used at all if used between unify and namespace brick as there is no FileI/O over namespace brick.

cache size = maximum that can be pending inside a thread.

```
thread-count <n> (1)
cache-size <n> (64MB)
```

4.4.4 IO Cache

```
type performance/io-cache
```

IO-Cache translator helps one to reduce to load on server (if loaded on client side) if client is accessing some files just for reading (and the file is not edited in server actually between two reads). For example, the header files are accessed for compilation of kernel.

```
page-size <n> (128KB)
cache-size (n) (32MB)
force-revalidate-timeout <n> (1)
priority <pattern> (*:0)
```

4.5 Features Translators

4.5.1 POSIX Locks

```
type features/posix-locks
```

This translator provides storage independent POSIX record locking support (fcntl locking). Typically you'll want to load this on the server side, just above the POSIX storage translator. Using this translator you can get both advisory locking and mandatory locking support. flock not supported.

Caveat: Consider a file that does not have its mandatory locking bits (+setgid, -group execution) turned on. Assume that this file is now opened by a process on a client that has the write-behind xlator loaded. The write-behind xlator does not cache anything for files which have mandatory locking enabled, to avoid incoherence. Let's say that mandatory locking is now enabled on this file through another client. The former client will not know about this change, and write-behind may erroneously report a write as being successful when in fact it would fail due to the region it is writing to being locked.

There seems to be no easy way to fix this. To work around this problem, it is recommended that you never enable the mandatory bits on a file while it is open.

`mandatory [on|off] (on)`

Turns mandatory locking on.

4.5.2 Fixed ID

```
type features/fixed-id
```

`fixed-uid <n> [if not set, not used]`
`fixed-gid <n> [if not set, not used]`

4.6 Miscallaneous Translators

4.6.1 ROT-13

```
type encryption/rot-13
```

v.simple translator

`encrypt-write [on|off] (on)`
`decrypt-read [on|off] (on)`

4.6.2 Trace

```
type debug/trace
```

debugging

5 Usage scenarios

- usage as network filesystem - clustering with four bricks (Julian Perez example, multi-server config example)

- HA setup (from HA tutorial by Paul England)

- encrypted glusterfs setup using ssh tunnels.

- Vserver guest (actually need a better organization for both tunneled setup and vserver thing)

6 Performance

- effect of direct_io mode.

7 Troubleshooting

GlusterFS log files.

Reporting a bug: –from howto report bug doc

Appendix A GNU Free Documentation Licence

0. PREAMBLE

   The purpose of this License is to make a manual, textbook, or other functional and useful document *free* in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondarily, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

   This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

   We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

1. APPLICABILITY AND DEFINITIONS

   This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you". You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

   A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

   A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

   The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

   The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

A section "Entitled XYZ" means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as "Acknowledgements", "Dedications", "Endorsements", or "History".) To "Preserve the Title" of such a section when you modify the Document means that it remains a section "Entitled XYZ" according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

2. VERBATIM COPYING

   You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

   You may also lend copies, under the same conditions stated above, and you may publicly display copies.

3. COPYING IN QUANTITY

   If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible.

You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.

B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.

C. State on the Title page the name of the publisher of the Modified Version, as the publisher.

D. Preserve all the copyright notices of the Document.

E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.

F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.

G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.

H. Include an unaltered copy of this License.

I. Preserve the section Entitled "History", Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.

J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.

K. For any section Entitled "Acknowledgements" or "Dedications", Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.

L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.

M. Delete any section Entitled "Endorsements". Such a section may not be included in the Modified Version.

N. Do not retitle any existing section to be Entitled "Endorsements" or to conflict in title with any Invariant Section.

O. Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section Entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties—for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

5. COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled "History" in the various original documents, forming one section Entitled "History"; likewise combine any sections Entitled

"Acknowledgements", and any sections Entitled "Dedications". You must delete all sections Entitled "Endorsements."

6. COLLECTIONS OF DOCUMENTS

   You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

   You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

7. AGGREGATION WITH INDEPENDENT WORKS

   A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an "aggregate" if the copyright resulting from the compilation is not used to limit the legal rights of the compilation's users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

   If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document's Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

8. TRANSLATION

   Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

   If a section in the Document is Entitled "Acknowledgements", "Dedications", or "History", the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

9. TERMINATION

   You may not copy, modify, sublicense, or distribute the Document except as expressly provided for under this License. Any other attempt to copy, modify, sublicense or distribute the Document is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

10. FUTURE REVISIONS OF THIS LICENSE

    The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See http://www.gnu.org/copyleft/.

    Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation.

A.0.1 ADDENDUM: How to use this License for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

```
Copyright (C)  year  your name.
Permission is granted to copy, distribute and/or modify this document
under the terms of the GNU Free Documentation License, Version 1.2
or any later version published by the Free Software Foundation;
with no Invariant Sections, no Front-Cover Texts, and no Back-Cover
Texts.  A copy of the license is included in the section entitled ``GNU
Free Documentation License''.
```

If you have Invariant Sections, Front-Cover Texts and Back-Cover Texts, replace the "with...Texts." line with this:

```
with the Invariant Sections being list their titles, with
the Front-Cover Texts being list, and with the Back-Cover Texts
being list.
```

If you have Invariant Sections without Cover Texts, or some other combination of the three, merge those two alternatives to suit the situation.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.

Index