



ImageMagick アレコレ


2015/02/18

“よや” yoya@awm.jp

もくじ

- 自己紹介
- ImageMagick って何？
- ImageMagick の構造
- 色々なバージョンのビルド
- よく知られた負荷対策
- 最近(?)のトピック
- その他

自己紹介

- 一年すこし前まで、この場所の会社にいました
 - 画像サーバの御守りがメイン 
 - 容量が溢れる前に別HDDに rsync するタンポポ仕事
 - 今は高度に自動化されてるはず！
- ImageMagick のコミット実績
 - 小さな不具合修正2つと機能改善1つ only
- ImageMagick のストーキングが趣味
 - バージョンアップの度に差分を見てます
 - <http://d.hatena.ne.jp/yoya/searchdiary?word=ImageMagick>

はじめにことわり

- ImageMagick-6 前提の発表だけれど、本家 trunk はとっくに ImageMagick-7
- 6 はそのバックポートで運用
- 話が少し古いかもしれませんが
- しかし、新機能は 6,7 同時に入っているので、今の所 6 でも問題ないはず

ImageMagick-6 と 7 の違い

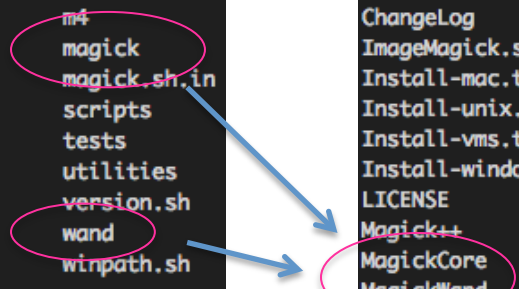
- 誰か面白い違いを知ってたら教えて下さい
- 自分は API 名に合わせてフォルダ名が2箇所変わった事しか知りません

ImageMagick-6

ImageMagick-7

```
pro$ ls
AUTHORS.txt      NOTICE          index.html
ChangeLog        PerlMagick       m4
ImageMagick.spec.in  Platforms.txt   magick
Install-mac.txt  QuickStart.txt  magick.sh.in
Install-unix.txt  README.txt      scripts
Install-vms.txt  aclocal.m4      tests
Install-windows.txt  coders         utilities
LICENSE          common.shi.in   version.sh
Magick++         config          wand
Magickshr.opt    configure       winpath.sh
Makefile.am      configure.ac     www
Makefile.in      filters
NEWS.txt         images
```

```
pro$ ls
AUTHORS.txt      Makefile.in      configure.ac
ChangeLog        NEWS.txt         filters
ImageMagick.spec.in  NOTICE         images
Install-mac.txt  PerlMagick       index.html
Install-unix.txt  Platforms.txt    m4
Install-vms.txt  QuickStart.txt  magick.sh.in
Install-windows.txt  README.txt     scripts
LICENSE          aclocal.m4      tests
Magick++         api_examples    utilities
MagickCore       coders          version.sh
MagickWand       common.shi.in   winpath.sh
Magickshr.opt    config          www
Makefile.am      configure
```

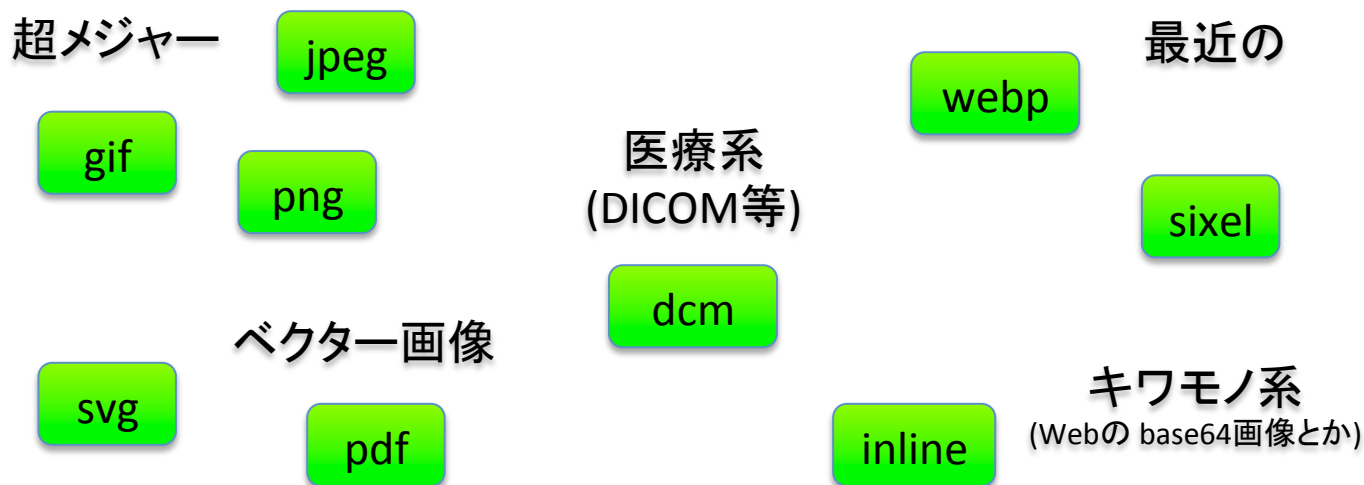


ImageMagick って何？

- 画像を処理出来る何か
 - (機能はコマンドのオプションで大体分かる)
- コマンドが用意されてる
 - リサイズとかフィルタの画像変換が出来る (convert)
 - 複数の画像をまとめて GIFアニメが作れる
 - 画像の情報を文字列で見られる (identify)
- 色々な言語から使える
 - Java, Ruby, Python, Perl, .NET, PHP 等。。
 - <http://imagemagick.org/script/api.php>

対応フォーマット沢山

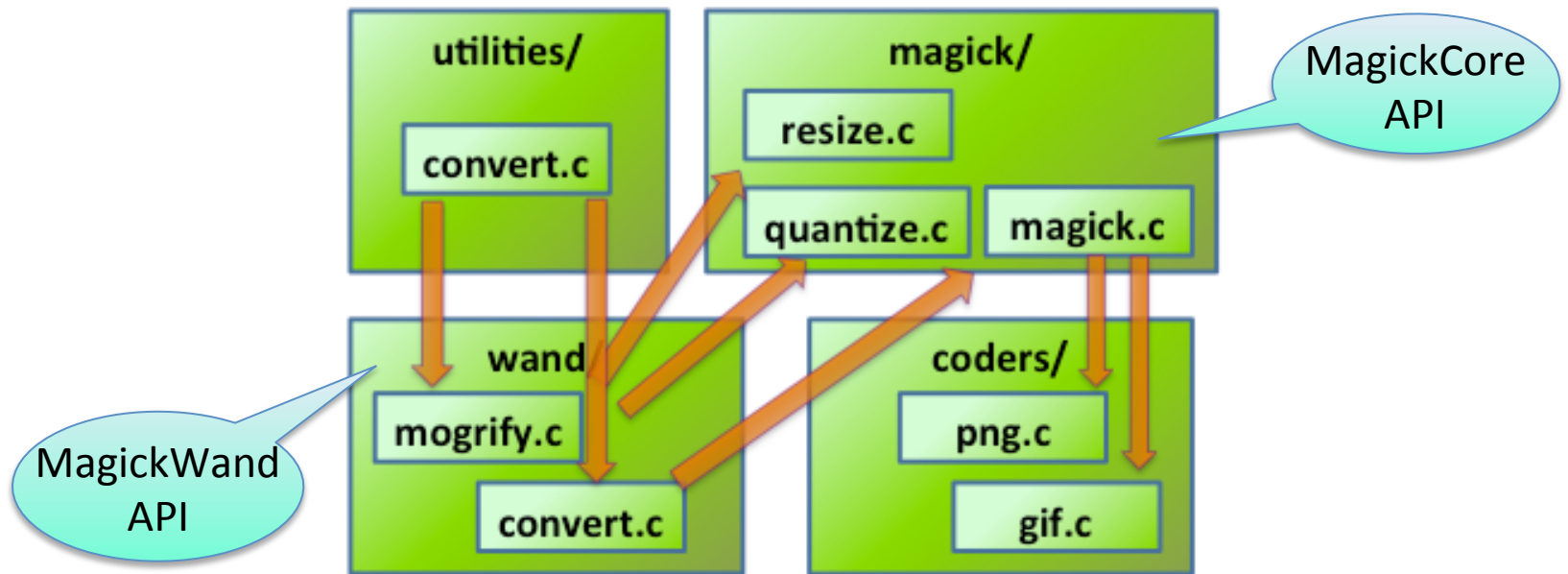
- メジャーな画像フォーマットからマイナーなものまで100種類以上に対応してる



- <http://www.imagemagick.org/script/formats.php>

で、ImageMagick って何なのさ？

- 分らないので1コマンドの処理を追って見る

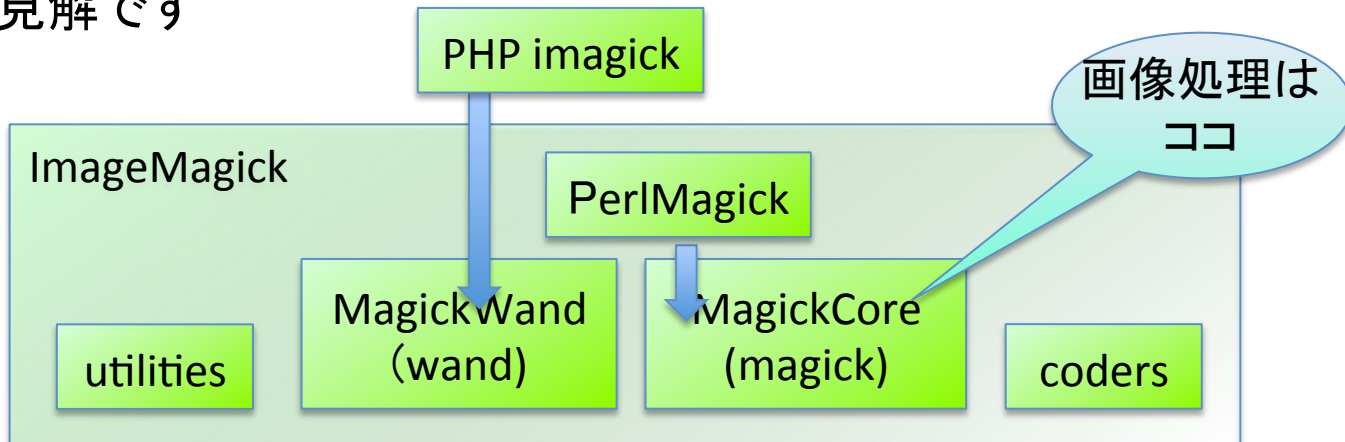


- 引用元) <http://labs.gree.jp/blog/2012/07/2446/>

ImageMagick とは？

- MagickCore というエンジンと、プラグイン的な coder 集合体と、ユーティリティ的な MagickWand API と、そこに結びつくコマンドラインや言語バインディングの仕組み

※個人的見解です



ImageMagick の開発傾向

- 活発(良くも悪くも)
 - 新しい機能や画像形式をすぐ取り込んでリリース
 - よく細かいデグレードするけど、それもすぐ治る
 - デグレを治したつもりで更にそのデグレが入ったりするけど、めげずにすぐ治す
 - APIのバイナリ互換とかあまり気にしない。(最近大人しい気がする)
 - ちなみに、このやり方にキレてフォークしたのが **GraphicsMagick**
- (個人的には)素晴らしい。。。
 - が、サービス導入する場合、この傾向を頭に入れる必要がある

バージョン間の差異の把握

- 変換の性質がバージョンによって結構変わる

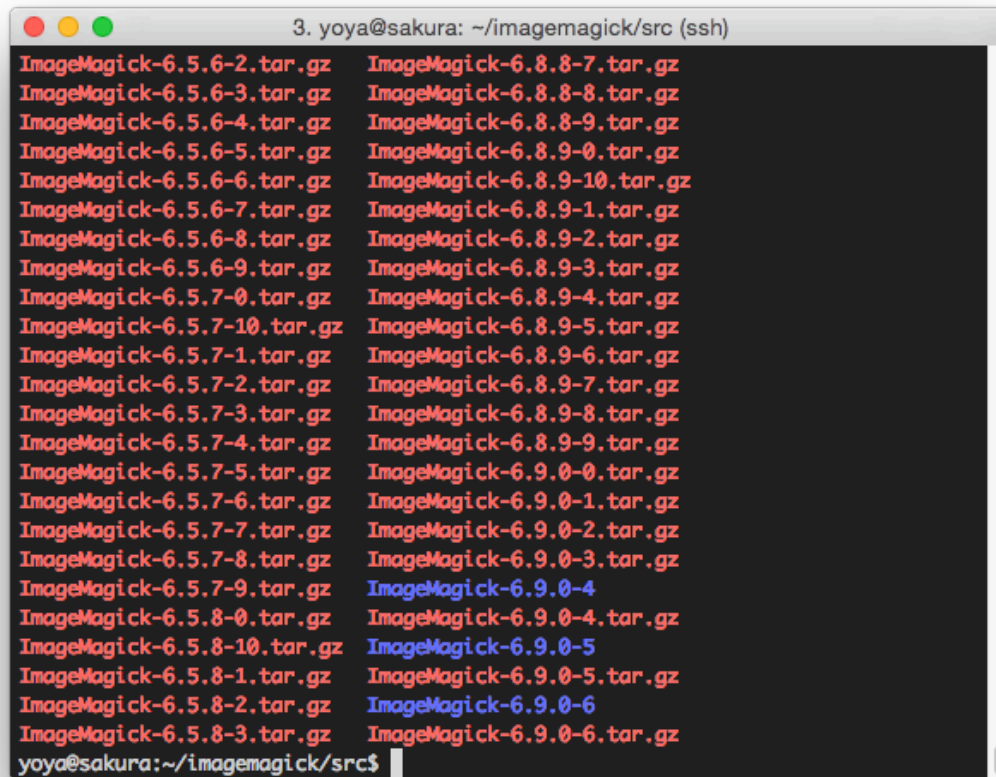
– 減色結果の比較



- 引用元) <http://labs.gree.jp/blog/2013/12/9290/>

各バージョンでの動作を知りたい

- 持ってるバージョンの tar.gz 全て並べる



```
3. yoya@sakura: ~/imagemagick/src (ssh)
ImageMagick-6.5.6-2.tar.gz  ImageMagick-6.8.8-7.tar.gz
ImageMagick-6.5.6-3.tar.gz  ImageMagick-6.8.8-8.tar.gz
ImageMagick-6.5.6-4.tar.gz  ImageMagick-6.8.8-9.tar.gz
ImageMagick-6.5.6-5.tar.gz  ImageMagick-6.8.9-0.tar.gz
ImageMagick-6.5.6-6.tar.gz  ImageMagick-6.8.9-10.tar.gz
ImageMagick-6.5.6-7.tar.gz  ImageMagick-6.8.9-1.tar.gz
ImageMagick-6.5.6-8.tar.gz  ImageMagick-6.8.9-2.tar.gz
ImageMagick-6.5.6-9.tar.gz  ImageMagick-6.8.9-3.tar.gz
ImageMagick-6.5.7-0.tar.gz  ImageMagick-6.8.9-4.tar.gz
ImageMagick-6.5.7-10.tar.gz ImageMagick-6.8.9-5.tar.gz
ImageMagick-6.5.7-1.tar.gz  ImageMagick-6.8.9-6.tar.gz
ImageMagick-6.5.7-2.tar.gz  ImageMagick-6.8.9-7.tar.gz
ImageMagick-6.5.7-3.tar.gz  ImageMagick-6.8.9-8.tar.gz
ImageMagick-6.5.7-4.tar.gz  ImageMagick-6.8.9-9.tar.gz
ImageMagick-6.5.7-5.tar.gz  ImageMagick-6.9.0-0.tar.gz
ImageMagick-6.5.7-6.tar.gz  ImageMagick-6.9.0-1.tar.gz
ImageMagick-6.5.7-7.tar.gz  ImageMagick-6.9.0-2.tar.gz
ImageMagick-6.5.7-8.tar.gz  ImageMagick-6.9.0-3.tar.gz
ImageMagick-6.5.7-9.tar.gz  ImageMagick-6.9.0-4
ImageMagick-6.5.8-0.tar.gz  ImageMagick-6.9.0-4.tar.gz
ImageMagick-6.5.8-10.tar.gz ImageMagick-6.9.0-5
ImageMagick-6.5.8-1.tar.gz  ImageMagick-6.9.0-5.tar.gz
ImageMagick-6.5.8-2.tar.gz  ImageMagick-6.9.0-6
ImageMagick-6.5.8-3.tar.gz  ImageMagick-6.9.0-6.tar.gz
yoya@sakura:~/imagemagick/src$
```

バッチで全バージョンをビルド

```
#!/bin/sh
for file in `ls -r ImageMagick*.tar.*` ; do
  version=`echo $file | sed 's/ImageMagick-¥(.*)\.tar¥(.*)/¥1/'`
  version2=`echo $file | sed 's/ImageMagick-¥(.*)-[0-9]¥+.tar¥(.*)/¥1/'`
  if [ "$pre_version" != "$version" ] ; then
    echo === $file ===
    tar xf $file
    dir="ImageMagick-$version"
    if ! [ -d $dir ] ; then
      dir="ImageMagick-$version2";
    fi
    if [ -d $dir ] ; then
      (cd $dir ; ./configure --without-perl --without-magick-plus-plus ¥
        --prefix=$HOME/ImageMagick/$version ; make install)
      rm -rf $dir
    fi
    pre_version=$version
  fi
done
```

- 引用元) <http://labs.gree.jp/blog/2013/12/9290/>

ビルドについて

- Debian だと apt-get build-dep ImageMagick で必要なライブラリが入るので、後は configure と make をするだけ
- Zlib が 1.2.6 だと ImageMagick-6.7.5-0 以降しかコンパイル出来ないなので、それより古いバージョンをビルドする場合は zlib 1.2.5 以下が必要
- 昔のバージョンは PerlMagick の pm を PREFIX を無視して /usr/lib 辺りに入れてたので、--without-perl した方が良い
- その他、細かい対応はこちらのブログを参考
 - <http://d.hatena.ne.jp/yoya/20121020/imagemagick>
 - <http://d.hatena.ne.jp/yoya/20121021/imagemagick>

全バージョンをビルドした結果

```
3. yoya@sakura: ~/ImageMagick (ssh)
yoya@sakura:~$
yoya@sakura:~$ cd ImageMagick/
yoya@sakura:~/ImageMagick$ ls -l | wc
632 5681 30967
yoya@sakura:~/ImageMagick$ ls | tail
6.8.9-7
6.8.9-8
6.8.9-9
6.9.0-0
6.9.0-1
6.9.0-2
6.9.0-3
6.9.0-4
6.9.0-5
6.9.0-6
yoya@sakura:~/ImageMagick$ ls 6.9.0-6
bin etc include lib share
yoya@sakura:~/ImageMagick$ ls 6.9.0-6/bin
animate conjure identify MagickCore-config montage
compare convert import MagickWand-config stream
composite display Magick-config mogrify Wand-config
yoya@sakura:~/ImageMagick$
```

```
3. yoya@sakura: ~/ImageMagick (ssh)
6.3.7-4 6.4.4-5 6.5.2-2 6.5.9-4 6.6.6-6 6.7.4-0 6.8.1-2 6.8.8-5
6.3.7-5 6.4.4-6 6.5.2-3 6.5.9-5 6.6.6-7 6.7.4-1 6.8.1-3 6.8.8-6
6.3.7-6 6.4.4-7 6.5.2-4 6.5.9-6 6.6.6-8 6.7.4-10 6.8.1-4 6.8.8-7
6.3.7-7 6.4.4-8 6.5.2-5 6.5.9-7 6.6.6-9 6.7.4-2 6.8.1-5 6.8.8-8
6.3.7-8 6.4.4-9 6.5.2-6 6.5.9-8 6.6.7-0 6.7.4-3 6.8.1-6 6.8.8-9
6.3.7-9 6.4.5-0 6.5.2-7 6.5.9-9 6.6.7-1 6.7.4-4 6.8.1-7 6.8.9-0
6.3.8-0 6.4.5-1 6.5.2-8 6.6.0-0 6.6.7-10 6.7.4-5 6.8.1-8 6.8.9-1
6.3.8-1 6.4.5-2 6.5.2-9 6.6.0-1 6.6.7-2 6.7.4-6 6.8.1-9 6.8.9-10
6.3.8-10 6.4.5-3 6.5.3-0 6.6.0-10 6.6.7-3 6.7.4-7 6.8.2-1 6.8.9-2
6.3.8-11 6.4.5-4 6.5.3-1 6.6.0-2 6.6.7-4 6.7.4-8 6.8.2-10 6.8.9-3
6.3.8-2 6.4.5-5 6.5.3-10 6.6.0-3 6.6.7-5 6.7.4-9 6.8.2-2 6.8.9-4
6.3.8-3 6.4.5-6 6.5.3-2 6.6.0-4 6.6.7-6 6.7.5-0 6.8.2-3 6.8.9-5
6.3.8-4 6.4.5-7 6.5.3-3 6.6.0-5 6.6.7-7 6.7.5-1 6.8.2-4 6.8.9-6
6.3.8-5 6.4.5-8 6.5.3-4 6.6.0-6 6.6.7-8 6.7.5-10 6.8.2-5 6.8.9-7
6.3.8-6 6.4.5-9 6.5.3-5 6.6.0-7 6.6.7-9 6.7.5-2 6.8.2-6 6.8.9-8
6.3.8-7 6.4.6-0 6.5.3-6 6.6.0-8 6.6.8-0 6.7.5-3 6.8.2-7 6.8.9-9
6.3.8-8 6.4.6-1 6.5.3-7 6.6.0-9 6.6.8-1 6.7.5-4 6.8.2-8 6.9.0-0
6.3.8-9 6.4.6-2 6.5.3-8 6.6.1-0 6.6.8-10 6.7.5-5 6.8.2-9 6.9.0-1
6.3.9-0 6.4.6-3 6.5.3-9 6.6.1-1 6.6.8-2 6.7.5-6 6.8.3-0 6.9.0-2
6.3.9-1 6.4.6-4 6.5.4-0 6.6.1-10 6.6.8-3 6.7.5-7 6.8.3-1 6.9.0-3
6.3.9-10 6.4.6-5 6.5.4-1 6.6.1-2 6.6.8-4 6.7.5-8 6.8.3-10 6.9.0-4
6.3.9-2 6.4.6-6 6.5.4-10 6.6.1-3 6.6.8-5 6.7.5-9 6.8.3-2 6.9.0-5
6.3.9-3 6.4.6-7 6.5.4-2 6.6.1-4 6.6.8-6 6.7.6-0 6.8.3-3 6.9.0-6
6.3.9-4 6.4.6-8 6.5.4-3 6.6.1-5 6.6.8-7 6.7.6-1 6.8.3-4
yoya@sakura:~/ImageMagick$
```

バッチで全バージョンを実行

```
#!/bin/sh
imagemagick_dir=$HOME/ImageMagick
last_arg=`eval echo '${ $#}'`
for ver in `ls -r $imagemagick_dir`
do
  option=""
  for arg in $*
  do
    if [ "$arg" != "$last_arg" ]; then
      option="$option $arg"
    else
      option="$option $ver-$arg"
    fi
  done
  convert=$imagemagick_dir/$ver/bin/convert
  echo $convert $option
  time $convert $option
done
```

- 引用元) <http://labs.gree.jp/blog/2013/12/9290/>

全バージョンで実行した結果

6.3.0-7	6.3.1-6	6.3.2-9	6.3.3-10	6.3.4-10	6.3.5-10	6.3.6-10	6.3.7-10	6.3.8-11	6.3.9-10
6.4.0-11	6.4.1-10	6.4.2-10	6.4.3-10	6.4.4-10	6.4.5-9	6.4.6-9	6.4.7-10	6.4.8-10	6.4.9-10
6.5.0-10	6.5.1-10	6.5.2-10	6.5.3-10	6.5.4-10	6.5.5-10	6.5.6-10	6.5.7-10	6.5.8-10	6.5.9-10
6.6.0-10	6.6.1-10	6.6.2-10	6.6.3-10	6.6.4-10	6.6.5-10	6.6.6-10	6.6.7-10	6.6.8-10	6.6.9-10
6.7.0-10	6.7.1-10	6.7.2-10	6.7.3-2	6.7.4-10	6.7.5-10	6.7.6-10	6.7.7-10	6.7.8-10	6.7.9-10
6.8.0-10	6.8.1-10	6.8.2-10	6.8.3-10	6.8.4-10	6.8.5-6	6.8.6-10	6.8.7-8	none	none

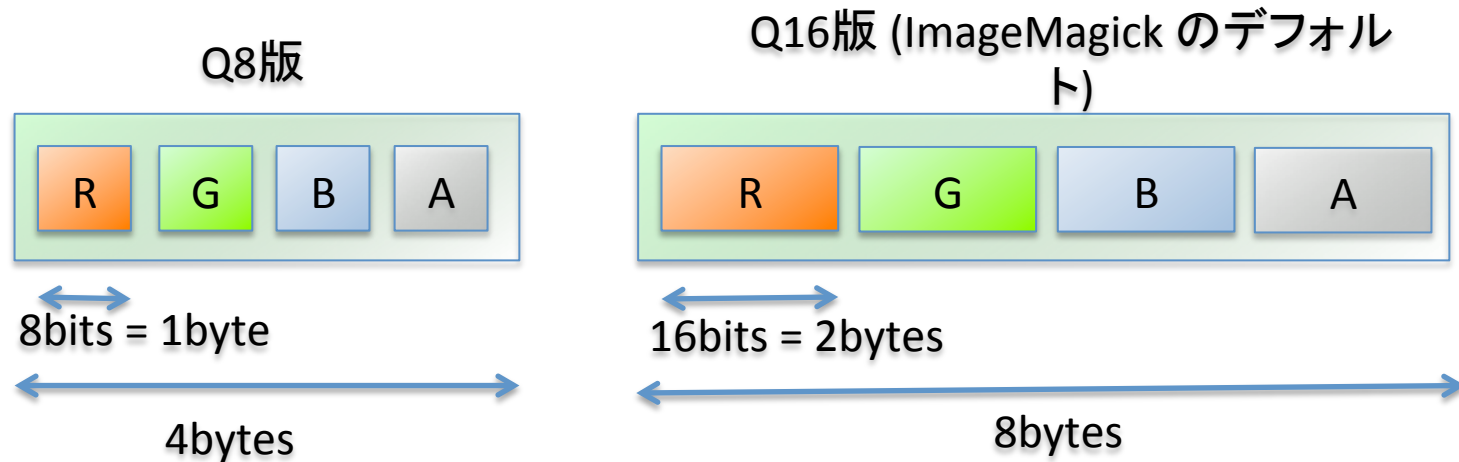
引用元) <http://labs.gree.jp/blog/2013/12/9290/>

性能改善アレコレ

- QuantumDepth
- --disable-openmp
- -define jpeg:size={width}x{height}

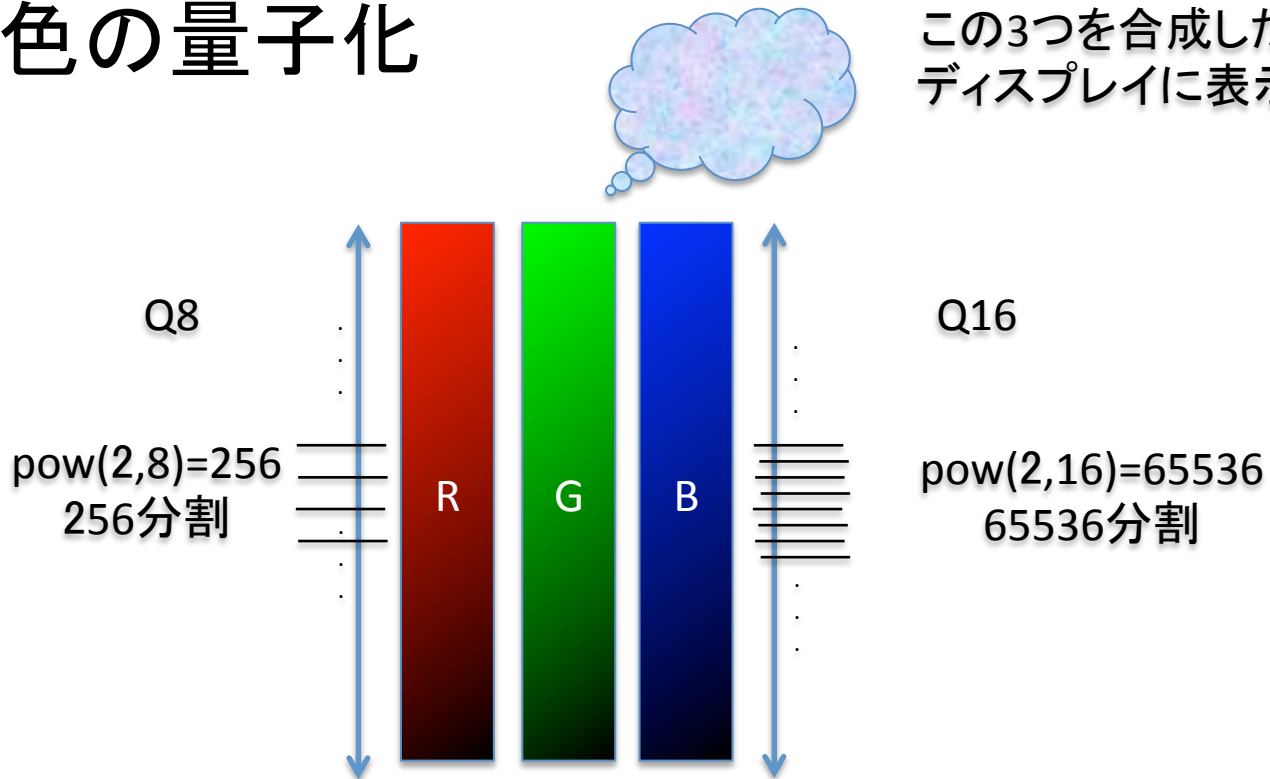
QuantumDepth 知っていますか？

- Windows 版バイナリで Q8, Q16 というアレ
- RGB の各値を 8bit、16bitどちらで持つか
- PNG,GIF,JPEG でトゥルーカラーと呼ばれるものは大体8bit



RGB色の量子化

- RGB色の量子化



- 普通の人間の目なら Q8 で十分

ImageMagick はデフォで Q16

- ImageMagick は入力した画像をメモリ上にラスター画像として展開して処理する
- Q8 版はQ16版の使用メモリ半分で済む。メモリがキツイ環境では考慮すると良いかも。(CPUも軽くなる。ハマれば3割位)

```
./configure --with-quantum-depth=8
```

- ただし、減色処理に関しては、別の格納形式を使うので、変わらない。(MagickRealType という浮動小数点の型、確か double)

Q8 の問題点

- 医療用画像(レントゲン等)は Q16らしいので使えない (Web ではこの点では問題ない)
- リサイズだと*あまり*問題にならない
- カラー効果のフィルタに通す計算で、量子化誤差が増えて気になる可能性がある。
 - 例えば擬似 HDR
 - ガンマカーブを捻じ曲げて明るい所と暗い所の明暗を強調する手法らしい
 - 本物の HDR は露出を変えて撮った写真を合成する手法

--disable-mp

- OpenMP を有効にすると過負荷時に弱くなるので、サーバでは disable を忘れずに
- ビルドし直さなくても環境変数の OMP_NUM_THREADS=1 もしくは MAGICK_THREAD_LIMIT=1 でも良い。
- 参考)
<https://twitter.com/yoya/status/409340134782275584>

-define jpeg:size={width}x{height}

- <http://blog.mirakui.com/entry/20110123/1295795409>

昼メシ物語

2011 01 23

本当は速いImageMagick: サムネイル画像生成を10倍速くする方法 imagemagick

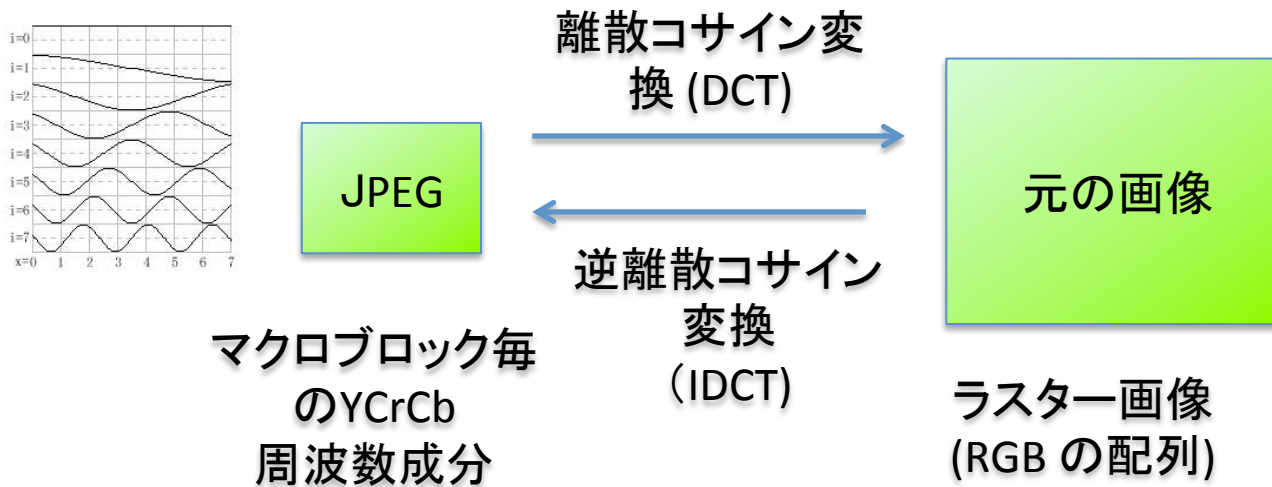
一般的に [ImageMagick](#) のサムネイル画像生成は遅いとされており、パフォーマンスが求められるシーンでは [Imlib2](#) などのより高速な画像処理ライブラリが使われることが多いです。

Imlib2 の高速さについては、以前「[Imlib2でImageMagickより3倍高速かつ美しいサムネイル画像の生成 - 床のトルストイ、ゲイとするとのこと](#)」という記事で紹介しました。この記事のベンチマークにおいて、Imlib2 によるサムネイル画像の生成は、ImageMagick の3倍程高速でした。

しかし、ImageMagick は Imlib2 より画質がよく、高機能で使いやすく、今も頻繁にメンテナンスされており、とてもよく出来ています。その点 Imlib2 は、2004年からメンテナンスされておらず、サポートが止まっています。各

JPEG画像展開

- JPEG は RGB 値をそのまま持っていない

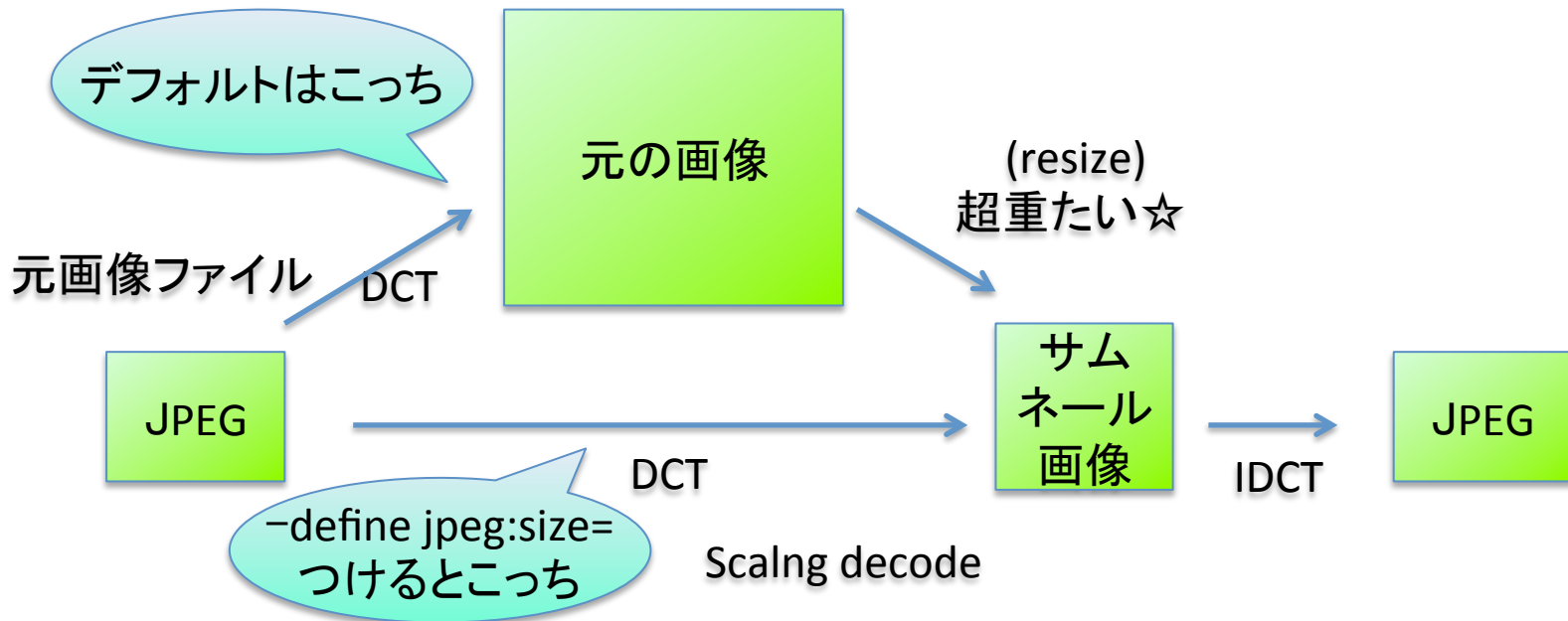


- 画像引用)

<http://webs.lanset.com/crazy17/jp/lab/jpeg2.htm>

JPEG 拡張デコード機能

- JPEGlib の拡張デコード機能が動く
 - 同じサイズでなく、指定したサイズに画像展開



最近のトピック

- ImageMagick-6.8.7-4 が個人的にエポックメイキング的なバージョンだと思っている
 - 参考) <http://d.hatena.ne.jp/yoya/20131101/im>
 - OpenCL にガチで対応し始めた (Shi Chi Chan さん)
 - 減色処理の高速化 (グリーのひと)
 - GIF アニメもちよっ速になる。事が多い。

OpenCL ガチ対応

- <http://www.imagemagick.org/script/opencl.php>

blur
charcoal
contrast
contrast-stretch
convolve
despeckle
edge
equalize
emboss
function
gaussian-blur
grayscale
modulate
motion-blur
negate
noise
radial-blur
resize
sketch
unsharp

つまり、
色んなフィルタが
速くなる

OpenCL 注意点

- 元々の C のコードとは別に実装しているので、フィルタが全く同じ挙動になる保証は無い
- 各フィルタ処理の OpenCL 版が accelerate.c に入ってる

```
pro$ ls -l magick/accelerate.*  
-rw-r--r-- 1 yoya staff 249736 12  8 00:11 magick/accelerate.c  
-rw-r--r-- 1 yoya staff   2942 11 26 02:12 magick/accelerate.h
```

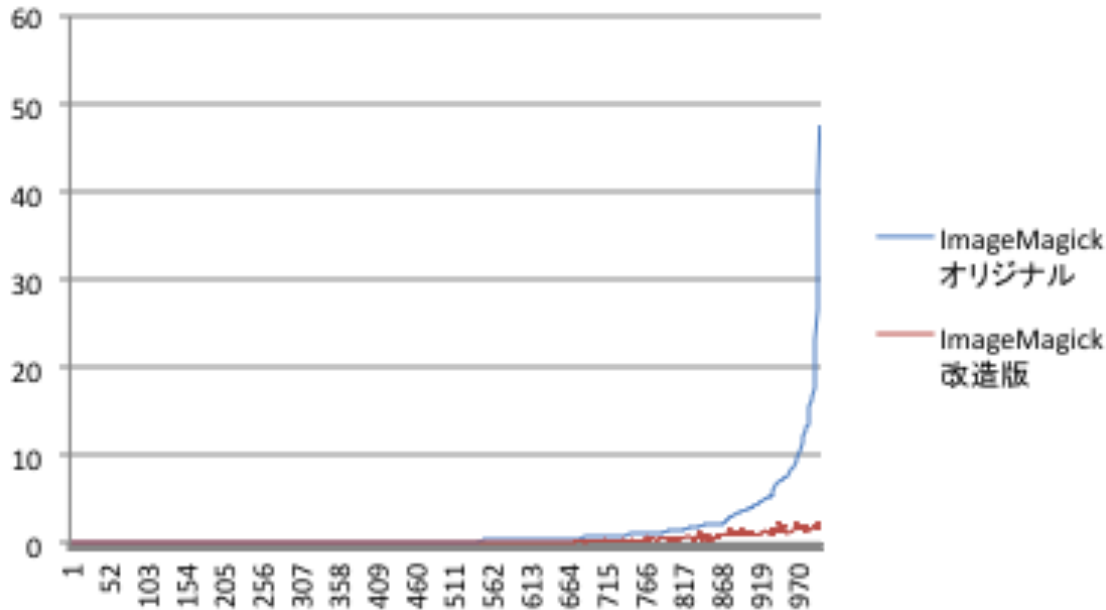
- これだけの量の実装なので、確認して使った方が良いでしょう

減色処理高速化

- 以下の記事が詳しいです
 - ImageMagick 改造入門 (その四)
 - <http://labs.gree.jp/blog/2013/12/9290/>
- GIF アニメーション生成も速くなるはず。
- GIFアニメの性能ネックは減色処理という話
 - GIFアニメ生成は本当にGraphicsMagickで行うべきか？
 - <http://labs.gree.jp/blog/2013/05/8132/>

どの位速くなったか

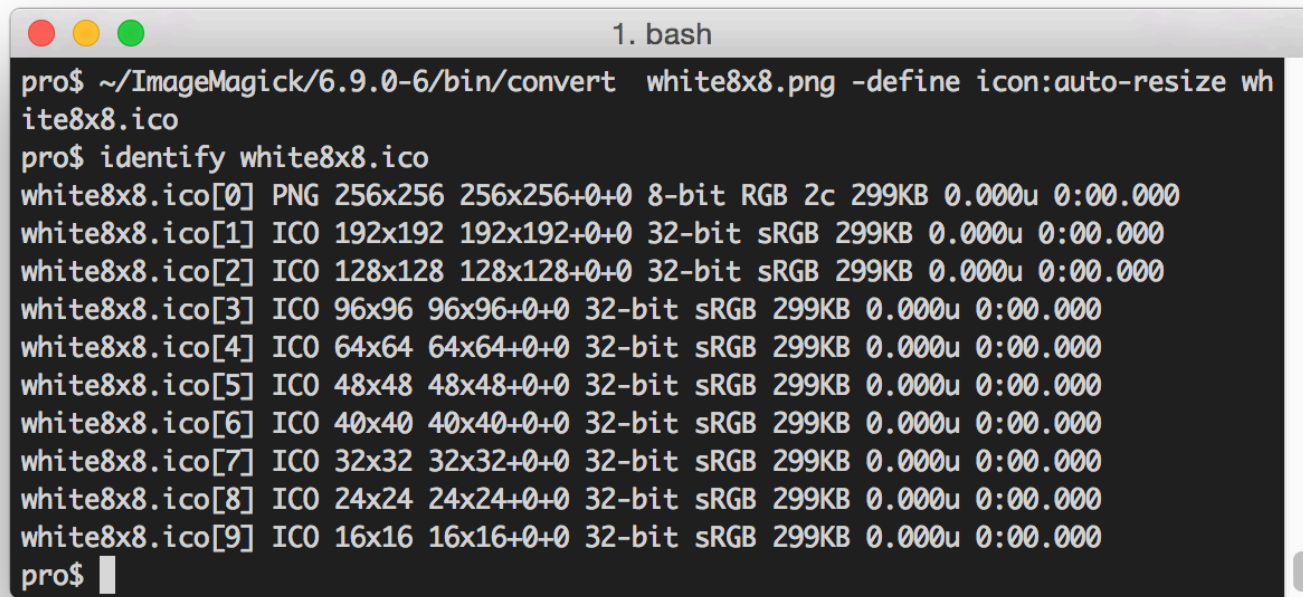
- グラデーション等で色数が多い時に超速い



- 引用元) <http://labs.gree.jp/blog/2012/10/5249/>

最近のトピック (1/2)

- ico の auto-resize (ImageMagick-6.8.8-2)
 - 1つファイルを渡せば favicon に必要なサイズにリサイズして格納してくれる。



```
1. bash
pro$ ~/ImageMagick/6.9.0-6/bin/convert white8x8.png -define icon:auto-resize white8x8.ico
pro$ identify white8x8.ico
white8x8.ico[0] PNG 256x256 256x256+0+0 8-bit RGB 2c 299KB 0.000u 0:00.000
white8x8.ico[1] ICO 192x192 192x192+0+0 32-bit sRGB 299KB 0.000u 0:00.000
white8x8.ico[2] ICO 128x128 128x128+0+0 32-bit sRGB 299KB 0.000u 0:00.000
white8x8.ico[3] ICO 96x96 96x96+0+0 32-bit sRGB 299KB 0.000u 0:00.000
white8x8.ico[4] ICO 64x64 64x64+0+0 32-bit sRGB 299KB 0.000u 0:00.000
white8x8.ico[5] ICO 48x48 48x48+0+0 32-bit sRGB 299KB 0.000u 0:00.000
white8x8.ico[6] ICO 40x40 40x40+0+0 32-bit sRGB 299KB 0.000u 0:00.000
white8x8.ico[7] ICO 32x32 32x32+0+0 32-bit sRGB 299KB 0.000u 0:00.000
white8x8.ico[8] ICO 24x24 24x24+0+0 32-bit sRGB 299KB 0.000u 0:00.000
white8x8.ico[9] ICO 16x16 16x16+0+0 32-bit sRGB 299KB 0.000u 0:00.000
pro$
```


最近のトピック (2/2)

- inline 形式出力変換 (ImageMagick-6.9.0-4)
 - Web の data スキーマ base64 inline 画像

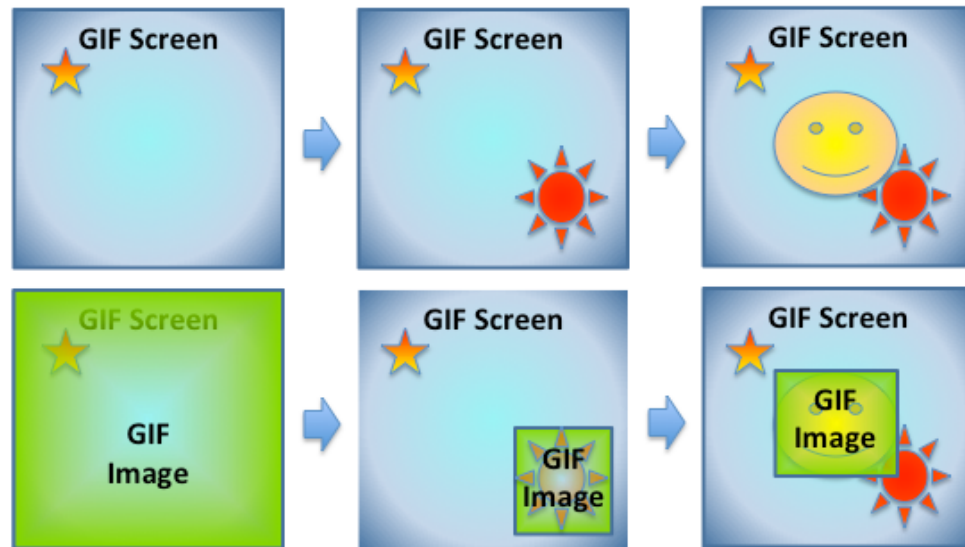
```
% convert white8x8.png inline:jpeg:/dev/stdout  
data:image/jpeg;base64,/9j/4AAQSkZJRgABAQIAJgAmAAD/  
2wBDAAMCAgICAgMCAgIDAwMDBAYEBAQEBAgGBgUGCQgKCgkICQk  
KDA8MCgsOCwkJDRENDg8QEBEQCgwSExIQEw8QEBD/  
wAALCAAIAAgBAREA/8QAFAABAAAAAAAAAAAAAAAAAAACf/  
EABQQAQAAAAAAAAAAAAAAAAAAAD/2gAIAQEAAAD8AVN//2Q==
```

GraphicsMagick は？ (1/2)

- GIFアニメ生成は本当にGraphicsMagickで行うべきか？
 - <http://labs.gree.jp/blog/2013/05/8132/>
- この記事から考察すると、
 - ImageMagick で GIF 最適化が入る辺りで遅くなった
 - GraphicsMagick はその前の ImageMagick-5.5.4 からフォークしてるので速い
 - けど、今の ImageMagick は速度的に遜色ない
 - GraphicsMagick はGIF最適化出来ないし画質も微妙

閑話休題)GIFアニメ最適化

- 変化のある部分をクリップして保持する画素数を減らす
- 更に変化のないピクセルを透明にして圧縮率を上げる



- 引用元) <http://labs.gree.jp/blog/2012/07/2446/>

GraphicsMagick は？ (2/2)

- GraphicsMagick のデフォルトが Q8 なので、ImageMagick も(Q16でなく) Q8 で作り直して比較しないとアンフェアなので注意

(尚、減色は Q8,16の影響はほぼ無い)

最後に (1/2)

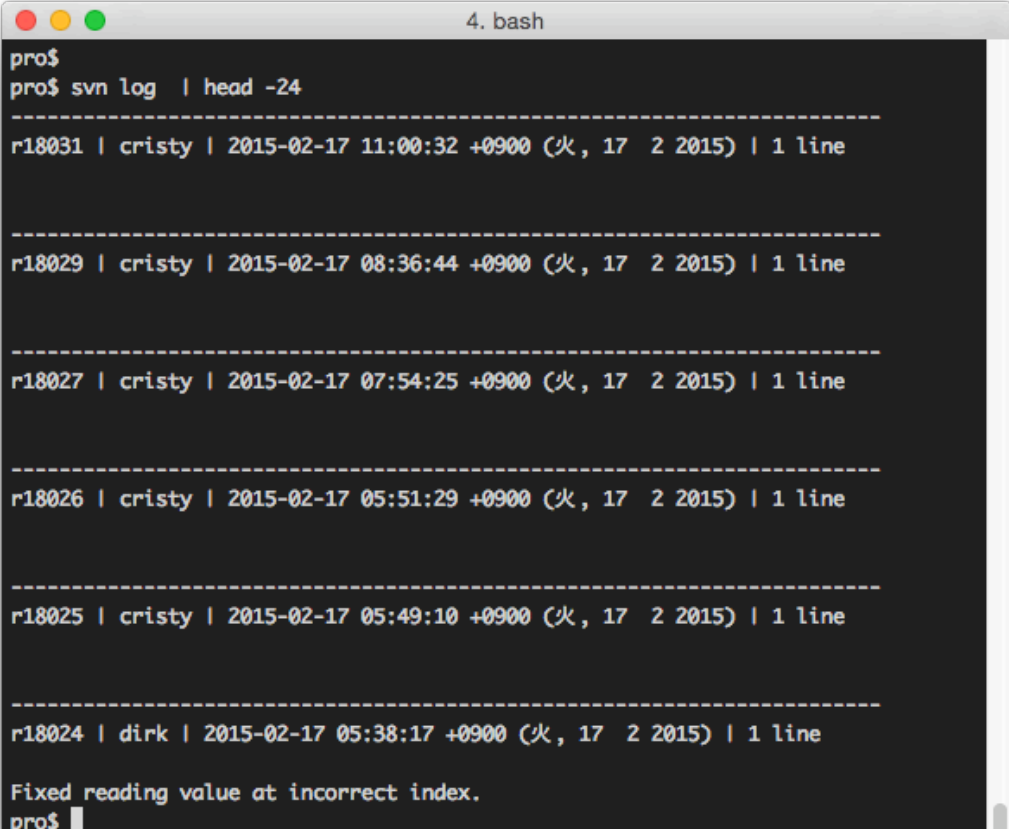
- 現在進行形の開発情報を知りたかったり、バグ報告をしたい時は、こちらへ。
 - <http://www.imagemagick.org/discourse-server/index.php>
- 今日の話を聞いて、自分も ImageMagick のコードを弄りたくなった人へ ⇒

最後 (2/2)

※ Cristyさんは
メインプログラマ

svnレポジトリに
コメントが無い！

コードが分かり
易いのでOK！！

A terminal window titled "4. bash" showing the output of the command "svn log | head -24". The output lists several revisions by user "cristy" on February 17, 2015, with timestamps and commit messages in Japanese. The last revision is by user "dirk". The terminal ends with an error message "Fixed reading value at incorrect index." and a prompt "pro\$".

```
pro$  
pro$ svn log | head -24  
-----  
r18031 | cristy | 2015-02-17 11:00:32 +0900 (火, 17 2 2015) | 1 line  
  
-----  
r18029 | cristy | 2015-02-17 08:36:44 +0900 (火, 17 2 2015) | 1 line  
  
-----  
r18027 | cristy | 2015-02-17 07:54:25 +0900 (火, 17 2 2015) | 1 line  
  
-----  
r18026 | cristy | 2015-02-17 05:51:29 +0900 (火, 17 2 2015) | 1 line  
  
-----  
r18025 | cristy | 2015-02-17 05:49:10 +0900 (火, 17 2 2015) | 1 line  
  
-----  
r18024 | dirk | 2015-02-17 05:38:17 +0900 (火, 17 2 2015) | 1 line  
Fixed reading value at incorrect index.  
pro$
```

質問タイム

- まとめ
 - ImageMagick について
 - ImageMagick の構造
 - 色んなバージョンの実行比較
 - 高速化でよく聞く話 Q8,openmp,jpegsize
 - ImageMagick-6.8.7-4 のトピック (opencl,減色)
 - 最近の気になるトピック (ico, inline)
 - GraphicsMagick の dis り

- ありがとうございます