

fstab

The options can usefully include noatime for better performance (we do not know of any software we run that uses the atime).

/etc/fstab can be prettified using this script (copied from <http://unix.stackexchange.com/questions/96037/how-to-align-fstab-entries-easily>):

```
#!/bin/bash
# usage: fstabalign [FILE]
# This script will output fstab or other file as column aligned.
# It will not alter blank lines or #hash comments.
if [ -z "$1" ]; then
    FILE=$(cat /etc/fstab)
else
    FILE=$(cat "$1")
fi
# Separate the file contents into aligned and unaligned parts.
OUT_ALIGNED=$(echo "$FILE" | sed 's/^\s*#.*/' | nl -ba | column -t)
OUT_UNALIGNED=$(echo "$FILE" | sed 's/^\s*[^\#].*/' $src | nl -ba)
# Remerge aligned and unaligned parts.
while read; do
    line_aligned="$REPLY"
    read -u 3; line_unaligned="$REPLY"
    line_aligned=$(echo "$line_aligned" | sed 's/\s*[0-9]*\s*//')
    line_unaligned=$(echo "$line_unaligned" | sed 's/\s*[0-9]*\s*//')
    echo "$line_aligned$line_unaligned"
done < <(echo "$OUT_ALIGNED") 3< <(echo "$OUT_UNALIGNED")
```

When run with no arguments, it processes /etc/fstab and writes the aligned version to stdout, from where it can be copied and pasted.

The result, after some manual tidying, is for example:

```
# /etc/fstab: static file system information.
#
# <file system>          <mount point> <type> <options>          <dump> <pass>
/dev/mapper/pitanga--system-home /home      ext4    noatime              0      2
/dev/mapper/pitanga--system-root /           ext4    errors=remount-ro,noatime 0      1
/dev/mapper/pitanga--system-swap none        swap    sw                   0      0
/dev/mapper/pitanga--system-var /var       ext4    noatime              0      2
UUID=ef9a6540-774f-449f-8be4-35592905002d /boot      ext3    defaults             0      2
proc /proc   proc      nodev,noexec,nosuid 0      0
```

TODO: enhance the script so less manual tidying is required