

Quick guide to CPU profiling Rails apps with k/qcachegrind

Step 1:

Acquire kcachegrind(KDE) or qcachegrind
(everything else)

- qcachegrind is available in homebrew

Step 2:

Grab `development_profiler.rb` or something similar to abstract away some of the profiling configuration, plop it in `lib/`

development_profiler.rb

```
class DevelopmentProfiler

  def self.prof(file_name)
    if Rails.env.development?
      RubyProf.start
      yield
      results = RubyProf.stop

      File.open "#{Rails.root}/tmp/performance/callgrind.#{file_name}.clt", 'w' do |file|
        RubyProf::CallTreePrinter.new(results).print(file)
      end
    else
      yield
    end
  end

end

end
```

Step 3:

Wrap your suspect code in a .prof block:

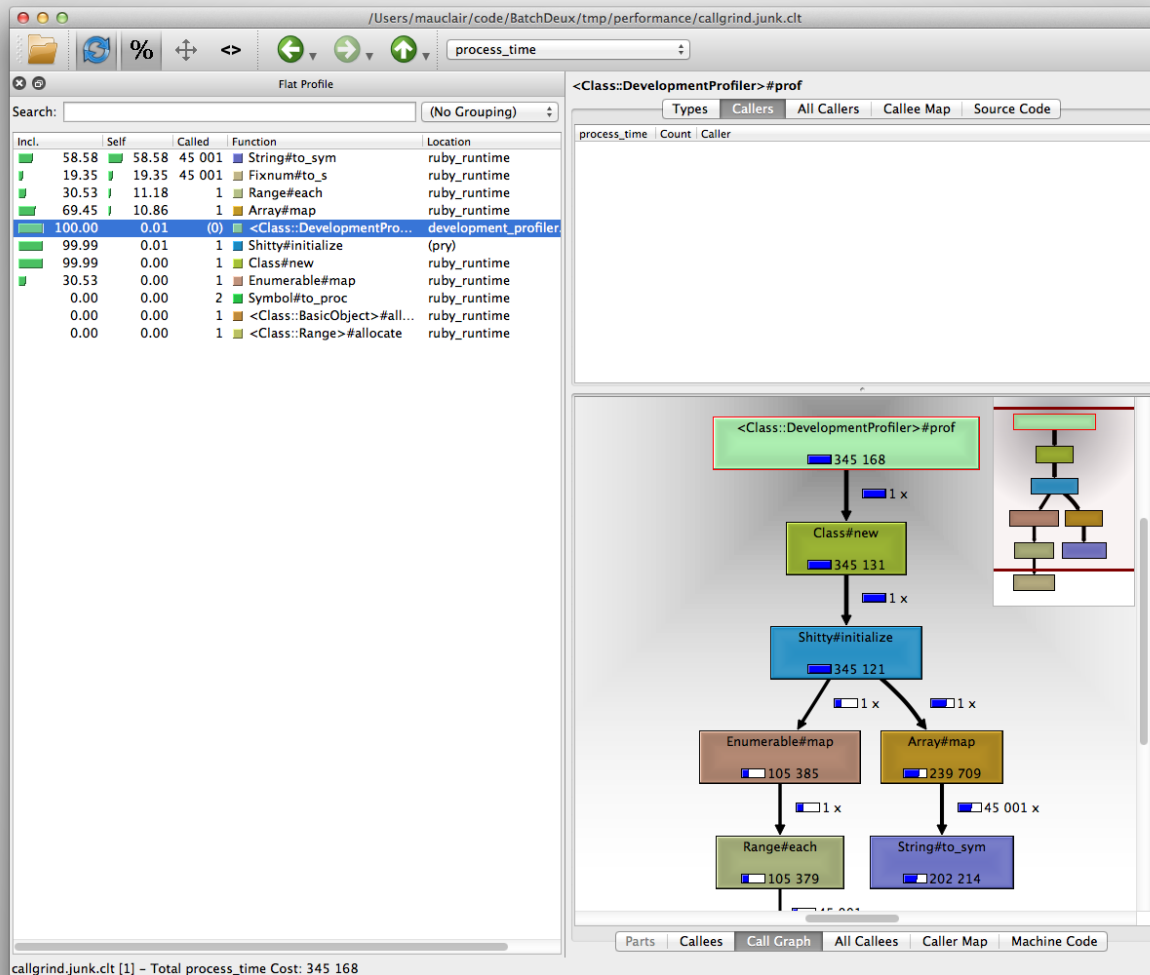
```
DevelopmentProfiler.prof('junk') do  
  SlowThing.new(45000)  
end
```

Step 4:

Run your code and bust open the file in
qcachegrind

Contrived Example

```
class Shitty
  def initialize(some_int)
    @symbols = (0..some_int).map(&:to_s).map(&:to_sym)
  end
end
```



Improved Contrived Example

```
class Better
  def initialize(some_int)
    @symbols = (0..some_int).map{|i| i.to_s.to_sym}
  end
end
```

