

# Application Cache Files

---

 [readme.phys.ethz.ch/linux/application\\_cache\\_files/](https://readme.phys.ethz.ch/linux/application_cache_files/)

Some files in users' home directories should be symlinked or via automount redirected to the local disk instead of cluttering NFS homes.

## Ways to redirect such files

### Planned Logic

- Is it already a Symlink? -> If yes, create local directory if non-existent (if /scratch is full in this case -> Mail to ISG). If no continue
- Check if there's enough disk space on /scratch/
  - Directory > 100 MB -> Mail to ISG
- Move directory to /scratch or delete it
- Create symlink

### Further details

- Cronjob to clean up /scratch/
- Use /scratch/.cache/\$user/ as base

### Implementation: unburden-home-dir

- `unburden-home-dir` is an implementation of some of these ideas.
- For keeping core dumps off the NFS homes or SSDs: Set `/proc/sys/kernel/core_pattern` accordingly. (See also [corekeeper](#))

### What could work

- Reconfigure each application individually
  - For Mozilla products, see the setting [browser.cache.disk.parent\\_directory](#)
- Use automounter/autofs.
- Use wrappers around applications which create proper symlinks
- Using [libeatmydata](#) (Yeah, that's the meaning: *eat my data*, so some care is necessary), probably as backport of the [eatmydata sid/squeeze package](#).)
- **Login scripts**, e.g. PAM, bashrc, profile, Xsession.d

### What will probably not work

- inotify based scripts
  - inotify does not work efficiently on NFS and will probably make things worse than they are
  - Probably need to be configured per User

- Anything on the file server itself
  - Symlinking to /scratch/whatever-cache doesn't suffice. That directory or file needs to be created locally on all workstations for each user (who uses that software), too.

## Caches

The following files are candidates for redirecting from NFS homes to local disks for performance reasons:

- .cache/
- .config/google-chrome/\*/Thumbnails/
- .mozilla/firefox/\*/Cache/
- .thunderbird/\*/Cache/
- .mozilla-thunderbird/\*/Cache/
- .conkeror.mozdev.org/conkeror/\*/Cache/
- .kazehakase/mozilla/kazehakase/Cache
- .opera/cache/
- .opera/opcache/
- .thumbnails/
- .xsession-errors
  - This can be solved by redirecting output in /etc/X11/Xsession to somewhere else, e.g. logger plus an appropriately configured syslogd.
  - Redirecting to /tmp/xsession-errors-\$USER can cause a full /tmp/ filesystem.
- See also [unburden-home-dir.list in the git repo](#) for the current cache file list in our implementation.

## Core Dumps

- Core dumps are not common for normal users but for developers. And if they happen, they can have several GB of size which you really don't want to save over NFS (or on an SSD).

## strange stuff

- ~/.local/share/simias (rotated iFolder logfiles, already fixed in iFolder configuration)

## Trash and automatic Log Files

The following files are possibly candidates for automatic age-based purging from NFS homes or rotating for performance reasons:

- .Trash/
- .local/Trash/
- .xsession-errors

## Big files which may be needed/wanted by the user

The following files can cause performance issues on NFS homes but changing or removing them is possibly regarded as data loss by the user:

- .config/google-chrome//History
- .liferea\*/cache/
- .opera/vps/
- procmail.log (probably only existent on explicit configuration by user)
- .kde/share/apps/klipper/history2.lst\* (one of our user found one with >500 MB after he ran into his quota)

## Big files which are written in big bursts

The following files can cause performance issues on NFS homes but are definitely privacy issues:

- .gnupg
- .spamassassin
- .liferea\*/liferea.db

Macs with portable homes write their changes over the day back in one big burst in the later afternoon.