

# Image Optimization in Rails

For an up-to-date version and discussions see  
<https://github.com/consti/image-optimization-howto>

Constantin Hofstetter

November 7, 2014

# Reduce file size of static assets

## ImageOptim

Reduce image file sizes significantly by using multiple optimization (lossless compression) libraries packaged into one tool. Available as:

- ▶ GUI Application
- ▶ CLI Application
- ▶ Extension for Middleman

## ImageAlpha

Convert PNG24 images with transparent background to PNG8+alpha images to reduce file size. Available [here](#).

# Reduce file size of static assets

## ImageOptim CLI as pre-commit hook

Automatically optimize images that are committed to your git repo:

```
# in your_project/.git/hooks/pre-commit
images=$(git diff --exit-code --cached --name-only --diff-
  filter=ACM -- '*.png' '*.jpg')
$(exit $? ) || echo $images | imageoptim && git add $images
```

# Paperclip integration

## Using imagemagick parameters

Add options to the `convert_options` parameter:

- ▶ `strip`: remove EXIF data
- ▶ `quality PERCENTAGE`: compress image data (not lossless)
- ▶ `interlace METHOD`: make the image progressive

Example:

```
# In your model
has_attached_file :avatar, {
  styles: { thumb: '60x60#' },
  convert_options: {
    all: '-strip -quality 70 -interlace Line'
  }
}
```

# Paperclip integration

## Using `paperclip-optimizer` processor

The `paperclip-optimizer` gem uses the `image_optim` library instead of `imagemagick` to yield better optimization results.

# Paperclip integration

## Using paperclip-optimizer processor

```
# Add paperclip-optimizer gem
# Install image_optim binaries / image_optim_bin gem (HEROKU)
has_attached_file :avatar, {
  styles: {
    thumb: {
      geometry: '60x60#',
      paperclip_optimizer: {
        # add compression (not lossless) to optimization (
          lossless)
        jpegoptim: { max_quality: 70 },
        # Disable unavailable binaries
        svgo: false
      }
    }
  },
  # We still need to strip EXIF and make the image progressive
  convert_options: { all: '-strip -interlace Line' },
  processors: [:paperclip_optimizer]
}
```

# Paperclip integration

paperclip-optimizer: gotcha

If paperclip does not resize the images to the different styles, add a dummy processor that pre-processes the image to the different styles.

```
# in lib/paperclip_processors/resizer.rb
module Paperclip
  class Resizer < Thumbnail
    def transformation_command
      super
    end
  end
end
```

```
# in your model
has_attached_file :avatar, {
  processors: [:resizer, :paperclip_optimizer]
}
```