Ruby+ImageMagick™
Version 2.12.0

User's Guide and Reference

Table of Contents

User's Guide

Reference

RVG - Ruby Vector Graphics

Introduction

What is RMagick?

RMagick is a binding from Ruby to the ImageMagick ™ image manipulation library. Here's how the ImageMagick home page describes ImageMagick:

> ImageMagick™... is a free software suite to create, edit, and compose bitmap images. It can read, convert and write images in a large variety of formats. Images can be cropped, colors can be changed, various effects can be applied, images can be rotated and combined, and text, lines, polygons, ellipses and Bézier curves can be added to images and stretched and rotated.

ImageMagick offers a full range of image processing tools that provide the capability to:

- Convert an image from one format to another (e.g. TIFF to JPEG)
- Resize, rotate, sharpen, color reduce, or add special effects to an image
- Create a montage of image thumbnails
- Create a transparent image suitable for use on the Web
- Turn a group of images into a GIF animation sequence
- Create a composite image by combining several separate images
- Draw shapes or text on an image
- Decorate an image with a border or frame
- Describe the format and characteristics of an image

RMagick is a complete interface to ImageMagick. Version 1.0.0 was released in February, 2003. Within its four major classes and 18 minor classes RMagick defines over 650 methods and 350 constants. RMagick exploits Ruby idioms such as blocks and iterators, Struct classes, symbols, ?- and !-suffixed methods, and exceptions.

About this document

This document describes Version 2.12.0 of RMagick. It is divided into 4 parts. The first is this page. The second part is a user's guide covering both RMagick and ImageMagick usage and conventions. The third part is a reference guide to the ImageList, Image, and Draw classes. This guide includes many examples. The fourth part covers Ruby Vector Graphics (RVG). RVG is a facade for Draw that provides a high-level API for scalable vector graphics. The RVG section includes a tutorial and complete reference documentation.

Accompanying the HTML documentation is a set of over 175 example RMagick programs. Each is a complete, stand-alone program that either creates an image from scratch or modifies an input image using one or more RMagick methods. The output images are used as illustrations in the HTML documentation. Throughout this document, click any example image to see the program that created it.

Conventions

Names I've tried to follow existing Ruby conventions in this document. A class is identified by its name, which always starts with a capital letter. An object is referred to by its class name, starting with a small letter. For example, a generic ImageList object is referred to as an *imagelist*. Class methods are identified like this: `Class.method`. Instance methods are identified like this: `Class#method`.

RMagick is implemented in the **Magick** module, therefore its constants are in the Magick namespace. However, for clarity I've omitted the `Magick::` prefix in most places in this document. You can use the `include Magick` statement to add the constants and methods to the Object namespace.

ImageMagick documentation Text that looks like this is quoted from the ImageMagick documentation.

Example programs and images All of the example images on these pages were created by RMagick. Click the image to see the program code. (You may need to configure your browser to allow JavaScript and pop-up windows.) All example images show the result of applying the method. Many example images show the "before" image as well. When the image is accompanied by this symbol 🌀 mouse over the image to see the "before" version.

How to get help

If you have a question that is not answered by these pages, you can post it at the Support Request Tracker or the Feature Request Tracker on RubyForge, or email me at `rmagick at rubyforge dot org`.

Reporting bugs

Please report problems with RMagick installation and usage to the Bug Tracker at RubyForge, or email me at `rmagick at rubyforge dot org`.

For quickest results, include the RMagick, ImageMagick and Ruby version numbers, along with a thorough description of the problem. RMagick will tell you both its version number and ImageMagick's version number. Simply bring up irb and run this command:

```
$ ruby -rRMagick -e "puts Magick::Long_version"
This is RMagick 2.0.0 ($Date: 2009/02/28 23:50:16 $) Copyright (C) 2007 by Timothy P. Hunter
Built with ImageMagick 6.3.7 12/20/07 Q16 http://www.imagemagick.org
Built for ruby 1.8.6
Web page: http://rmagick.rubyforge.org
Email: rmagick@rubyforge.org
```

(Using RubyGems? If you haven't added `rubygems` to your RUBYOPT environment variable you may need to use this command instead: `ruby -rubygems -r RMagick -e "puts Magick::Long_version"`.)

It will help a lot if you supply a small Ruby program that reproduces the problem. Don't forget to include any input image files!

Please remember I can't help with Ruby or ImageMagick installation and configuration problems. For help with Ruby, post your questions to `comp.lang.ruby`. For help with ImageMagick, you can post to the ImageMagick Discourse Server.

I'm not an image processing guru, either, so I probably won't be able to help with questions about specific image formats and transformations. I will be glad, however, to intermediate between RMagick users and the ImageMagick developers.

Copyright Notices

« Prev | Contents | Next »