SUSE Linux Enterprise Real Time 15 SP6

Virtualization Guide

SUSE Linux Enterprise Real Time 15 SP6 supports virtualization and Docker usage as a technology preview only (best-effort support).

To see the degree of interference from running within a particular KVM configuration versus running on a bare-metal configuration, each RT application has to be assessed individually. We do not give any specific guarantees on performance or deadlines being missed.

Virtualization inevitably introduces overhead, but there is currently no rule of thumb for the performance penalty incurred. It is up to each RT application developer to set performance and deadline requirements and evaluate if those requirements are met.

This guide provides the following three examples for user reference only.

Publication Date: 12/12/2024

Contents

1 Running RT applications with non-RT KVM guests

It is possible to achieve isolation of real-time workloads running alongside KVM by using standard methods—for example, cpusets and routing IRQs to dedicated CPUs. These can be done using the **`cset`** utility. Both libvirtd and KVM work fine in such configurations. System configurations that share CPUs between both RT and KVM workloads are not supported; proper isolation of workloads is imperative for achieving RT deadline constraints. None of the below observations and recommendations are specific to virtualization. Nevertheless, they can be considered "best-effort" for isolating RT and KVM workloads. The basic steps are:

1. *Section 1.1, "Setup"*
2. *Section 1.2, "Observations"*
3. *Section 1.3, "Recommendations"*

1.1 Setup

All examples were carried out on a 48-core Xeon machine with two NUMA nodes and 64 GB of RAM running SUSE Linux Enterprise Real Time. The virtual machine was installed with **`vm-install`**, running SUSE Linux Enterprise Server on four CPUs and 2 GB of memory. The disk was a physical disk `/dev/sdb` as recommended by the SUSE Linux Enterprise Server *Virtualization Guide*.

The **`cpuset`** utility was used to shield the RT workload from KVM as described in the *SLE RT Shielding Guide* (see *Book "Shielding Linux Resources"*):

```
cset shield --kthread=on -c 8-47
```

Affinity for the KVM vCPU tasks was modified via the **`virsh vcpupin`** command, with a 1-1 mapping. For example, vCPU 0 pinned to CPU 0, etc.

The CPUs were split into two groups. CPUs 0-7 were allocated to the `system` cpuset, and CPUs 8-47 were allocated to the `user` group. Having CPUs on the same socket in two groups was done intentionally to monitor the effects on shared CPU resources, such as last-level cache (LLC).

The RT workload used throughout is `cyclictest`, executed as so:

```
cset shield --exec cyclictest -- -a 8-47 -t 40 -n -m -p99 -d 0 -D 120 --quiet
```

1.2 Observations

The following observations were made:

1. VM Heavy I/O

   The test for this was to do the following in a VM:

   ```
   dd if=/dev/zero of=empty bs=4096 count=$(((80*1024*1024)/4096))
   ```

   Doing large amounts of disk I/O in the VM guests has a noticeable impact on the latency of RT tasks. This is because of the constant eviction of LLC data, resulting in more cache misses.

   The maximum latencies in for the real-time workload are seen on those CPUs on the same socket as the CPUs available to the KVM workload. For example, where the LLC is a shared resource between the `system` and `user` cpuset.

2. cpufreq drivers incur timer latency

   Drivers like `intel_pstate` will set up a timer on each CPU to periodically sample and adjust the CPU's current P-state. If this fires at an inopportune time it can add delays to the scheduling of RT tasks, particularly because lots of the IRQ/timer code paths run with interrupts disabled.

3. Interrupt handling introduces delays

   The handling of interrupts can result in latencies that affect RT workloads. Interrupts should be routed to “housekeeping” CPUs that are not running RT applications.

4. Some kernel threads cannot be controlled with cpuset

   Performing heavy I/O in the VM may cause kthreads to be scheduled on the CPUs dedicated for RT. This can occur, for example, when a kthread is flushing dirty pages to disk.

   While it is impossible to move some kworker threads into the `system` cpuset, the above issue can be mitigated by setting the CPU affinity for those threads via:

   ```
   /sys/devices/virtual/workqueue/writeback/cpumask
   ```

1.3 Recommendations

Suggestions for tuning machines running both RT and KVM workloads are as follows:

1. Use CPU affinity to schedule RT tasks to their own CPUs, and if possible, to CPUs on their own dedicated socket. Using a dedicated socket avoids the issue from *Section 1.2, "Observations"* above, where the LLC occupancy is churned by VMs doing lots of I/O operations. If that is not an option, some customers should look at Intel's Cache Allocation Technology to further enforce cache-allocation policies.

2. Disable drivers that arm per-CPU timers such as cpufreq drivers, for example, `intel_pstate=disable`.

3. Set IRQ affinity to CPUs that are not running RT workloads and disable irqbalance.

4. Set IRQ affinity to CPUs that are not running RT workloads. This can be achieved by setting the `IRQBALANCE_BANNED_CPUS` environment variable used by **`irqbalance`**(1) with a bitmask of banned CPUs. For the examples used throughout this document the following setting was used:

```
IRQBALANCE_BANNED_CPUS="ffff,ffffff00"
```

5. Search for cpumask control files in `/sys` and set them appropriately for those cases that cannot be controlled via cpuset. The following command will list those files:

```
find /sys -name cpumask
```

2 Running real-time applications within Docker

It is important to note that real-time processes will be affected by container activity as there is insufficient isolation to guarantee zero cross-talk. There are no special settings, nor container-specific interactions to consider as from a RT prespective, nothing changes due to containers. Whether a noise source in a container is irrelevant. Interference may be considerably higher if multiple RT applications are executed in separate containers. Also bear in mind that while worst-case latency may be better than SLE, it will not necessarily perform better than NOPREEMPT due to the overhead required for RT.

Some shielding is possible but there is no tool-based support for it. There is a generic shield script attached that can move Docker contents onto shielded cores once running. Launching of either KVM or Docker directly into a shielded home did not appear to be possible but the Docker or virtualization team may be able to do better. The basic steps are:

1. *Section 2.1, "Running real-time applications in a virtualized environment"*
2. *Section 2.2, "Docker shielding"*
3. *Section 2.3, "Scripts"*

2.1 Running real-time applications in a virtualized environment

If you intend to run compute-intensive applications with real-time priority, you must make sure that kernel threads cannot starve. (This is general advice that applies to other real-time scenarios as well.)

A simple precaution is to use the `rtkthread=PRIORITY` and `rtworkqueues=PRIORITY` kernel boot parameters. Set the `PRIORITY` values higher than the priority of any process that has the potential to dominate a CPU. This is not strictly real-time capable, but it is safer overall.

DOCKER PREREQUISITES

- The kernel must be booted with `nortsched` command-line parameter
  This is to hide cgroup scheduling from Docker. If cgroup scheduling is required, then isolating Docker containers is very problematic.
- The **`docker run`** command must be passed `--privileged=true`.
  This is required for using the RT classes.
- Your container is equipped with the **`chrt`** system tool.

If no isolation is required for your use case, then it is ready. Start your container with **`docker run`**, using **`chrt`** to set the RT class/priority of the program you execute when starting of the container. For example:

```
docker run --privileged=true ... /usr/bin/chrt -f 1 /usr/sbin/sshd -D
```

The above (with additional arguments, of course) will start `sshd` within the container as a `SCHED_FIFO` task of priority 1. `ssh` into it, and whatever you run in the container will inherit the scheduler's RT class/priority.

2.2 Docker shielding

There is currently no facility within Docker to launch a container directly into an isolated cpuset. You must do this manually.

EXAMPLE 1: PSEUDO SCRIPT

```
# note cpuset mount point
cpuset_mnt=$(mount|grep cpuset|cut -d' ' -f3)

# create an isolated cpuset for your container
cset shield --userset=rtcpus --cpu=4-7 --kthread=on

# note path and id of your container
docker_path=$(docker run...)
docker_id=$(docker ps -q)

# move container content into the isolated cpuset
for i in $(cat ${cpuset_mnt}/system/docker/${docker_path}/tasks);
do
  echo $i > ${cpuset_mnt}/rtcpus/tasks;
done

# stop/destroy the container
docker stop ${docker_id}
docker rm ${docker_id}

# remove dir docker abandons in the shield system directory
rmdir ${cpuset_mnt}/system/docker

# tear down the shield, and you're done
cset shield --userset=rtcpus --cpu=4-7 --reset
```

2.3 Scripts

EXAMPLE 2: SAMPLE SHIELD SCRIPT

```
#!/bin/sh

let START_CPU=4
let END_CPU=63
let ONLINE=1
let SHIELD_UP=0
GOVERNOR="performance"

DEFAULT_MASK=ffffffff,ffffffff
```

```
SHIELD_MASK=00000000,0000000f

if [ -f /proc/sys/kernel/sched_rt_runtime_us ]; then
  RT_RUNTIME=$(cat /proc/sys/kernel/sched_rt_runtime_us)
fi
if [ -f /proc/sys/kernel/nmi_watchdog ]; then
  NMI_WATCHDOG=$(cat /proc/sys/kernel/nmi_watchdog)
fi

CPUSET_ROOT=$(grep cpuset /proc/mounts|cut -d ' ' -f2)
if [ ! -z $CPUSET_ROOT ]; then
  if [ -d ${CPUSET_ROOT}/rtcpus ]; then
    let SHIELD_UP=1
  fi
  if [ -f ${CPUSET_ROOT}/cpuset.cpus ]; then
    CPUSET_PREFIX=cpuset.
  fi
fi

if [ $SHIELD_UP -eq 1 ]; then
  # take it down
  echo 1 > ${CPUSET_ROOT}/${CPUSET_PREFIX}sched_load_balance
  cset shield --userset=rtcpus --reset

  # restore default irq affinity
  echo ${DEFAULT_MASK} > /proc/irq/default_smp_affinity
  for irqlist in $(ls /proc/irq/*/smp_affinity); do
    echo ${DEFAULT_MASK} > $irqlist 2>/dev/null
  done

  if [ -f /proc/sys/kernel/timer_migration ]; then
    echo 1 > /proc/sys/kernel/timer_migration
  fi
  if [ -f /proc/sys/kernel/sched_rt_runtime_us ]; then
    echo ${RT_RUNTIME} > /proc/sys/kernel/sched_rt_runtime_us
  fi
  if [ -f /sys/kernel/debug/tracing/tracing_on ]; then
    echo 1 > /sys/kernel/debug/tracing/tracing_on
  fi
  if [ -f /sys/kernel/mm/transparent_hugepage/enabled ]; then
    echo always > /sys/kernel/mm/transparent_hugepage/enabled
  fi
  if [ -f /proc/sys/kernel/nmi_watchdog ]; then
   echo ${NMI_WATCHDOG} > /proc/sys/kernel/nmi_watchdog
  fi
  if [ -f /sys/devices/system/machinecheck/machinecheck0/check_interval ]; then
   echo 300 > /sys/devices/system/machinecheck/machinecheck0/check_interval
```

```
  fi
  if [ -f /sys/devices/virtual/workqueue/writeback/cpumask ]; then
   echo ${DEFAULT_MASK} > /sys/devices/virtual/workqueue/writeback/cpumask
  fi
  if [ -f /sys/devices/virtual/workqueue/cpumask ]; then
    echo ${DEFAULT_MASK} > /sys/devices/virtual/workqueue/cpumask
  fi
  if [ -f /proc/sys/vm/stat_interval ]; then
    echo 1 > /proc/sys/vm/stat_interval
  fi
  if [ -f /sys/module/processor/parameters/latency_factor ]; then
   echo 2 > /sys/module/processor/parameters/latency_factor
  fi
  if [ -f /sys/module/processor/parameters/ignore_ppc ]; then
   echo 0 > /sys/module/processor/parameters/ignore_ppc
  fi
  if [ -f /sys/module/processor/parameters/ignore_tpc ]; then
   echo 0 > /sys/module/processor/parameters/ignore_tpc
  fi
  if [ -f /etc/init.d/sgi_irqbalance ]; then
   /etc/init.d/sgi_irqbalance start
  fi
else
  # route irqs away from shielded cpus
  if [ -f /etc/init.d/sgi_irqbalance ]; then
    /etc/init.d/sgi_irqbalance stop
  fi
  echo $SHIELD_MASK > /proc/irq/default_smp_affinity
  for irqlist in $(ls /proc/irq/*/smp_affinity); do
    echo $SHIELD_MASK > $irqlist 2>/dev/null
  done

  # poke some buttons..
  if [ -f /proc/sys/kernel/sched_rt_runtime_us ]; then
    echo -1 > /proc/sys/kernel/sched_rt_runtime_us
  fi
  if [ -f /sys/kernel/debug/tracing/tracing_on ]; then
    echo 0 > /sys/kernel/debug/tracing/tracing_on
  fi
  if [ -f /sys/kernel/mm/transparent_hugepage/enabled ]; then
    echo never > /sys/kernel/mm/transparent_hugepage/enabled
  fi
  if [ -f /proc/sys/kernel/nmi_watchdog ]; then
    echo 0 > /proc/sys/kernel/nmi_watchdog
  fi
  if [ -f /sys/devices/system/machinecheck/machinecheck0/check_interval ]; then
    echo 0 > /sys/devices/system/machinecheck/machinecheck0/check_interval
```

```
fi
if [ -f /sys/devices/virtual/workqueue/writeback/cpumask ]; then
  echo ${SHIELD_MASK} > /sys/devices/virtual/workqueue/writeback/cpumask
fi
if [ -f /sys/devices/virtual/workqueue/cpumask ]; then
  echo ${SHIELD_MASK} > /sys/devices/virtual/workqueue/cpumask
fi
if [ -f /proc/sys/vm/stat_interval ]; then
  echo 999999 > /proc/sys/vm/stat_interval
fi
if [ -f /sys/module/processor/parameters/latency_factor ]; then
  echo 1 > /sys/module/processor/parameters/latency_factor
fi
if [ -f /sys/module/processor/parameters/ignore_ppc ]; then
  echo 1 > /sys/module/processor/parameters/ignore_ppc
fi
if [ -f /sys/module/processor/parameters/ignore_tpc ]; then
  echo 1 > /sys/module/processor/parameters/ignore_tpc
fi

# ...and fire up the shield
cset shield --userset=rtcpus --cpu=${START_CPU}-${END_CPU} --kthread=on

# If cpuset wasn't previously mounted (systemd will, like it or not),
# it has now been mounted. Find the mount point.
if [ -z $CPUSET_ROOT ]; then
 CPUSET_ROOT=$(grep cpuset /proc/mounts|cut -d ' ' -f2)
 if [ -z $CPUSET_ROOT ]; then
   # If it's not mounted now, bail.
   echo EEK, cupset is not mounted!
   exit
 else
   # ok, check for cgroup mount
   if [ -f ${CPUSET_ROOT}/cpuset.cpus ]; then
    CPUSET_PREFIX=cpuset.
   fi
 fi
fi

echo 0 > ${CPUSET_ROOT}/${CPUSET_PREFIX}sched_load_balance
echo 1 > ${CPUSET_ROOT}/system/${CPUSET_PREFIX}sched_load_balance
echo 0 > ${CPUSET_ROOT}/rtcpus/${CPUSET_PREFIX}sched_relax_domain_level
# this ain't gonna happen in -rt kernels, but...
if [ -f ${CPUSET_ROOT}/rtcpus/cpu.rt_runtime_us ]; then
  echo 300000 > ${CPUSET_ROOT}/system/cpu.rt_runtime_us
  echo 300000 > ${CPUSET_ROOT}/rtcpus/cpu.rt_runtime_us
fi
```

```
  echo 0 > ${CPUSET_ROOT}/rtcpus/${CPUSET_PREFIX}sched_load_balance

  # wait a bit for sched_domain rebuild
  sleep 1

  # now go to hpc
  if [ -f ${CPUSET_ROOT}/rtcpus/${CPUSET_PREFIX}sched_hpc_rt ]; then
    echo 1 > ${CPUSET_ROOT}/rtcpus/${CPUSET_PREFIX}sched_hpc_rt
  fi

  # offline/online to migrate timers and whatnot
  if [ $ONLINE -eq 1 ]; then
    for i in `seq ${START_CPU} ${END_CPU}`; do
      echo 0 > /sys/devices/system/cpu/cpu$i/online
    done
    for i in `seq ${START_CPU} ${END_CPU}`; do
      echo 1 > /sys/devices/system/cpu/cpu$i/online
    done

    # re-add CPUs the kernel removed on offline
    echo ${START_CPU}-${END_CPU} > ${CPUSET_ROOT}/rtcpus/${CPUSET_PREFIX}cpus

    # and prioritize re-initialized kthreads
    systenctl restart set_kthread_prio
  fi
  if [ -f /proc/sys/kernel/timer_migration ]; then
    echo 0 > /proc/sys/kernel/timer_migration
  fi
  GOVERNOR="performance"
fi

if [ -f /sys/devices/system/cpu/cpu0/cpufreq/scaling_governor ]; then
  CURRENT_GOVERNOR=$(cat /sys/devices/system/cpu/cpu0/cpufreq/scaling_governor)
  if ! [ $GOVERNOR = $CURRENT_GOVERNOR ]; then
    for i in $(ls /sys/devices/system/cpu/cpu*/cpufreq/scaling_governor); do
     echo $GOVERNOR > $i;
    done
  fi
fi
```

EXAMPLE 3: PATCH TO `sysjitter` TO USE THE USER AFFINITY INSTEAD OF WHOLE BOX

```
sysjitter.c |   10 +++++++---
 1 file changed, 7 insertions(+), 3 deletions(-)

--- a/sysjitter.c
+++ b/sysjitter.c
```

```
@@ -412,7 +412,7 @@ static void write_raw(struct thread *thr
  FILE *f;
  int i;
  for (i = 0; i < g.n_threads; ++i) {
-  sprintf(fname, "%s.%d", outf, i);
+  sprintf(fname, "%s.%d", outf, threads[i].core_i);
   if ((f = fopen(fname, "w")) == NULL) {
    fprintf(stderr, "ERROR: Could not open '%s' for writing\n", fname);
    fprintf(stderr, "ERROR: %s\n", strerror(errno));
@@ -578,6 +578,7 @@ int main(int argc, char *argv[])
  const char *outf = NULL;
  char dummy;
  int i, n_cores, runtime = 70;
+ cpu_set_t cpus;

  g.max_interruptions = 1000000;

@@ -609,10 +610,13 @@ int main(int argc, char *argv[])
      sscanf(argv[0], "%u%c", &g.threshold_nsec, &dummy) != 1)
   usage(app);

+ CPU_ZERO(&cpus);
+ sched_getaffinity(0, sizeof(cpus), &cpus);
+
  n_cores = sysconf(_SC_NPROCESSORS_ONLN);
- TEST(threads = malloc(n_cores * sizeof(threads[0])));
+ TEST(threads = malloc(CPU_COUNT(&cpus) * sizeof(threads[0])));
  for (i = 0; i < n_cores; ++i)
-  if (move_to_core(i) == 0)
+  if (CPU_ISSET(i, &cpus) && move_to_core(i) == 0)
    threads[g.n_threads++].core_i = i;

  signal(SIGALRM, handle_alarm);
```

3 Running RT applications with RT KVM guests

Section 1, "Running RT applications with non-RT KVM guests" shows that it is possible to isolate real-time workloads running alongside KVM by using standard methods. In SLE RT 15 SP2 this can be done in user space using libvirt/qemu.

Applications and guest operating systems run inside KVM guests similarly to how they run on bare metal. Guests interface with emulated hardware presented by QEMU, which submits I/O requests to the host on behalf of the guest. Then the host kernel treats the guest I/Os like any user-space application.

In SLE RT 15 SP4, both QEMU and libvirt support isolating the CPUs, partitioning the memory for guests, and setting the vCPU/iothread scheduler policy and priority for running both non-RT KVM and RT KVM.

3.1 Support of QEMU/libvirt

1. QEMU includes the `-realtime mlock=on|off` option. Mlocking QEMU and guest memory is enabled with `mlock=on` (which is enabled by default) .
2. libvirt supports CPU Allocation, CPU Tuning, and Memory Backing, which allows you to control RT parameters, see *Section 3.2, “Sample of* `libvirt.xml`*”*.

   CPU allocation

   You can define the maximum number of virtual CPUs allocated for the guest OS.

   CPU tuning

   - Pinning is a tuning option for the virtual CPUs in KVM guests. With pinning, you can control where the guest runs in order to reduce the overhead of scheduler switches, pin vCPUs to physical CPUs that have low utilization, and improve the data cache performance. Overall performance is improved when the memory that an application uses is local to the physical CPU, and the guest vCPU is pinned to this physical CPU.
   - We can specify the vCPU scheduler type (values `batch`, `idle`, `fifo`, or `rr`), and priority for particular vCPU threads. Priority `99` is too high, and it will massively interfere with the host's ability to function properly. There are host-side per-CPU threads that must be always be able to preempt, such as timer `sirq` threads.

   Memory backing

   Use memory backing to allocate enough memory in the guest to avoid memory over-commit, and to lock the guest page memory in host memory to prevent it from being swapped out. This will show a performance improvement in some workloads.

3.2 Sample of `libvirt.xml`

```
<domain>
```

```
    …
    <vcpu placement='static' cpuset="1-4,^3,6" current="1">4</vcpu>
    …
    <cputune>
        <vcpupin vcpu="0" cpuset="1-5,^2"/>
        <vcpupin vcpu="1" cpuset="0,1"/>
        <vcpupin vcpu="2" cpuset="2,3"/>
        <vcpupin vcpu="3" cpuset="0,4"/>
        <vcpusched '0-4,^3' scheduler='fifo' priority='1'/>
    </cputune>
    …
    <memoryBacking>
        <locked/>
    <memoryBacking/>
    …
</domain>
```

3.3 Other host settings

1. **Power management.** Intel processors have a power management feature that puts the system into power-saving mode when the system is under-utilized. The system should be configured for maximum performance, rather than allowing power-saving mode.

2. **Turboboost and Speedstep.** Turboboost overclocks a core when CPU demand is high, whereas Speedstep dynamically adjusts the frequency of processor to meet processing needs. Turboboost requires Speedstep to be enabled, as it is an extension of Speedstep. For maximum performance, enable both Turboboost and Speedstep in the BIOS. The host OS may also need configuration to support running at higher clock speeds. For example:

   ```
   cpupower -c all frequency-set -g performance
   ```

3. **Disable interrupt balancing (irqbalance).** The irqbalance daemon is enabled by default. It distributes hardware interrupts across CPUs in a multi-core system to increase performance. When irqbalance is disabled, all interrupts will be handled by cpu0, and therefore the guest should NOT run on cpu0.

4. **RT throttling.** The default values for the realtime throttling mechanism allocate 95% of the CPU time to realtime tasks, and the remaining 5% to non-realtime tasks. If RT throttling is disabled, realtime tasks may use up to 100% of CPU time. Hence, programming failures in real-time applications can cause the entire system to hang because no other task can preempt the realtime tasks.

The above settings are just part of the configurations for the RT KVM to run at the “best-effort” performance. Other factors must be considered, such as storage and network. The overall KVM performance is dependent on the host hardware, firmware, BIOS settings, and the guest OS and application charactistics.

A GNU Licenses

This appendix contains the GNU Free Documentation License version 1.2.

GNU Free Documentation License

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/ (https://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.

ADDENDUM: How to use this License for your documents

```
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 &ldquo;GNU
Free Documentation License&rdquo;.
```

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.