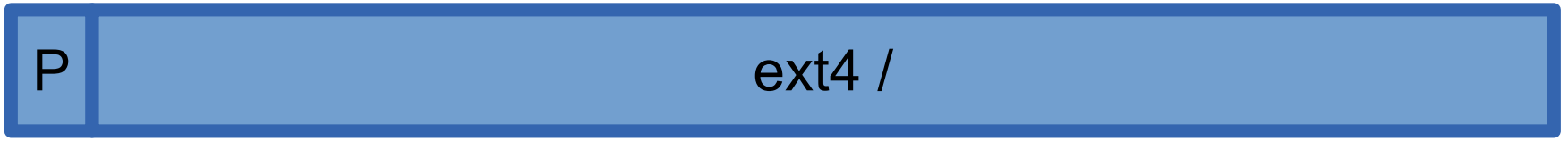
Automated Instance Creation

Network Startup Resource Center

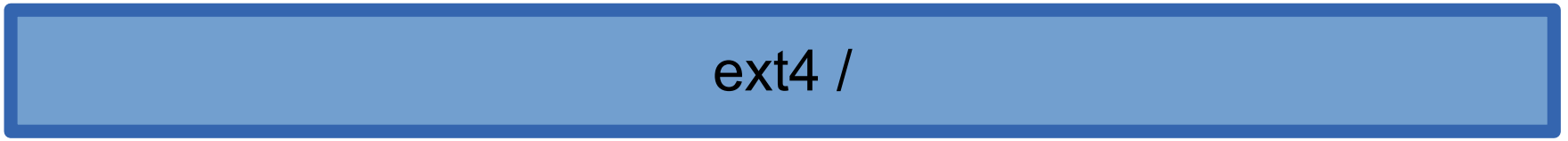
Partitioning: choices, choices!

MSDOS partitioning, single partition, Linux filesystem



MSDOS partitioning, multiple partitions

No partitions (whole disk filesystem)



Master boot record is here (required to boot from the instance disk, and contains partition table)

Why does partitioning matter?

- To automatically create instances you need to have scripts to make the final adjustments
  - e.g. change hostname, change IP address, set password...
- Hence the guest filesystem(s) must be mounted temporarily
- Therefore scripts must know how it's partitioned

With a partition table / MBR

- BIOS boots from the Master Boot Record in the guest's disk image
  - standard, just like a normal PC
- Fairly easy to grow the *last* partition
  - increase size of disk
  - delete and re-add the last partition
  - grow the filesystem (resize2fs)

Without a partition table / MBR

- Very easy to grow filesystem
  - increase size of disk, grow the filesystem (resize2fs)
  - but no partitions = no MBR = can't boot from disk!
- KVM can be told to boot a kernel and initrd stored on the *host* filesystem
  - -H kernel_path, initrd_path, kernel_args
  - no bootstrap loader (e.g. GRUB) required
  - only works for Linux VMs

Instance export/import

- The OS definition scripts also include 'export' and 'import' functions
- Useful for backups and for moving instances
  - gnt-backup {export|import}
  - gnt-instance move
- See "man gnt-backup" and "man gnt-instance"
- Different OS definitions export in different ways

Filesystem export/import

- Some OS definitions use a tool like "dump/restore" which understands the filesystem and skips over unused blocks
- Script selects the data partition(s) to export
- Can skip the swap partition
  - it would be a waste of space
  - it would be a security risk (contains RAM contents)
- Only works if the disk is partitioned and formatted in the way the scripts expect

Raw images

- Some OS definitions export a raw block-by-block image of the entire virtual disk
- Works with any filesystem
- Very large (but usually compresses well)
- Less efficient than ext3/4 dump/restore as it will include blocks from deleted files
- Can only be restored to a disk image of *exactly the same size*

Exporting: non-Linux guests

e.g. Windows instance

- This type of filesystem can only be exported as a raw disk image

Questions?

ganeti-os-noop

- Does no partitioning or OS installation
- The CD-ROM you're installing from can partition the disk however you like
- export/import scripts create raw dumps
  - Therefore can be used to export a VM's disks (even non-Linux filesystems) and restore them to disks of same size

ganeti-instance-debootstrap

- OS definition to automatically install Debian or Debian-derived OS (e.g. Ubuntu)
- Downloads all the .deb packages and unpacks them
- Creates a cache so that subsequent installs are very fast

ganeti-instance-debootstrap

- Default is to create a partition table and *one* partition

- Or you can install without any partition table
  - PARTITION_STYLE="none"
  - makes resizing filesystem very easy
  - but then you *must* boot from a kernel on the host

Debootstrap booting

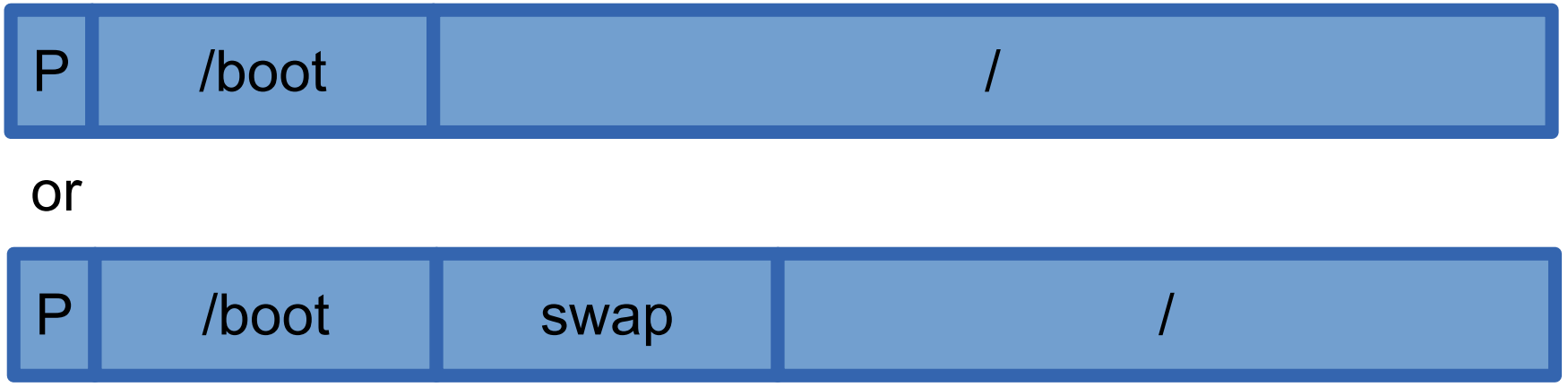
- If you want to boot from a kernel in the guest filesystem you need to install grub in the guest
- An example hook script is provided to do this, but it doesn't work properly :-(
- Good practice in how to recover a VM with broken boot loader :-)
- Or you can boot from a kernel on the host
  - reasonable if you have many nearly-identical VMs

ganeti-instance-image

- Installs OS from tarball or filesystem dumps that you previously prepared
- Two distinct versions
  - original from OSU OSL (described here)
  - GRNET's extended version for GanetiMgr
  - confusingly, both same name and version 0.6
- Fixed partitioning scheme. Linux guests only
  - snf-image is a more flexible alternative

ganeti-instance-image

- Script creates 2 or 3 partitions (boot+root, or boot+swap+root) and unpacks one tarball or two filesystem images
  - See `/usr/share/ganeti/os/image/create`



Consequences

- If you are preparing an OS image to clone using instance-image, you need to create either a full system tarball or two dumps (boot and root)
  - The README file explains this
  - /usr/share/doc/ganeti-instance-image/README.gz
- Needs a script to install grub
- Export does two filesystem dumps and wraps them in a tar file
  - can also do a raw dump if set NOMOUNT=yes

Now the good news!

- The instance creation scripts are simple, easy to read/understand and modify
- Look in directories under /usr/share/ganeti/os/
- Settings in `/etc/ganeti/<xxx>/*` and `/etc/default/ganeti-<xxx>`
- Documentation: "man ganeti-os-interface"
  - docs.ganeti.org/ganeti/master/man/ganeti-os-interface.html

Finally, two more options

Importing existing image

- You can take a VM image created somewhere else and copy the disk
  - VM "appliances" are now supplied this way
- You will need to convert the disk image
  - "qemu-img convert" will do this
- Create instance with exactly the right sized disk
- Beware writing to drbd-replicated volumes
  - Safer to create -t plain, then convert to -t drbd

snf-image

- Standalone component of the synnefo cloud solution
- Provides ready-made images you can clone
  - or create your own (snf-image-creator tool)
- Very robust
- Works with Windows and BSD images too!

snf-image architecture

- Uses raw disk dumps
  - boot loader already installed, no messing around
  - partition however you like (except no LVM)
- Post-install "helper" enlarges the last partition and filesystem to fit the chosen disk size
  - also sets password, installs ssh keys etc
  - helper runs inside a temporary VM for security
  - Ganeti is moving to this model too

Summary of installation options

- ganeti-os-noop: install manually from ISO
- instance-debootstrap (Debian/Ubuntu only)
- instance-image: unpack filesystem dumps or tarballs that you prepared earlier
- import an existing VM disk image
- snf-image
  - probably the best option for self-service installs

Exercises (depending on time)

- Create VM using debootstrap
- Create VM using VDMK disk image
- Create VM using snf-image