Entwicklungsprozesse in Open-Source-Projekten

Sommersemester 2011
Martin v. Löwis


Termin: Mi, 11:00-12:30, A-2.1

Veranstaltungsüberblick

Veranstaltungstermine

<table>
<thead>
<tr>
<th>Datum</th>
<th>Vortragende</th>
<th>Thema</th>
</tr>
</thead>
<tbody>
<tr>
<td>13.04.11</td>
<td>Martin v. Löwis</td>
<td>Einführung</td>
</tr>
<tr>
<td>20.04.11</td>
<td>Martin v. Löwis</td>
<td>Themenvergabe, Open Source Definition</td>
</tr>
<tr>
<td>27.04.11</td>
<td>Martin v. Löwis</td>
<td>Projektdiskussion</td>
</tr>
<tr>
<td>04.05.11</td>
<td>Martin v. Löwis</td>
<td>GNU Public License</td>
</tr>
<tr>
<td>11.05.11</td>
<td>Martin v. Löwis</td>
<td>Wie Python entwickelt wird</td>
</tr>
<tr>
<td>18.05.11</td>
<td>Martin v. Löwis</td>
<td>Eric Raymond: The Cathedral and the Bazaar</td>
</tr>
<tr>
<td>25.05.11</td>
<td>Martin v. Löwis</td>
<td>Keine Veranstaltung</td>
</tr>
</tbody>
</table>
Bug 632019 -

Summary: Middle-click on tab should close tab
LET'S DO THIS!
<table>
<thead>
<tr>
<th>Name</th>
<th>Date/Time</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nicolas Marchildon</td>
<td>2010-10-12 23:56:58 UTC</td>
<td>Firefox and Chrome support that: middle-click closes that tab.</td>
</tr>
<tr>
<td>chuchiperriman</td>
<td>2010-11-18 15:24:09 UTC</td>
<td>I like...</td>
</tr>
<tr>
<td>Tobias Pfeiffer</td>
<td>2011-07-12 14:38:57 UTC</td>
<td>Created attachment 191815 [details] [review]. Proposed Patch to close tabs on middle mouse button click. On my systems this patch works and adds the desired functionality. However the implementation is kind of ugly since the position of a gtk widget had to be determined and the only solution I found was copying a private method of GtkWidget. This is not good. I hope for feedback regarding this issue, plus other feedback since this is my first time programming GTK. Also please have a look at the commit message.</td>
</tr>
<tr>
<td>Tobias Pfeiffer</td>
<td>2011-07-12 22:27:05 UTC</td>
<td>Sorry I am a bit tired, the method wasn't copied from GTKWidget but from GTKNotebook... I am sorry for the confusion!</td>
</tr>
<tr>
<td>Sebastien Bacher</td>
<td>2011-08-03 10:50:11 UTC</td>
<td>duplicate of bug-445402?</td>
</tr>
<tr>
<td>Ignacio Casal Quinteiro (nacho)</td>
<td>2011-08-03 12:46:26 UTC</td>
<td>Thanks for the bug report. This particular bug has already been reported into our bug tracking system, but please feel free to report any further bugs you find. *** This bug has been marked as a duplicate of bug-445402 ***</td>
</tr>
</tbody>
</table>
Thanks for the bug report. This particular bug has already been reported into our bug tracking system, but please feel free to report any further bugs you find.

*** This bug has been marked as a duplicate of bug-445402 ***
The End
The End
The End

needless work

missing triaging
The End

no appreciation

missing triaging

needless work
The End

needless work
missing triaging
no appreciation
frustration
Stories in Open Source

Tobias Pfeiffer
@PragTob
pragtob.info
Welcome to Mendicant University

We are a group of software developers who use our technical skills to make a real positive impact on the world. Rather than learning in a traditional classroom setting, we sharpen our skills by working on real projects while helping each other out.
Shoes!

Ever wanted to build a GUI? Annoyed that it's so much effort? Shoes makes building little graphical programs for Mac, Windows, and Linux super simple.

Learn more »
Shoes.app width: 300, height: 200 do
  background lime..blue

  stack do
    para "Welcome to the world of Shoes!"
    button "Click me" do
      alert "Nice click!"
    end

  image "http://shoesrb.com/img/shoes-icon.png",
    margin_top: 20, margin_left: 10
  end
end
I love shoes.
Welcoming, friendly & helpful
OSS friends
Give people access
Release early, release often
You are not your OSS

Why

[Stick figures and colorful blocks]
class Dog
  def bark
    puts 'Wofooof'
  end
end

Dog.extend AfterDo
Dog.after :bark do puts 'I just heard a dog bark!' end

dog = Dog.new
dog.bark

# Output is:
# Wofooof
# I just heard a dog bark!
micro libraries are great
you have no idea how it’s used
list = Enum.to_list(1..10_000)
map_fun = fn i -> [i, i * i] end

Benchee.run(
  %{
    "flat_map" => fn -> Enum.flat_map(list, map_fun) end,
    "map.flatten" => fn -> list ▸ Enum.map(map_fun) ▸ List.flatten() end
  },
  time: 10,
  memory_time: 2
)
Use this!
learn as you go
just being better isn’t enough
co-maintainers

Use this!
We made it!

defmodule FileBench do
  @fixture "path/to/words.txt"
  def run do
    Benchee.run(%{
      "with read" => &with_read/0,
      "with stream" => &with_stream/0,
    }, time: 10)
  end
  def with_read do
    @fixture
    |> File.read!
    |> String.split("\n")
    |> Enum.max_by(&:String.length/1)
  end
  def with_stream do
    @fixture
    |> File.stream!
    |> Enum.max_by(&:String.length/1)
    |> String.Trim()
  end
end

The file has three functions. run executes the entire benchmark, comparing the with read results with the with stream results. The other two functions provide the implementations we’re benchmarking.

We’re ready to run it:

$ mix run -r perf/file_bench.exs -e "FileBench.run"
Number of Available Cores: 4
Available memory: 17.179869184 GB
Elixir 1.5.0
Erlang 19.0
Benchmark suite executing with the following configuration:
  warmup: 2.00 s
  time: 10.00 s
  parallel: 1
  inputs: none specified
Estimated total run time: 24.00 s
## All Files (93.0% covered at 3322.59 hits/line)

208 files in total. 6027 relevant lines. 5805 lines covered and 422 lines missed

<table>
<thead>
<tr>
<th>File</th>
<th>% covered</th>
<th>Lines</th>
<th>Relevant Lines</th>
<th>Lines covered</th>
<th>Lines missed</th>
<th>Avg. Hits / Line</th>
</tr>
</thead>
<tbody>
<tr>
<td>shoes-stdlib/shoes/swt/text_block/painter.rb</td>
<td>100.0%</td>
<td>33</td>
<td>16</td>
<td>16</td>
<td>0</td>
<td>113.7</td>
</tr>
<tr>
<td>shoes-stdlib/shoes/swt/text_block/fitter.rb</td>
<td>100.0%</td>
<td>220</td>
<td>100</td>
<td>100</td>
<td>0</td>
<td>1434.2</td>
</tr>
<tr>
<td>shoes-stdlib/shoes/swt/text_block/cursor_painter.rb</td>
<td>100.0%</td>
<td>54</td>
<td>30</td>
<td>30</td>
<td>0</td>
<td>3.0</td>
</tr>
<tr>
<td>shoes-stdlib/shoes/swt/text_block/centered_text_segment.rb</td>
<td>100.0%</td>
<td>24</td>
<td>9</td>
<td>9</td>
<td>0</td>
<td>1.2</td>
</tr>
<tr>
<td>shoes-stdlib/shoes/swt/text_block.rb</td>
<td>100.0%</td>
<td>161</td>
<td>89</td>
<td>89</td>
<td>0</td>
<td>1974.0</td>
</tr>
<tr>
<td>shoes-stdlib/shoes/swt/button.rb</td>
<td>100.0%</td>
<td>56</td>
<td>33</td>
<td>33</td>
<td>0</td>
<td>54.8</td>
</tr>
<tr>
<td>shoes-stdlib/shoes/swt/star.rb</td>
<td>100.0%</td>
<td>30</td>
<td>20</td>
<td>20</td>
<td>0</td>
<td>24.8</td>
</tr>
<tr>
<td>shoes-stdlib/shoes/swt/shoes_layout.rb</td>
<td>100.0%</td>
<td>62</td>
<td>37</td>
<td>37</td>
<td>0</td>
<td>5214.6</td>
</tr>
<tr>
<td>shoes-stdlib/shoes/swt/shape_painter.rb</td>
<td>100.0%</td>
<td>26</td>
<td>12</td>
<td>12</td>
<td>0</td>
<td>6.0</td>
</tr>
<tr>
<td>shoes-stdlib/shoes/swt/rect_painter.rb</td>
<td>100.0%</td>
<td>43</td>
<td>25</td>
<td>25</td>
<td>0</td>
<td>15.8</td>
</tr>
<tr>
<td>shoes-stdlib/shoes/swt/rect.rb</td>
<td>100.0%</td>
<td>31</td>
<td>20</td>
<td>20</td>
<td>0</td>
<td>23.2</td>
</tr>
<tr>
<td>shoes-stdlib/shoes/swt/radio.rb</td>
<td>100.0%</td>
<td>28</td>
<td>12</td>
<td>12</td>
<td>0</td>
<td>18.8</td>
</tr>
<tr>
<td>shoes-stdlib/shoes/swt/progress.rb</td>
<td>100.0%</td>
<td>37</td>
<td>22</td>
<td>22</td>
<td>0</td>
<td>11.2</td>
</tr>
<tr>
<td>shoes-stdlib/shoes/swt/oval.rb</td>
<td>100.0%</td>
<td>33</td>
<td>19</td>
<td>19</td>
<td>0</td>
<td>26.5</td>
</tr>
<tr>
<td>shoes-stdlib/shoes/swt/list_box.rb</td>
<td>100.0%</td>
<td>55</td>
<td>33</td>
<td>33</td>
<td>0</td>
<td>23.3</td>
</tr>
<tr>
<td>shoes-stdlib/shoes/swt/link_segment.rb</td>
<td>100.0%</td>
<td>113</td>
<td>63</td>
<td>63</td>
<td>0</td>
<td>19.5</td>
</tr>
<tr>
<td>shoes-stdlib/shoes/swt/line_painter.rb</td>
<td>100.0%</td>
<td>18</td>
<td>9</td>
<td>9</td>
<td>0</td>
<td>4.1</td>
</tr>
<tr>
<td>shoes-stdlib/shoes/swt/image_pattern.rb</td>
<td>100.0%</td>
<td>37</td>
<td>19</td>
<td>19</td>
<td>0</td>
<td>2.7</td>
</tr>
<tr>
<td>shoes-stdlib/shoes/swt/font.rb</td>
<td>100.0%</td>
<td>42</td>
<td>21</td>
<td>21</td>
<td>0</td>
<td>308.3</td>
</tr>
<tr>
<td>shoes-stdlib/shoes/swt/common/visibility.rb</td>
<td>100.0%</td>
<td>19</td>
<td>8</td>
<td>8</td>
<td>0</td>
<td>734.1</td>
</tr>
<tr>
<td>shoes-stdlib/shoes/swt/common/update_position.rb</td>
<td>100.0%</td>
<td>21</td>
<td>8</td>
<td>8</td>
<td>0</td>
<td>88.0</td>
</tr>
</tbody>
</table>
“legacy” OSS
joy of impact
you don’t owe people
nightmares of integration
First step into OSS
Howdy! Thanks for reporting an issue <3

Before you go ahead please search existing issues for your problem, chances are someone else already reported it.

To make sure that we can help you quickly please include and check the following information:

* Include how you run your tests and which testing framework or frameworks you are running.
  - please ensure you are requiring and starting SimpleCov before requiring any application code.
  - If running via rake, please ensure you are requiring SimpleCov at the top of your Rakefile
    For example, if running via RSpec, this would be at the top of your spec_helper.
  - Have you tried using a [`.simplecov` file](https://github.com/colszowka/simplecov#using-simplecov-for-centralized-config)?
* Include the SimpleCov version you are running in your report.
* If you are not running the latest version (please check), and you cannot update it, please specify in your report why you can’t update to the latest version.
* Include your `ruby -e “puts RUBY_DESCRIPTION”`
* Please also specify the gem versions of Rails (if applicable).
* Include any other coverage gems you may be using and their versions.

Include as much sample code as you can to help us reproduce the issue. (Inline, repo link, or gist, are fine. A failing test would help the most.)

Attach files by dragging & dropping, selecting or pasting them.
Howdy! Thanks for reporting an issue <3

Before you go ahead please search existing issues for your problem, chances are someone else already reported it.

To make sure that we can help you quickly please include and check the following information:

* Include how you run your tests and which testing framework or frameworks you are running.
  * please ensure you are requiring and starting SimpleCov before requiring any application code.
  * If running via rake, please ensure you are requiring SimpleCov at the top of your Rakefile.
  * For example, if running via RSpec, this would be at the top of your spec_helper.
  * Have you tried using a [`.simplecov` file](https://github.com/colszowka/simplecov#using-simplecov-for-centralized-config)?
* Include the SimpleCov version you are running in your report.
* If you are not running the latest version (please check), and you cannot update it, please specify in your report why you can’t update to the latest version.
* Include your `ruby -e “puts RUBY_DESCRIPTION”`.
* Please also specify the gem versions of Rails (if applicable).
* Include any other coverage gems you may be using and their versions.

Include as much sample code as you can to help us reproduce the issue. (Inline, repo link, or gist, are fine. A failing test would help the most.)

Attach files by dragging & dropping, selecting or pasting them.
Be nice!

Minimize

Howdy! Thanks for reporting an issue <3

Before you go ahead please search existing issues for your problem, chances are someone else already reported it.

To make sure that we can help you quickly please include and check the following information:

* Include how you run your tests and which testing framework or frameworks you are running.
  * please ensure you are requiring and starting SimpleCov before requiring any application code.
  * If running via rake, please ensure you are requiring SimpleCov at the top of your Rakefile. For example, if running via RSpec, this would be at the top of your spec_helper.
  * Have you tried using a [.simplecov file](https://github.com/colsowka/simplecov#using-simplecov-for-centralized-config)?
* Include the SimpleCov version you are running in your report.
* if you are not running the latest version (please check), and you cannot update it, please specify in your report why you can't update to the latest version.
* Include your `ruby -e "puts RUBY_DESCRIPTION"`.
* Please also specify the gem versions of Rails (if applicable).
* Include any other coverage gems you may be using and their versions.

Include as much sample code as you can to help us reproduce the issue. (Inline, repo link, or gist, are fine. A failing test would help the most.)

Attach files by dragging & dropping, selecting or pasting them.

Submit new issue
Be nice!

Minimize

Reproduce

Howdy! Thanks for reporting an issue <3

Before you go ahead please search existing issues for your problem, chances are someone else already reported it.

To make sure that we can help you quickly please include and check the following information:

* Include how you run your tests and which testing framework or frameworks you are running.
  * please ensure you are requiring and starting SimpleCov before requiring any application code.
  * If running via rake, please ensure you are requiring SimpleCov at the top of your Rakefile. For example, if running via RSpec, this would be at the top of your spec_helper.
  * Have you tried using a `.simplecov` file? (https://github.com/colszowka/simplecov#using-simplecov-for-centralized-config)?
* Include the SimpleCov version you are running in your report.
* If you are not running the latest version (please check), and you cannot update it, please specify in your report why you can't update to the latest version.
* Include your `ruby -e "puts RUBY_DESCRIPTION"`.
* Please also specify the gem versions of Rails (if applicable).
* Include any other coverage gems you may be using and their versions.

Include as much sample code as you can to help us reproduce the issue. (Inline, repo link, or gist, are fine. A failing test would help the most.)

Attach files by dragging & dropping, selecting or pasting them.
Hi @PragTob, thanks for such a thorough and well-researched bug report! And such an appreciative and friendly tone 😊! I'll be happy to take a look and see if I can help.
klyonrad commented on Jan 9 • edited

That's another thing where you/we might discuss on how to actually show the coverage of methods. Currently: when a file is loaded, all the `def method_name` lines are considered as a hit, the actual code in the method is shown as red and the closing of the method is also not considered as a hit (from the article)

Coverage was disabled for line number 7 as it contains just end keyword.

It always itched me that just declaring the method names are recorded as a hit or miss (except when nothing loads the file, which is the default for a rails project).

Anyway, that's maybe a distraction from the original topic...

**What exactly should be shown for the method coverage?** How does it affect the total big number of coverage a file/the project (that metric that we all care about). How do dynamically defined methods work with this? Playing devil's advocate: What does it even matter - I mean, what decisions would made knowing that a file has x amount of uncovered methods?

---

tycoon commented on Jan 9

In my experience, method coverage mostly helps finding unused methods that can be safely deleted in case you have 100% line (and probably branch) coverage.

For example you might have `attr_reader :foo, :bar` where only :foo is actually used. Or simply `attr_reader :bar` that is never used. In both cases only method coverage can find such stuff.

---

tycoon commented on Jan 9

Wait a second, you can detect unused `attr_reader` methods? That's amazing! Thanks for clearing that up.

Then I would say that for an unused method the `def method_name` line should also stay red - even when the LineCoverage API reports it as covered.
Add subprocess handling to simplecov #881

robotdana commented 5 hours ago

Using Process.fork (whether using the Parallel gem or directly) creates code that was invisible to SimpleCov by starting SimpleCov within the subprocess with its own command name and etc we can see that code :)

This also adds documentation for what to do when using Process.spawn or similar.

fixes: #414 and probably others

This seems the more correct place to fix this than my previous attempt here: grosser/parallel#275

Testing was a bit of a mess, perhaps you have some ideas. Also what are your thoughts on this being enabled by default or not?

Thanks :)

---

PragTob reviewed 3 hours ago

(I'm not clear why test 2.4.9 didn't work, it doesn't seem related?)

---

(Maybe) add a PR!
OSS vs Work
OSS

Be nice!
OSS

Be nice!

Your OSS, your time
OSS

Your OSS, your time

Adoption takes time

Be nice!
seemingly even less diverse

OSS
OSS seemingly even less diverse is this sustainable?
seemingly even less diverse

OSS

is this sustainable?

what about people using OSS for “evil”?
friends
friends

first talks
"try walking in my shoes" :) Learn about ruby shoes the Ruby GUI library at our next meetup this Thursday.

bit.ly/NZg8sD

3:30 PM · Jul 3, 2012 · Echofon
friends

first talks

@rug_b

"try walking in the Ruby Gumbo
bit.ly/NZg8sD"

3:30 PM · Jul 3, 2012 · Echofon

I love Shoes
I ♥️ Shoes

friends

first talks
friends

first talks

learning

all other projects
THE NOT SO LOW COST OF CALLING DYNAMICALLY

There are a couple of mantras that exist across programming communities, one of them is to avoid duplication or to keep it DRY. Programming languages equip us with different tools to avoid duplication. In Ruby, a popular way to achieve this is Metaprogramming. Methods are dynamically defined to get rid of all duplication and we expose them. There might be other problems with metaprogrammed solutions, but at least we are sure that the performance is the same as if we’ve written that duplicated code. Or are we?

As the title suggests, this post examines the performance of these meta-programmed methods. If you are looking for method definition performance or pure call overhead you’ll find this information in this post by Aaron Patterson.

Before we get into the details, I want to quickly highlight that this is not some theoretical improvements on an actual project. That work was done by my friend Jason R. Clark on the pull request over at Shoes 4. As he doesn’t have time to write it up, I get to do so, let’s get to it!