# Methods to Reduce PC Overhead When Using Virtualization Technology in Volunteer Computing

**Author:** Khrapov Nikolay
Institute for Information Transmission Problems of the Russian Academy of Sciences (IITP RAS)

# Make minimal VB-image for your task

- Remove all unnecessary things from the standard system layout.

- Integrate only the necessary packages into the empty image.

# Remove all unnecessary things

- Remove packets:

  apt remove acpi acpid busybox debconf-i18n eject groff-base iamerican ibritish info ispell laptop-detect logrotate installation-report manpages man-db net-tools os-prober rsyslog tasksel tasksel-data traceroute usbutils wamerican ss

  dpkg --purge `dpkg --get-selections | grep deinstall | cut -f1`

- Remove folders:

  /usr/share/locale/ - since we don't need all locales.

  /usr/share/doc/ - since we don't need the documentation.

  /usr/share/man/ - since we don't need the manuals and we've removed the man program.

  /var/log/ - since we won't be needing all that logging.

  /var/cache/debconf/ - since that cache is disposable.

  /var/lib/apt/lists/ - since those lists are huge and can quickly be recreated with apt-get update.

- **Image size is 2 GB**

# Integrate only the necessary

- Use debootstrap for debian-based linux (size of root partition is 300MB):

  debootstrap --arch amd64 bookworm rootfs http://http.debian.net/debian

  Standart edition has no kernel and kernel utilities.

- For minimal OS use option --variant=minbase. Size of the root partition in minbase is 200МБ. Minbase variant has no systemd and nano edit.

```
# Copying files to /mnt/rootfs directory

debootstrap  --arch amd64 bookworm /mnt/rootfs http://http.debian.net/debian

# Set the Linux environment
mount --make-rslave --rbind /dev /mnt/rootfs/dev
mount --make-rslave --rbind /run /mnt/rootfs/run
mount --make-rslave --rbind /proc/ /mnt/rootfs/proc/
mount --make-rslave --rbind /sys/ /mnt/rootfs/sys/
chroot /mnt/rootfs/ /bin/bash

# Install necessary packages (mpi as example)
apt install openmpi-bin

# Cleaning
apt-get clean
apt-clean
```

# Compression for OS

```
# Get list of partitions
grub> ls
(hd0) (hd0,msdos1) (hd0,msdos2) (hd1) (hd1,msdos1)

# Set root as partition with kernel and initrd
set root=(hd1,msdos1)

# Load the kernel
grub> linux /vmlinuz-6.1.0.25-amd64 root=/dev/mapper/rootfs rw break=premount

# Load the initrd
grub> initrd /initrd.img-6.1.0-25-amd64

# Boot the operation system
grub> boot
```

```
# GUEST_ROOT is a way to content of root partition
# of the guest system
dd if=/dev/random of=template.img bs=1M count=512
mkfs.btrfs template.img

echo "ZLIB INFO:"
cp template.img zlib.img
mkdir zlib_mp
mount -o compress=zlib:9 zlib.img zlib_mp/
cp -r ${GUEST_ROOT} zlib_mp
df zlib_mp/
btrfs filesystem df zlib_mp/

echo "LZO INFO:"
cp template.img lzo.img
mkdir lzo_mp
mount -o compress=lzo lzo.img lzo_mp/
cp -r ${GUEST_ROOT} lzo_mp
df lzo_mp/
btrfs filesystem df lzo_mp/

echo "ZSTD INFO:"
cp template.img zstd.img
mkdir zstd_mp
mount -o compress=zstd:15 zstd.img zstd_mp/
cp -r ${GUEST_ROOT} zstd_mp
df zstd_mp/
btrfs filesystem df zstd_mp/
```

| Algorithm | standart | minbase |
|---|---|---|
| ucomperessed | 300 MB | 200 MB |
| ZLIB | 172 MB | 120 MB |
| LZO | 221 MB | 156 MB |
| ZSTD | 165 МБ | 114 MB |

# Use RAM instead disk memory

- The result is a structured file 500 MB in size.
- The compressed result is 50 MB.
- The amount of data to be overwritten is 550 MB.

# Use RAM instead disk memory

- Before starting the problem-oriented sequence of operations, the control script mounts a tmpfs partition in the system. An example of running the command is as follows:

  ```
  mount -t tmpfs -o size=1000MB tmpfs /mnt/tmpwork
  ```

- where /mnt/tmpwork is the working directory of the application.
- The necessary input data files are copied to this partition.
- The problem-oriented sequence is launched from the tmpfs directory.