# Core ideas

- Don't rely on pacman at all, in fact, we should just disable pacman entirely. We update the entire system in a similar fashion to Android: Entire system is read-only. User data stored separately. Updates are applied as images written over the old system. Another thing is that we don't expect the user to install extra packages, as doing so LIKELY will make Arch update its packages with its repositories, if we're keeping lower package versions, that will probably install a bunch of mixed packages on root.
- Uses the same images as release images.
- Total control the set of packages and applications preinstalled. Guarantees that successful updates ARE successful. We don't need any kind of clean up/re-configure
- Usable with only the winesapOS disk, it's no fun if we can just ask the user to boot into another environment then image the disk themselves
- A failed update should be recoverable from by just re-imaging the system image

Downsides:

- System is offline while updating, so it's not completely transparent

Plans (if this follows):

- Partition the release image like this:
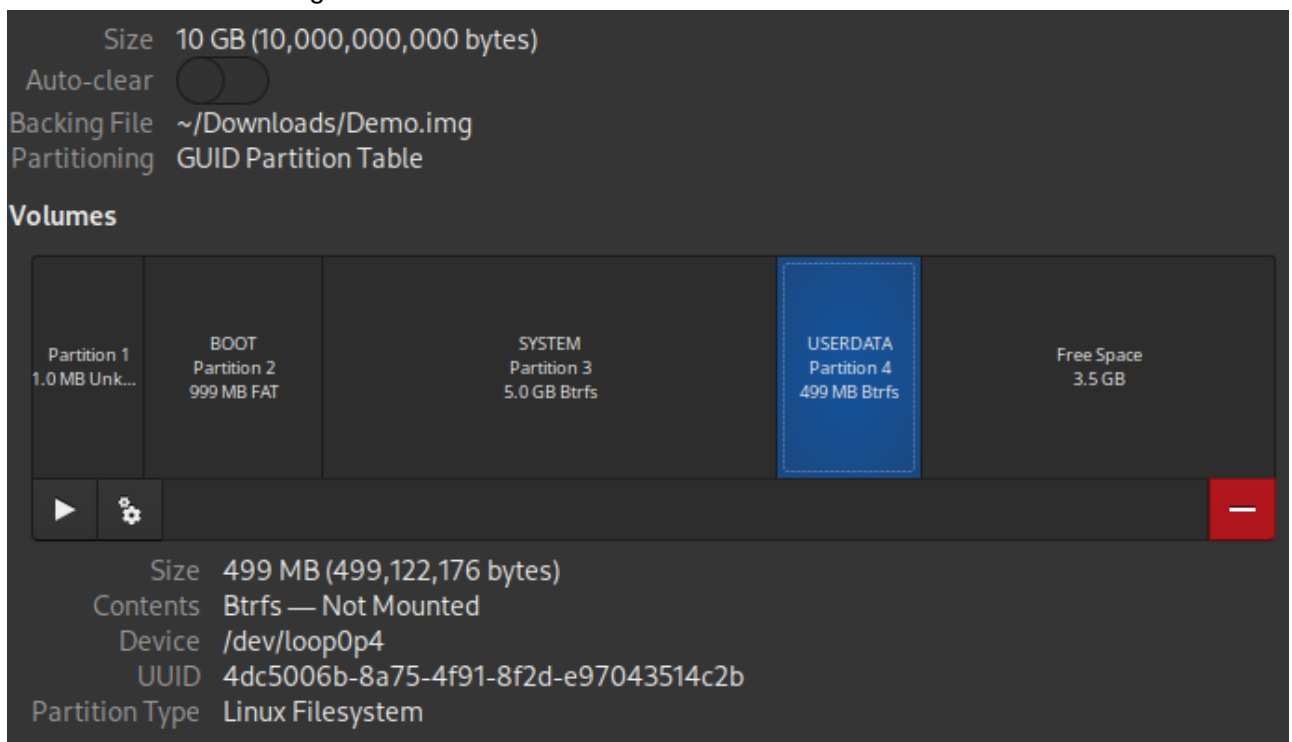
Here's what they do

- Partition 1: BIOS Boot (can be thrown away, but allows winesap to be compatible with legacy boot if the user need it)
- Partition 2: BOOT: contains the bootloader
- Partition 3: SYSTEM (winesap rootfs)
  - mounted r/w or r/o is your choice, but we could make it partially writable later with overlayfs so that users can customize if need be.

- Partition 4: USERDATA: very small initial partition for /home, just enough for us to pre-load flatpak should be sufficient
  - Mounted at /home
  - Has a flag `.firstboot`. Upon reading this file (probably with a systemd service?) , the OS should resize the partition to fill the rest of space

Inject a custom script into `initramfs` that does the follows:

- Unmounts everything
- Mounts USERDATA as read-only
- Read a specific image file on USERDATA and mounts it as loop.
- Beg the user not to unplug life support.
- dd the `SYSTEM` partition of the image to the `SYSTEM` partition on disk. — this part needs attention. Overwriting something else would be... not good 😛
- Verifies the update has completed as expected (maybe with a quick hash?)
- Reboots.

The update process:

- User starts update from winesap-update in the OS.
- winesap-update downloads the update from GitHub and puts it on the Home directory.
- winesap-update then trigger a reboot to initramfs (might be difficult, we'll see. I know some systems like dracut is capable of restoring the initramfs. If it is then we can just reboot)
- winesap-update execute the custom script above, takes care of any significant changes, etc.
- Reboots.
- New system image, /home kept (and presumably user-installed flatpaks are kept)