Optimization

```
# JPEG/JFIF - lossless
jpegoptim "${filepath}"

# PNG (and ICO that is PNG) - lossless
# zopfli makes the entire thing MUCH slower, but saves more space - question is whether it's
worth it inflating compress time by 80x (decompress is the same)
# --strip all or --strip safe options may be of interest if EXIF cleanup is desired, but
EXIFtool should suffice for that
# --preserve is of interest if saving timestamp is necessary, and --fast is necessary for --
zopfli unless you have a lot of CPU or a lot of time
oxipng --zopfli --opt max "${filepath}"

# SVG - might lose data that other scripts may rely on if the SVG is somehow parsed
#   see `plasma/breeze:.../build.sh` where name metadata is fed to inkscape
# Verification that result is smaller is necessary as svgo may actually be
#  adding data necessary for correct rendering everywhere.
#  That may or may not be desirable, bad for filesize, good for compatibility.
# Corrupt output is possible, take care
svgo --multipass --input "${filepath}" --output "${tempDir}/svgo-$$.temp"

# BMP -> PNG - lossless, but probably requires editing other things that link to BMP
convert "${filepath}" "${filepathWithoutExtension}.png"

# PDF - lossless recompression
# Corrupt output is possible, take care
pdftk "${filepath}" output "${filepath}.tmp" uncompress
pdftk "${filepath}.tmp" output "${filepath}.packie" compress

# GIF - lossless optimization
gifsicle "${filepath}" --output "${tempDir}/gifsicle-$$.temp"

# TTF - Converting to ttx and back with fonttools can gain a little bit of savings
fonttools ttx -q -o "${tempDir}/fonttools-$$.ttx" "${filepath}"
fonttools ttx -q -o "${tempDir}/fonttools-$$.ttf" "${tempDir}/fonttools-$$.ttx"
```

```
# EXIF in PNG/JPEG/JFIF
# Delete ALL EXIF data, might be dangerous and remove useful things like color profiles
exiftool -v -preserve -overwrite_original -All= "${filepath}"
# Delete old hostcomputer watermark, anything related to thumbnails and Software used for creation
exiftool -v -preserve -overwrite_original -HostComputer= -ifd1:all= -Software= "${filepath}"
```

A lot of files scattered around the internet have pointlessly large file sizes, which modern software can optimize out, or they weren't meant to be there in the first place.

The above commands try to do so without losing any data. Further file size savings can be achieved by means of `exiftool` and removing extra unnecessary EXIF information, but that is entering lossy territory, as it can often carry useful or necessary information.

Revision #11
Created 2024-03-25 15:58:56 UTC by C0rn3j
Updated 2024-04-15 22:33:34 UTC by C0rn3j