README

May 2, 2016

Contents

1 TODO 1

2 Packaging 2

2.1 Traveling Ruby 2

2.1.1 Preparation 2

2.1.2 Creating a batch file 3

2.1.3 Modifying the Rakefile 3

2.1.4 Creating and testing the package 7

2.1.5 NB 7

3 Shoes 7

3.1 Issues 9

3.1.1 Packaging 9

4 Main 11

5 Classes 16

6 Modules 17

6.1 Sheets 17

1 TODO

- ☒ input as xls file
- ☐ gui?
  - ☐ shoes!

2 Packaging

2.1 Traveling Ruby

Tutorial 4: creating packages for Windows

https://github.com/phusion/traveling-ruby/blob/master/TUTORIAL-4.md

./packages/traveling-ruby/README.org

2.1.1 Preparation

```
rbenv local 2.1.9
```

```
mkdir packaging
```

./Gemfile

```
source 'https://rubygems.org'

gem 'faker'
gem 'sqlite3', '1.3.9'

group :development do
  gem 'rake'
end
```

./hello.rb

```
#!/usr/bin/env ruby
require 'faker'
require 'sqlite3'

db = SQLite3::Database.new("hello.sqlite3")
db.execute("create table if not exists foo (name varchar(255))")
db.execute("insert into foo values ('hello world')")
db.close
puts "Hello #{Faker::Name.name}, database file modified."
```

./packaging/wrapper.sh

```
#!/bin/bash
set -e
```

```
# Figure out where this script is located
SELFDIR="‘dirname \"$0\"‘"
SELFDIR="‘cd \"$SELFDIR\" && pwd‘"

# Tell Bundler where the Gemfile and gems are.
export BUNDLE_GEMFILE="$SELFDIR/lib/vendor/Gemfile"
unset BUNDLE_IGNORE_CONFIG

# Run the actual app using the bundled Ruby interpreter, with Bundler activated.
exec "$SELFDIR/lib/ruby/bin/ruby" -r bundler/setup "$SELFDIR/lib/app/hello.rb"
```

```
chmod +x packaging/wrapper.sh
```

```
   ./packaging/bundler-config
```

```
BUNDLE_PATH: .
BUNDLE_WITHOUT: development
BUNDLE_DISABLE_SHARED_GEMS: ’1’
```

```
bundle install
```

2.1.2 Creating a batch file

```
./packaging/wrapper.bat
```

```
@echo off

:: Tell Bundler where the Gemfile and gems are.
set "BUNDLE_GEMFILE=%~dp0\lib\vendor\Gemfile"
set BUNDLE_IGNORE_CONFIG=

:: Run the actual app using the bundled Ruby Interpreter, with Bundler activated.
@"%~dp0\lib\ruby\bin\ruby.bat" -rbundler/setup "%~dp0\lib\app\hello.rb"
```

2.1.3 Modifying the Rakefile

```
./Rakefile
```

```
# For Bundler.with_clean_env
require ’bundler/setup’
```

```
PACKAGE_NAME = "hello"
VERSION = "1.0.0"
TRAVELING_RUBY_VERSION = "20150210-2.1.5"
SQLITE3_VERSION = "1.3.9"  # Must match Gemfile

desc "Package your app"
task :package => ['package:linux:x86', 'package:linux:x86_64', 'package:osx', 'package

namespace :package do
  namespace :linux do
    desc "Package your app for Linux x86"
    task :x86 => [:bundle_install,
                  "packaging/traveling-ruby-#{TRAVELING_RUBY_VERSION}-linux-x86.tar.gz"
                  "packaging/traveling-ruby-#{TRAVELING_RUBY_VERSION}-linux-x86-sqlite3
    ] do
      create_package("linux-x86")
    end

    desc "Package your app for Linux x86_64"
    task :x86_64 => [:bundle_install,
      "packaging/traveling-ruby-#{TRAVELING_RUBY_VERSION}-linux-x86_64.tar.gz",
      "packaging/traveling-ruby-#{TRAVELING_RUBY_VERSION}-linux-x86_64-sqlite3-#{SQLITE
    ] do
      create_package("linux-x86_64")
    end
  end

  desc "Package your app for OS X"
  task :osx => [:bundle_install,
                "packaging/traveling-ruby-#{TRAVELING_RUBY_VERSION}-osx.tar.gz",
                "packaging/traveling-ruby-#{TRAVELING_RUBY_VERSION}-osx-sqlite3-#{SQLIT
               ] do
    create_package("osx")
  end

  desc "Package your app for Windows x86"
  task :win32 => [:bundle_install,
                  "packaging/traveling-ruby-#{TRAVELING_RUBY_VERSION}-win32.tar.gz"#,
                  #"packaging/traveling-ruby-#{TRAVELING_RUBY_VERSION}-win32-sqlite3-#
```

```
                 ] do
    create_package("win32", :windows)
  end

  desc "Install gems to local directory"
  task :bundle_install do
    if RUBY_VERSION !~ /^2\.1\./
      abort "You can only ’bundle install’ using Ruby 2.1, because that’s what Travelin
    end
    sh "rm -rf packaging/tmp"
    sh "mkdir packaging/tmp"
    sh "cp Gemfile Gemfile.lock packaging/tmp/"
    Bundler.with_clean_env do
      sh "cd packaging/tmp && env BUNDLE_IGNORE_CONFIG=1 bundle install --path ../vendo
    end
    sh "rm -rf packaging/tmp"
    sh "rm -f packaging/vendor/*/*/cache/*"
    sh "rm -rf packaging/vendor/ruby/*/extensions"
    sh "find packaging/vendor/ruby/*/gems -name ’*.so’ | xargs rm -f"
    sh "find packaging/vendor/ruby/*/gems -name ’*.bundle’ | xargs rm -f"
    sh "find packaging/vendor/ruby/*/gems -name ’*.o’ | xargs rm -f"
  end
end

file "packaging/traveling-ruby-#{TRAVELING_RUBY_VERSION}-linux-x86.tar.gz" do
  download_runtime("linux-x86")
end

file "packaging/traveling-ruby-#{TRAVELING_RUBY_VERSION}-linux-x86_64.tar.gz" do
  download_runtime("linux-x86_64")
end

file "packaging/traveling-ruby-#{TRAVELING_RUBY_VERSION}-osx.tar.gz" do
  download_runtime("osx")
end

# file "packaging/traveling-ruby-#{TRAVELING_RUBY_VERSION}-win32-sqlite3-#{SQLITE3_VERS
#   download_runtime("win32", "sqlite3-#{SQLITE3_VERSION}")
# end
```

```
file "packaging/traveling-ruby-#{TRAVELING_RUBY_VERSION}-win32.tar.gz" do
  download_runtime("win32")
end

file "packaging/traveling-ruby-#{TRAVELING_RUBY_VERSION}-linux-x86-sqlite3-#{SQLITE3_VE
  download_native_extension("linux-x86", "sqlite3-#{SQLITE3_VERSION}")
end

file "packaging/traveling-ruby-#{TRAVELING_RUBY_VERSION}-linux-x86_64-sqlite3-#{SQLITE3
  download_native_extension("linux-x86_64", "sqlite3-#{SQLITE3_VERSION}")
end

file "packaging/traveling-ruby-#{TRAVELING_RUBY_VERSION}-osx-sqlite3-#{SQLITE3_VERSION}
  download_native_extension("osx", "sqlite3-#{SQLITE3_VERSION}")
end

def create_package(target, os_type = :unix)
  package_dir = "#{PACKAGE_NAME}-#{VERSION}-#{target}"
  sh "rm -rf #{package_dir}"
  sh "mkdir #{package_dir}"
  sh "mkdir -p #{package_dir}/lib/app"
  sh "cp hello.rb #{package_dir}/lib/app/"
  sh "mkdir #{package_dir}/lib/ruby"
  sh "tar -xzf packaging/traveling-ruby-#{TRAVELING_RUBY_VERSION}-#{target}.tar.gz -C #

  if os_type == :unix
    sh "cp packaging/wrapper.sh #{package_dir}/hello"
  else
    sh "cp packaging/wrapper.bat #{package_dir}/hello.bat"
  end

  sh "cp -pR packaging/vendor #{package_dir}/lib/"
  sh "cp Gemfile Gemfile.lock #{package_dir}/lib/vendor/"
  sh "mkdir #{package_dir}/lib/vendor/.bundle"
  sh "cp packaging/bundler-config #{package_dir}/lib/vendor/.bundle/config"
  if os_type == :unix
    sh "tar -xzf packaging/traveling-ruby-#{TRAVELING_RUBY_VERSION}-#{target}-sqlite3-#
  else
    sh "tar -xzf packaging/traveling-ruby-#{TRAVELING_RUBY_VERSION}-#{target}.tar.gz "
  end
  if !ENV['DIR_ONLY']
    if os_type == :unix
      sh "tar -czf #{package_dir}.tar.gz #{package_dir}"
    else
      sh "zip -9r #{package_dir}.zip #{package_dir}"
    end
    sh "rm -rf #{package_dir}"
  end
end

def download_runtime(target, gem_name_and_version)
  sh "cd packaging && curl -L -O --fail " +
    "https://d6r77u77i8pq3.cloudfront.net/releases/traveling-ruby-#{TRAVELING_RUBY_VERS
end

def download_native_extension(target, gem_name_and_version)
  sh "curl -L --fail -o packaging/traveling-ruby-#{TRAVELING_RUBY_VERSION}-#{target}-#
    "https://d6r77u77i8pq3.cloudfront.net/releases/traveling-ruby-gems-#{TRAVELING_RUBY
end
```

2.1.4 Creating and testing the package

```
rake package:win32
```

2.1.5 NB

important Windows-specific caveats

3 Shoes

```
./little.rb
```

```
require 'cj-parser'

cj_file = ARGV[0].to_s
days = ARGV[1]


Shoes.app(title: "Case Jewelry Label Maker", width: 800, height: 1000, resizable: true
  #background "#c3f4f8".."#fff"
```

```
#background "#000"
background "img/sheet.png"

@days_display = stack(margin: 12) do
  @days_para = para strong("0")

  @days_para
end

stack(margin: 12) do
  para "Number of Days"
  flow {
    #@days = edit_line :width => 50

    button "Choose File" do
      @file = ask_open_file
    end

    @days_button = button "Days" do
      @days = ask("How many days back?")
      @days_display.clear do
        @days_para = para strong(@days)

        @days_para
      end
    end

    button "Submit" do
      #alert "Making sheets from #{@file} from last #{@days.text} days"
      #alert "Making sheets from #{@file} from last #{@days} days"
      #set_variables(@days.text)

      if confirm("Make sheets from #{@file} from last #{@days} days?")
        set_variables(@days)
        Sheets.make_sheets(@file)
      end
    end

    button "X" do
      exit()
      end
    }

    # @sheet = image(
    #   #"img/label.png",
    #   "img/sheet.png",
    #   width: 850,
    #   height: 1100
    # )

  end
}
```

3.1 Issues

3.1.1 Packaging

- Mon May 2 02:59:40 CDT 2016
  - try `https://github.com/phusion/traveling-ruby`
    - tutorial 1: hello world `file:///home/son/bin/Ruby/TravelingRuby/hello_app/README.org`
    - tutorial 2: gem dependencies `file:///home/son/bin/Ruby/TravelingRuby/hello_app_with_gem_dependencies/README.org`
    - tutorial 3: native extensions `file:///home/son/bin/Ruby/TravelingRuby/hello_app_with_native_extensions/README.org`
    - tutorial 4: windows `file:///home/son/bin/Ruby/TravelingRuby/hello_app_windows/README.org`
- Mon May 2 01:35:24 CDT 2016

  ```
  cd src
  gem install warbler
  shoes -p swt:jar little.rb
  ```

```
Packaging swt:jar...
Furoshiki::ConfigurationError: Invalid configuration.
 - Run file configured as 'bin/hello_from_warbler', \
   but couldn't find file at bin/hello_from_warbler
 - OS X icon file configured as 'static/Shoes.icns', \
but couldn't find file at static/Shoes.icns

initialize at /home/son/.rbenv/versions/jruby-9.0.5.0/lib/ruby/gems/shared/gems/fu
create_packager at /home/son/.rbenv/versions/jruby-9.0.5.0/lib/ruby/gems/shared/ge
block in run at /home/son/.rbenv/versions/jruby-9.0.5.0/lib/ruby/gems/shared/gems/
each at org/jruby/RubyArray.java:1560
run at /home/son/.rbenv/versions/jruby-9.0.5.0/lib/ruby/gems/shared/gems/shoes-swt
run at /home/son/.rbenv/versions/jruby-9.0.5.0/lib/ruby/gems/shared/gems/shoes-cor
run at /home/son/.rbenv/versions/jruby-9.0.5.0/lib/ruby/gems/shared/gems/shoes-cor
<top> at /home/son/.rbenv/versions/jruby-9.0.5.0/lib/ruby/gems/shared/gems/shoes-s
load at org/jruby/RubyKernel.java:955
<top> at /home/son/.rbenv/versions/jruby-9.0.5.0/bin/shoes-swt:23
```

- breakdown

  ```
  Packaging swt:jar...
  ```

  ```
  Furoshiki::ConfigurationError:
  ```

  ```
  Invalid configuration.
   - Run file configured as 'bin/hello_from_warbler', \
     but
  ```

  ```
  couldn't find file at bin/hello_from_warbler
  ```

  ```
   - OS X icon file configured as 'static/Shoes.icns', \
  but couldn't find file at static/Shoes.icns
  ```

  - remove app.yaml

    ```
    Packaging swt:jar...
    NameError: uninitialized constant Bundler
              const_missing at org/jruby/RubyModule.java:3212
    ```

    - `https://github.com/shoes/shoes4/issues/1148`

      ```
      ./src/Gemfile
      ```

      ```
      source 'https://rubygems.org'
      ```

```
gem 'shoes', '4.0.0.pre5'
gem 'shoes-swt'
gem 'warbler'
```

```
bundle
```

```
shoes -p swt:jar little.rb
java -jar little.rb
```

```
Ignoring shoes-core-4.0.0.pre5 because its extensions are not \
 built.  Try: gem pristine shoes-core --version 4.0.0.pre5
LoadError: no such file to load -- shoes
  require at org/jruby/RubyKernel.java:1040
  require at /tmp/jruby6735080100235658419extract/jruby-stdlib-1.7.25.j
   (root) at file:/home/son/IBT/jewelry/Retail_Jewelry/src/pkg/little.j
  require at org/jruby/RubyKernel.java:1040
   (root) at /tmp/jruby6735080100235658419extract/jruby-stdlib-1.7.25.j
  require at /tmp/jruby6735080100235658419extract/jruby-stdlib-1.7.25.j
ERROR: org.jruby.embed.EvalFailedException: (LoadError) no such file t
```

- Dec 14, 2015 : last message at `https://github.com/shoes/shoes4/issues/1148`
- Source `https://github.com/davorb/shoes4`
- fork it, let's do this!

  `https://github.com/son1112/shoes4`

  ```
  git clone git@github.com:son1112/shoes4.git
  ```

  `file:///home/son/src/shoes4-fork`

  haha, should call it shoes4rk

4 Main

```
./cj-parser.rb

require 'json'
require 'chronic'
require 'rubyXL'
require 'docx'
require 'roo'

require './lib/sheets.rb'
require './lib/label.rb'
```

```
# write to take in xml file
def set_variables(days)

  $days = days.to_f

  $cj_path = Dir.pwd()
  $pdf_path = "#{$cj_path}/pdfs"
  $templates_path = "#{$cj_path}/lib/templates"
  $template_top = File.open("#{$templates_path}/template-top.txt").readlines
  $template_bottom = File.open("#{$templates_path}/template-bottom.txt").readlines

  $sheets_dir = "#{$pdf_path}/sheets"
  $labels_dir = "#{$pdf_path}/labels"
  $sheet_top = File.open("#{$templates_path}/labelsTemplate-top.txt").readlines

end

def strip(s)
  s.gsub(/"/, ’’).
    gsub(/g/, ’’).
    gsub(/G/, ’’).
    gsub(/,/, ’’).
    split(’ ’)
end

def nil_convert(c)
  if c.nil?
    ""
  else
    c
  end
end

def get_labels(file)
  puts "getting labels"

  labels = []

  xls_file = Roo::Spreadsheet.open(file)
```

```
xls_file.sheets.each do |sheet|

  sheet = xls_file.sheet(sheet)

  sheet.parse[4..-1].each do |row|

    zero,one,two,four,five,ten = nil_convert(row[0]),
    nil_convert(row[1]),
    nil_convert(row[2]),
    nil_convert(row[4]),
    nil_convert(row[5]),
    nil_convert(row[10])

    sizes = strip(five.to_s)
    gauge = "#{sizes[0]}g"
    size = "#{sizes[1]}\""
    desc = two.gsub("&", "and")
    id = one.to_s.split(/-/)[0]
    price = "$#{four.to_s.split(".")[0]}"
    supply = five
    updated = Chronic.parse(ten).to_f

    label = Label.new(gauge,
                      size,
                      desc,
                      id,
                      price,
                      supply,
                      updated
                     )

    seconds = 60*60*24*$days

    if (Time.now.to_f - updated.to_f) < seconds
      puts label.id
      $labelID = label.id
      labels.push label
    end

  end
  end

  # old csv code, keeping around for a rainy day
  # CSV.foreach(
  #   file,
  #   headers: false,
  #   skip_blanks: true,
  #   skip_lines: Regexp.union([ /^(?:,\s*)+$/, /^(?:Product)/ ]) ) do |row|

  #   size = row[5].to_s.gsub(/"/, '').gsub(/g/, '').gsub(/G/, '').gsub(/,/, '').split
  #   updated = Chronic.parse(row[10])

  #   label = Label.new("#{size[0]}g",
  #                        "#{size[1]}\"",
  #                        row[2].gsub("&", "and"),
  #                        row[1].to_s.split(/-/)[0],
  #                        row[4].to_s.split(".")[0],
  #                        row[5],
  #                        updated.to_f
  #                       )

  #   unless row[1] == "CASE JEWELRY-CJ"
  #     unless row[1] == "Product ID"
  #       if (Time.now.to_f - updated.to_f) < 60*60*24*$days
  #         puts label.id
  #         labels.push label
  #       end
  #     end
  #   end
  # end

  return labels

end

def rows_to_json(file)

  puts "converting rows to javascript object notation"

  json_file = "cj_db.json"
  count = get_labels(file).size

  File.open(json_file, "w") do |file|
    file.puts '{ "products": ['
  end

  get_labels(file).each_with_index do |row, index|
    File.open(json_file, "a") do |json|
      json.puts row.to_json

      unless index == count - 1
        json.puts ","
      end
    end
  end

  File.open(json_file, "a") do |file|
    file.puts '] }'
  end
end

def labels_to_tex(file)

  get_labels(file).each do |row|

    puts row.id

    tex_file = "#{row.id}.tex"
    pdf_file = "#{row.id}.pdf"

    if row.size == "\""
      size = row.gauge
    elsif row.gauge == ""
      size = row.size
    else
      size = "#{row.gauge} #{row.size}"
    end

    type = row.desc
    id = row.id
    price = row.price

    File.open(tex_file, "w") do |file|
      pre_script = "{\\scriptsize\\textit{"
      pre_lg = "{\\large"
      pre_LG = "{\\Large"
      post = "}}\n\n"

      file.puts $template_top

      file.puts "\\begin{center}" +
                "#{pre_lg}{" +
                "#{type}#{post}" +
                "\\end{center}"

      file.puts "\\begin{center}" +
                "#{pre_LG}" + "\\textit{" +
                "#{size}#{post}" +
                "\\end{center}"

      file.puts "\\begin{center}" +
                "#{pre_lg}{" +
                "#{id}\\hspace{25mm}  \\#{price}#{post}" +
                "\\end{center}"

      file.puts $template_bottom
    end

    ‘pdflatex #{tex_file} && mv *.tex *.aux *.log *.out tmp && mv *.pdf #{$pdf_path}‘
  end
end
```

5 Classes

```
./lib/label.rb
```

```
class Label
  #include Sheets

  def initialize(gauge, size, desc, id, price, supply, updated)
```

```
    @gauge = gauge
    @size = size
    @desc = desc
    @id = id
    @price = price
    @supply = supply
    @updated = updated
  end

  attr_reader :gauge, :size, :desc, :id, :price, :supply, :updated

end
```

6 Modules

6.1 Sheets

./lib/sheets.rb

```
module Sheets

  def Sheets.get_sheet_rows()
    Dir.chdir($pdf_path)

    files = Dir.entries(".").reject { |entry| File.directory?(entry) }
    pdfs = files.select { |file| file.end_with? ’.pdf’ }
    label_count = pdfs.count

    fboxs = []

    pdfs.each do |pdf|
      fboxs.push "\\framebox[1.0\\width]{\\includegraphics{#{$labels_dir}/#{pdf}}}"
    end

    rows = fboxs.each_slice(4).to_a
    return rows
  end

  def Sheets.get_sheets()
    pages = []

    get_sheet_rows.each do |row|
      pages.push row
    end

    sheets = pages.each_slice(8).to_a

    return sheets
  end

  def Sheets.make_sheets(file)


    rows_to_json(file)
    labels_to_tex(file)

    sheet_count = get_sheets.count

    if sheet_count >= 1

      puts "creating sheets"

      sheets = get_sheets

      i = 0

      puts "entering sheets directory"
      Dir.chdir($sheets_dir)
      ‘mv *.pdf bak‘

      sheets.each do |page|

        name = "sheet_000#{i}"
        filename = "#{name}.tex"

        puts "making #{name} sheet"
        File.open(filename, "w") do |file|
          file.puts $sheet_top
          file.puts "\\begin{center}"
          file.puts "\\setlength{\\fboxsep}{1pt}"
          file.puts "\\setlength{\\fboxrule}{0.1pt}"
        end

        page.each do |row|
          File.open(filename, "a") do |file|

            file.puts row
            file.puts "\\newline"

            row.each do |box|
              pdf = box.split("{").last.split("}").first.split("/").last
              ‘mv ../#{pdf} #{$labels_dir}‘
            end
          end
        end

        File.open(filename, "a") do |file|
          file.puts "\\end{center}"
          file.puts "\\end{document}"
        end

        i += 1

        #‘pdflatex #{filename} && evince #{name}.pdf && mv *.aux *.log *.out *.tex tex
        ‘pdflatex #{filename} && mv *.aux *.log *.out *.tex texfiles‘

      end

    end

    Dir.chdir($cj_path)

  end


  end
```