

# Nemo Tips

## executable Text Files

By default, Nemo asks whether to display or run executable text files. As all files on Windows Samba shares are executable, this can become annoying very quickly. To change this: Edit → Preferences → Behaviour → Executable Text Files → View executable text files when they are opened

## Nemo keyboard shortcuts

By default, there is no keyboard shortcut to move files/directories from one pane to another (when working with multiple panes by pressing F3). A lot of other functions don't have keyboard shortcuts either.

The shortcuts are stored in `~/.gnome2/acceles/nemo`

Create the folder and edit this file:

```
mkdir -p ~/.gnome2/acceles
vi ~/.gnome2/acceles/nemo
```

or search for "Move to next pane" when the file exists (or any other function you require. Remove the preceding semicolon and define the shortcut key. In this example, the shortcut will be set to CTRL+j;

```
(gtk_accel_path "<Actions>/DirViewActions/Move to next pane" "<Primary>j")
```

Quit nemo from the command line using:

```
nemo -q
```

NOTE: Ensure not to use a shortcut already defined for something else!

## Nemo Context Menus

Nemo supports custom context menus by simply creating a file in an 'actions' folder. Nemo does not need to be restarted to pick up any new or changed actions.

Global action scripts need to be placed in `/usr/share/nemo/actions` and user actions can be placed in `~/.local/share/nemo/actions/`. File extension has to be `.nemo_action`.

A full description of the syntax is available in the sample file `/usr/share/nemo/actions/sample.nemo_action`

More examples can be found at <https://wiki.archlinux.org/index.php/Nemo> The sample file is also here:

[https://github.com/linuxmint/nemo/blob/master/files/usr/share/nemo/actions/sample.nemo\\_action](https://github.com/linuxmint/nemo/blob/master/files/usr/share/nemo/actions/sample.nemo_action)

## Mass renaming

```
sudo apt-get install gprename
```

### [bulk-rename.nemo\\_action](#)

```
[Nemo Action]
Name=Bulk Rename
Comment=Multiple Rename with GPRename. The action gets executed for
the marked or the higher level directory.
Exec=gprename %F
Selection=m
Extensions=any;
EscapeSpaces=true
Dependencies=gprename;
```

## MP3 Tagging example scripts

### [mp3-lyrics.nemo\\_action](#)

```
[Nemo Action]
Name=Add Lyrics to MP3s
Comment=Add Lyrics to MP3s
Exec=gnome-terminal -x sh -c "/home/wuff/getlyric/get_mp3_lyrics.py --
sites genius.com --in_dir '%F' --write2tag"
Icon-Name=gtk-cdrom
Selection=s
Extensions=dir;
Quote=double
EscapeSpaces=true
```

### [mp3-lyrics-2.nemo\\_action](#)

```
[Nemo Action]
Name=Add Lyrics to MP3s
Comment=Add Lyrics to MP3s
Exec=gnome-terminal -x sh -c "/home/wuff/getlyric/get_mp3_lyrics.py --
sites genius.com --in_dir '%P' --write2tag"
Icon-Name=gtk-cdrom
Selection=none
Extensions=dir;
Quote=double
EscapeSpaces=true
```

### [beets\\_import\\_album.nemo\\_action](#)

```
[Nemo Action]
Name=Import Album to Beets
Comment=Import Album to Beets
Exec=<beets_import_album.sh %F>
Icon-Name=gtk-cdrom
Selection=notnone
Extensions=dir;
#Quote=double
EscapeSpaces=true
Terminal=true
```

### beets\_import\_album.sh

```
for var in "$@"
do
    folder=$(basename "$var")
    echo "Importing '$folder' into beets..."
    beet im "$var"
done
sleep 10
```

```
chmod 755 .local/share/nemo/actions/beets_import_album.sh
```

## Nemo Performance

If opening folders with lots of images takes a long time, changes are a plugin is causing this:

disable the plugin "ChangeColorFolder+NemoPython" and restart nemo (nemo -q on command line) and start it again by opening any folder.

If opening folders with videos takes a long time, the thumbnailer for videos can be replaced with ffmpegthumbnailer which is faster (note that on updates of totem the config probably will be replaced):

```
sudo apt-get install ffmpegthumbnailer
cd /usr/share/thumbnaillers/
sudo cp totem.thumbnailer totem.thumbnailer.original
sudo vi totem.thumbnailer

#replace line 2-3 with:
TryExec=/usr/bin/ffmpegthumbnailer
Exec=/usr/bin/ffmpegthumbnailer -s %s -i %i -o %o -c png -f

#exit nemo
nemo -q
```

## Save/Restore session

Execute `nemo_save.sh` to save list of opened nemo windows in `~/.config/nemo/nemo.openwindows`.

Launch `nemo_restore.sh` to reopen saved list of windows.

Restrictions : the script cannot save opened tabs, and won't save a folder whose name is too long. For special folders like Trash, you can replace the window title name in `nemo_save.sh` if you know the full path. Also, only the currently active pane in two-pane mode is being saved.

Original source:

<https://github.com/x-hgg-x/Utils/tree/master/Nemo%20save%20and%20restore%20session>

[~/.local/bin/nemo\\_save.sh](#)

```
#!/bin/bash
declare -x DISPLAY=":0.0"
sleep 5
PATHS=$(wmctrl -lx | grep nemo.Nemo | sed -e "s#$hostname Home# -
$HOME#" | awk -F' - /' '{print "/"$2}')
if [ "$PATHS" == "" ];
then
    echo "" > ~/.config/nemo/nemo.openwindows
else
    echo "$PATHS" > ~/.config/nemo/nemo.openwindows
fi
```

[~/.local/bin/nemo\\_restore.sh](#)

```
#!/bin/bash
sleep 2
IFS=$'\n'; read -d '' -r -a dirs < ~/.config/nemo/nemo.openwindows

for dir in "${dirs[@]}"; do
    nemo "$dir" &
    sleep 0.5
done
```

```
chmod 755 ~/.local/bin/nemo_save.sh ~/.local/bin/nemo_restore.sh
#Add to current user's crontab
(crontab -l ; echo "*/5 * * * * /home/$USER/.local/bin/nemo_save.sh") |
crontab -
```

Auto-save on Cinnamon shutdown:

```
echo "session-cleanup-script = /home/$USER/.local/bin/nemo_save.sh" | sudo
tee -a /etc/lightdm/lightdm.conf.d/70-linuxmint.conf
```

Restore session on Cinnamon start:

```
cat > ~/.config/autostart/nemo_restore.desktop << EOD
[Desktop Entry]
Version=1.0
Type=Application
Encoding=UTF-8
Name=Restore Nemo Session
Exec=/home/$USER/.local/bin/nemo_restore.sh
Terminal=false
StartupNotify=false
Categories=Application;
X-GNOME-Autostart-enabled=true
X-GNOME-Autostart-Delay=10
EOD
```

## Action Scripts

[https://github.com/smurphos/nemo\\_actions\\_and\\_cinnamon\\_scripts](https://github.com/smurphos/nemo_actions_and_cinnamon_scripts)

Attach file to email Thunderbird:

```
cat > ~/.local/share/nemo/actions/attach_to_email_tbird.nemo_action << "EOD"
[Nemo Action]
Name=_Attach to email
Comment=Attach %F to email in Thunderbird
Exec=thunderbird -compose to=,"attachment='%F'"
Icon-Name=mail-attachment-symbolic
Selection=notnone
Extensions=nodirs
Separator=,
Dependencies=thunderbird;
EOD
```

Attach file to email general:

```
cat > ~/.local/share/nemo/actions/attach_to_email_xdg-email.nemo_action <<
"EOD"
[Nemo Action]
Name=_Attach to email
Comment=Attach %F to email in default mail application
Exec=<action_scripts/attach_to_email_xdg-email.sh %F>
Icon-Name=mail-attachment-symbolic
Selection=notnone
Extensions=nodirs
Quote=double
Dependencies=xdg-email;
EOD

mkdir -p ~/.local/share/nemo/actions/action_scripts
```

```
cat > ~/.local/share/nemo/actions/action_scripts/attach_to_email_xdg-  
email.sh << "EOD"  
#!/bin/bash  
COMMAND="xdg-email"  
for file in "$@"  
do  
    COMMAND="$COMMAND --attach '$file'"  
done  
sh -c "$COMMAND"  
EOD
```

Edit with xed as superuser:

```
cat > ~/.local/share/nemo/actions/xed_sudoedit.nemo_action << "EOD"  
[Nemo Action]  
  
Name=_Xed (Sudoedit)  
Comment=Open %F with xed opening as different user when needed via sudoedit.  
Exec=<action_scripts/xed_sudoedit_check.sh %F>  
Icon-Name=text-editor-symbolic  
#Stock-Id=  
Selection=s  
#Extensions=  
Mimetypes=text/plain;text/css;text/english;text/html;text/x-apt-sources-  
list;text/x-c;text/x-c++;text/x-c++hdr;text/x-c++src;text/x-chdr;text/x-  
csrc;text/x-diff;text/x-dsrc;text/x-java;text/x-makefile;text/x-moc;text/x-  
pascal;text/x-perl;text/x-python;text/x-sql;text/x-tcl;text/x-tex;  
#Separator=  
#Quote=  
Dependencies=sudoedit;xed;  
#Conditions=  
EscapeSpaces=true  
Terminal=false  
EOD
```

```
mkdir -p ~/.local/share/nemo/actions/action_scripts
```

```
cat > ~/.local/share/nemo/actions/action_scripts/xed_sudoedit_check.sh <<  
"EOD"  
#!/bin/bash  
FILEOWNER=$(stat -c %U "$1")  
if [ "$FILEOWNER" = "$USER" ]; then  
    xed "$1"  
else  
    export SUDO_EDITOR="xed -w"  
    export  
SUDO_ASKPASS="$HOME/.local/share/nemo/actions/action_scripts/zenity_askpass.  
sh"  
    sudoedit -A -u "$FILEOWNER" "$1"  
fi  
EOD
```

```
sudo apt-get install zenity
```

```
cat > ~/.local/share/nemo/actions/action_scripts/zenity_askpass.sh << "EOD"
#!/bin/bash
zenity --password --title="Authenticate"
EOD
```

Edit with vi as superuser:

```
cat > ~/.local/share/nemo/actions/vi_sudoedit.nemo_action << "EOD"
[Nemo Action]

Name=_vi (Sudoedit)
Comment=Open %F with vi opening as different user when needed via sudoedit.
Exec=<action_scripts/vi_sudoedit_check.sh %F>
Icon-Name=text-editor-symbolic
#Stock-Id=
Selection=s
#Extensions=
Mimetypes=text/plain;text/css;text/english;text/html;text/x-apt-sources-
list;text/x-c;text/x-c++;text/x-c++hdr;text/x-c++src;text/x-chdr;text/x-
csrc;text/x-diff;text/x-dsrc;text/x-java;text/x-makefile;text/x-moc;text/x-
pascal;text/x-perl;text/x-python;text/x-sql;text/x-tcl;text/x-tex;
#Separator=
#Quote=
Dependencies=sudoedit;vi;
#Conditions=
EscapeSpaces=true
Terminal=false
EOD
```

```
mkdir -p ~/.local/share/nemo/actions/action_scripts
```

```
cat > ~/.local/share/nemo/actions/action_scripts/vi_sudoedit_check.sh <<
"EOD"
#!/bin/bash
FILEOWNER=$(stat -c %U "$1")
if [ "$FILEOWNER" = "$USER" ]; then
    vi "$1"
else
    export SUDO_EDITOR="vi"
    sudoedit -A -u "$FILEOWNER" "$1"
fi
EOD
```

Show dpkg package of file or folder:

```
cat > ~/.local/share/nemo/actions/show_parent_package.nemo_action << "EOD"
[Nemo Action]
Name=Show _Parent Package
Comment=Display a dialogue that shows which package %F's is from
```

```
Exec=<action_scripts/show_parent_package.sh %F>
Icon-Name=package-x-generic-symbolic
Selection=s
Extensions=any;
EscapeSpaces=true
Dependencies=zenity;dpkg;
EOD

mkdir -p ~/.local/share/nemo/actions/action_scripts
sudo apt-get install zenity

cat > ~/.local/share/nemo/actions/action_scripts/show_parent_package.sh <<
"EOD"
#!/bin/dash
OUTPUT=$(dpkg -S "$1" | cut -f1 -d:)
if [ -z "$OUTPUT" ]; then
    zenity --info --ellipsize --text="$1\nis not provided by any known
package"
else
    zenity --info --ellipsize --text="$1\nis from the package(s) $OUTPUT"
fi
EOD
```

From:  
<http://wuff.dyndns.org/> - **Wulf's Various Things**

Permanent link:  
<http://wuff.dyndns.org/doku.php?id=config:nemo-performance&rev=1679960054>

Last update: **2023/05/29 11:53**

