image, imagemagick, code, shell, em

Imagemagick

Imagemagick est un logiciel en ligne de commande pour la manipulation de fichiers images. Il fait partie des logiciels “couteaux suisses” de manipulation de fichiers multimédia (avec ffmpeg et sox). Imagemagick est utile pour :

- redimensionner une image
- changer de format
- assembler des images pour en faire des animations ou des fichiers pdf
- composer des images
- etc.

Ce logiciel porte bien son nom, arriver à s'en servir correctement tient parfois de la magie noire…

Imagemegick se compose de plusieurs commandes indépendantes : **convert**, **identify**, **mogrify**, **composite**, **montage**, **display**, **animate**, **compare**, **stream**, **import**
Avec la version 7 d'Imagemagick, **magick** est ajoutée pour l'ensemble des commandes précédentes (magick convert, magick composite, etc.).

Animation

Voir Création de gifs animés avec Imagemagick

Composition d'images

Voir composition d'images avec imagemagick

Imposition

Voir les scripts de ce repo pour différents types d'imposition : https://github.com/emoc/edition_imposition/

Monter plusieurs exemplaires du même document
A partir d'un original en A4, en mettre deux exemplaires en A5 sur le même feuillet

```
# 20211113 / ImageMagick 6.9.7-4 Q16 x86_64 20170114 / Debian Stretch 9.5 @ Kirin
convert -density 300 ./original.pdf original.png
montage -geometry 1748x2480 -tile 2x1 original.png original.png 2xoriginal.png
convert ./2xoriginal.png -density 300 -units pixelsperinch -compress jpeg -quality 80% 2xoriginal.pdf
```

Traitement par lot

La commande **mogrify** permet de traiter toutes les images d'un répertoire. (cf : http://www.imagemagick.org/Usage/basics/#mogrify)
Si mogrify ne fait pas l'affaire, il est aussi possible d'utiliser **convert**, dans ce cas il vaut mieux passer par un script qui traitera les images une par une, car convert charge toutes les images en mémoire avant de les transformer.

Exemples

Réduire toutes les images d'un répertoire

```
# 20200510 / ImageMagick 6.9.7-4 Q16 x86_64 20170114 / Debian Stretch 9.5 @ Kirin
mogrify -path small -resize 16x24! *.png
```

Toutes les images au format png seront réduites à exactement 16×24 pixels et enregistrées dans le sous-dossier “small” (à créer au préalable)

Variante, en pourcentage avec adaptation de la qualité du JPEG (recherche poussée sur le sujet)

```
# 20220511 / ImageMagick 6.9.7-4 Q16 x86_64 20170114 / Debian Stretch 9.5 @ Kirin
mogrify -resize 50% -quality 82 *.JPG
```

Agrandir toutes les images d'un répertoire sans interpolation

```
# 20210525 / ImageMagick 6.9.7-4 Q16 x86_64 20170114 / Debian Stretch 9.5 @ Kirin
mogrify -resize 200% -filter point *.png
```

Utile pour agrandir du pixel art sans aliasing

Rotation de toutes les images correspondant à un certain motif

```
# 20220308 / ImageMagick 6.9.7-4 Q16 x86_64 20170114 / Debian Stretch 9.5 @ Kirin
mogrify -rotate 90 *td.jpg       # tous les fichiers finissant par td.jpg (tructd.jpg, machin_td.jpg)
```

90 pour 1/4 de tour dans le sens des aiguilles d'une montre (dextrogyre!)

Passer toutes les images PNG en JPG

```
# 202204017 / ImageMagick 6.9.7-4 Q16 x86_64 20170114 / Debian Stretch 9.5 @ Kirin
mogrify -format jpg -quality 85 *.png
```

Les images PNG originales ne seront pas effacées

Convertir des images WEBP en JPEG vers un sous-dossier

```
# 20220512 / ImageMagick 6.9.7-4 Q16 x86_64 20170114 / Debian Stretch 9.5 @ Kirin
mkdir ./conversion     # créer le dossier si nécessaire
mogrify -format JPEG -path conversion *.webp # enregistrer le résultat dans le dossier conversion
```

Redimensionnement d'images

Réduire en fixant une dimension

```
# 20220322 / ImageMagick 6.9.7-4 Q16 x86_64 20170114 / Debian Stretch 9.5 @ Kirin
convert in.jpg -resize 400x -quality 85% out.jpg
```

Toutes les images seront converties pour avoir une largeur de 400 pixels

Redimensionner à un format fixe

```
# 20220322 / ImageMagick 6.9.7-4 Q16 x86_64 20170114 / Debian Stretch 9.5 @ Kirin
convert *.jpg -resize 1200x900^ -gravity center -extent 1200x900  test_%d.jpg
```

Toutes les images JPEG seront converties à 1200×900, en agrandissant la plus petite dimension sans changer les proportions : l'image remplit toute la surface demandée et peut la dépasser, dans ce cas elle est tronquée. (nb : pas de *wildcard* dans le nom de fichier en sortie).

Assemblages d'images

Assembler deux images

```
# 20210824 / ImageMagick 6.9.7-4 Q16 x86_64 20170114 / Debian Stretch 9.5 @ Kirin
# deux images de 877x1240px assemblées en un A4 paysage @ 150dpi
convert -page +0+0 gauche.png -page +878+0 droite.png -mosaic compo.png
```

Composer une planche contact

```
# 20210913 / ImageMagick 6.9.7-4 Q16 x86_64 20170114 / Debian Stretch 9.5 @ Kirin
# une image est composée à partir d'une série d'images à taille réduite
montage 001.tif 002.tif 003.tif 004.tif 005.tif 006.tif 007.tif 008.tif -tile 2x4 -geometry 150x225 montage.jpg
```

Variante en gardant le fond transparent.

```
# 20220212 / ImageMagick 6.9.7-4 Q16 x86_64 20170114 / Debian Stretch 9.5 @ Kirin
montage p1.png p2.png p3.png p4.png -tile 4x -background none -geometry 32x24 PNG32:montage.png
```

Miniatures de tout un répertoire

```
# 20210913 / ImageMagick 6.9.7-4 Q16 x86_64 20170114 / Debian Stretch 9.5 @ Kirin
# [0] permet d'utiliser uniquement la 1re image de fichiers multi images (gif animé, par exemple)
montage *[0] montage.jpg
```

Montage d'une liste de fichiers

Aller chercher tous les fichiers d'un certain type depuis les sous-répertoires, les copier dans un répertoire et en faire une planche contact

```
# 20210913 / ImageMagick 6.9.7-4 Q16 x86_64 20170114 / Debian Stretch 9.5 @ Kirin
find ./ -name '*.png' -exec cp -prv '{}' '/home/emoc/fichiers_a_traiter/' ';'
ls *.png > files.txt
montage @files.txt -geometry 800x800+0+0 -tile 6x6 -background none montage.png
```

Petits trucs pratiques

Ajouter une bordure de couleur à une image (source)

```
# 20200515 / ImageMagick 6.9.7-4 Q16 x86_64 20170114 / Debian Stretch 9.5 @ Kirin
convert image.png -shave 1x1 -bordercolor black -border 1 image_avec_bordure.png
# mogrify -shave 1x1 -bordercolor black -border 1 -format png *.png # utile pour le faire sur tout un répertoire
```

Conversion d'un fichier .pdf en une série d'images

```
# 20200705 / ImageMagick 6.9.7-4 Q16 x86_64 20170114 / Debian Stretch 9.5 @ Kirin
convert -density 150 fichier.pdf image_%02d.png
```

Conversion d'une série d'images en pdf

```
# 20210604 / ImageMagick 6.9.7-4 Q16 x86_64 20170114 / Debian Stretch 9.5 @ Kirin
# Les images ont une définition identique + attention à l'ordre des arguments : placer les fichiers bitmap avant density et units
convert ./recto.png ./verso.png -density 300 -units pixelsperinch document.pdf
```

Conversion pdf avec images compressées en jpeg

```
# 20210913 / ImageMagick 6.9.7-4 Q16 x86_64 20170114 / Debian Stretch 9.5 @ Kirin
convert -density 100 -compress jpeg -quality 70% image.jpg document.pdf
```

Pour convertir toutes les images d'un répertoire, on peut utiliser le joker * :

```
# 20210913 / ImageMagick 6.9.7-4 Q16 x86_64 20170114 / Debian Stretch 9.5 @ Kirin
convert -density 150 -compress jpeg -quality 80% * file.pdf
```

Ou les nommer une par une

```
convert -density 150 -compress jpeg -quality 80% image_1.jpg image_2.jpg image_3.jpg file.pdf
```

Ou utiliser le «*joker*» *

```
convert -density 150 -compress jpeg -quality 80% image_*.jpg file.pdf
```

Réduire la taille d'un fichier pdf

Adapter *density* et *quality* selon les besoins

```
# 20210927 / ImageMagick 6.9.7-4 Q16 x86_64 20170114 / Debian Stretch 9.5 @ Kirin
convert -density 150 ./original.pdf -compress jpeg -quality 80% reduit.pdf
```

Créer des pages factices

Utile pour tester des modèles PAO, la boucle permet de définir le nombre de pages

```
#!/bin/bash
# 20210913 / ImageMagick 6.9.7-4 Q16 x86_64 20170114 / Debian Stretch 9.5 @ Kirin
for i in $(seq -f "%02g" 1 16)
do
  (convert -background white -fill black -size 877x1240 -gravity center label:"$i" page_"$i".png)
  # version alternative avec un bord noir de 1 pixel
  # (convert -background white -fill black -size 877x1240 -gravity center label:"$i" -shave 1x1 -bordercolor black -border 1 page_"$i".png)
done
```

Imposition

4 flyers sur la même page A4

D'un fichier pdf exporté depuis Scribus à un fichier pdf comprenant 4 exemplaires montés pour impression (pour imprimer des vero de flyers)

```
#!/bin/bash

# debian 9.5 Stretch + LXDE @ kirin 2022725
# Scribus 1.4.6 / ImageMagick 6.9.7-4 Q16 x86_64 20170114 / pdfjam 2.0.8

# prendre le fichier flyer-verso.pdf exporté depuis scribus (A5 :/ )
# le transformer en fichier png @ 300dpi
convert -density 300 flyer-verso.pdf flyer-verso.png

# le monter sous forme 2x2 pour imprimer sur un A4
montage flyer-verso.png flyer-verso.png flyer-verso.png flyer-verso.png -tile 2x2 -geometry 1748x2480 flyer_montage.png

# transformer en pdf
convert flyer_montage.png -density 300 -units pixelsperinch flyer_montage.pdf

# adapter le format en A4 avec pdfjam
pdfjam --paper a4paper --outfile flyer_montage_A4.pdf flyer_montage.pdf

# effacer les fichiers temporaires
rm flyer_montage.png
rm flyer_montage.pdf
```

Ressources

- documentation en détail de la version 6 http://www.imagemagick.org/Usage/
- traduction de la documentation version 6.2 par Yves Jacolin (2007) : http://softlibre.gloobe.org/doku.php?id=imagemagick:doc_start
- https://imagemagick.org/script/command-line-tools.php
- https://doc.ubuntu-fr.org/imagemagick
- exemples détaillés d'usages avancés (scripts bat pour windows) https://im.snibgo.com/index.htm
- autres exemples détaillés d'usages avancés (scripts shell pour linux) http://www.fmwconcepts.com/imagemagick/index.php

Article extrait de : http://lesporteslogiques.net/wiki/ - **WIKI Les Portes Logiques**
Adresse : **http://lesporteslogiques.net/wiki/ressource/logiciel/imagemagick/start?rev=1658847125**
Article mis à jour: **2022/07/26 16:52**