

Fuzzy Duplication Detection in Flay

Flay

What is Flay?

- Detects duplicate (identical) code.
- Detects structurally similar code.
- Language Independent; works on ASTs.
 - out of the box: ruby, erb.
 - Javascript, C, etc are all possible.

Structurally Similar?

```
def x(n)
  n + 1
end
```

```
def y(m)
  m + 2
end
```

Let's pick on rails!

flay rails

72,947!!

Identical (wow!)

```
#) IDENTICAL code found in :defn (mass*13 = 4563)
rails/activerecord/test/cases/finder_test.rb:851
rails/activerecord/test/cases/helper.rb:45
rails/activerecord/test/cases/migration_test.rb:405
rails/activesupport/test/core_ext/date_ext_test.rb:328
rails/activesupport/test/core_ext/date_time_ext_test.rb:320
rails/activesupport/test/core_ext/duration_test.rb:146
rails/activesupport/test/core_ext/numeric_ext_test.rb:123
rails/activesupport/test/core_ext/string_ext_test.rb:350
rails/activesupport/test/core_ext/time_ext_test.rb:798
rails/activesupport/test/core_ext/time_with_zone_test.rb:758
rails/activesupport/test/core_ext/time_with_zone_test.rb:953
rails/activesupport/test/json/encoding_test.rb:337
rails/activesupport/test/time_zone_test.rb:345
```

Identical, cont.

```
def with_env_tz(new_tz = 'US/Eastern')
  old_tz, ENV['TZ'] = ENV['TZ'], new_tz
  yield
ensure
  old_tz ? ENV['TZ'] = old_tz : ENV.delete('TZ')
end
```


Maybe useful?

Maybe refactor?

Answer:
YES!

Similar (across files!)

```
#) Similar code found in :block (mass = 344)
rails/activesupport/test/xml_mini/nokogiri_engine_test.rb:6
rails/activesupport/test/xml_mini/nokogirisax_engine_test.rb:6
```

Similar with diff!

```
#) Similar code found in :block (mass = 344)
A: rails/activesupport/test/xml_mini/nokogiri_engine_test.rb:6
B: rails/activesupport/test/xml_mini/nokogirisax_engine_test.rb:6

require("abstract_unit")
require("active_support/xml_mini")
require("active_support/core_ext/hash/conversions")
# Skip nokogiri tests
A: class NokogiriEngineTest < ActiveSupport::TestCase
B: class NokogiriSAXEngineTest < ActiveSupport::TestCase
    include(ActiveSupport)

    def setup
      @default_backend = XmlMini.backend
A:   XmlMini.backend = "Nokogiri"
B:   XmlMini.backend = "NokogiriSAX"
    end
...

```

Fuzzy?

Given This Code

```
def a
  f1; f2; f3; f4; f5; f6
end

def b
  f2; f3; f4; f5; f6
end

def c
  f2; f3; f4; f5; f6; f7
end
```

Fuzzy Diffs, pt 1

```
#) Similar code found in :block (mass = 160)
rails/actionpack/test/controller/resources_test.rb:798
rails/actionpack/test/controller/resources_test.rb:842 (FUZZY)
rails/actionpack/test/controller/resources_test.rb:855 (FUZZY)
rails/actionpack/test/controller/resources_test.rb:868 (FUZZY)
```


Fuzzy Diffs, pt 2

```
def test_resource_has_only_show_action
  with_routing do |set|
    set.draw do
      resources :products, :only => :show
    end

    assert_resource_allowed_routes('products', ...)
    assert_resource_allowed_routes('products', ...)
  end
end
```

Fuzzy Diffs, pt 3

```
def test_resource_has_only_update_action_and_named_route
  with_routing do |set|
    set.draw do
      resources :products, :only => :update
    end

    assert_resource_allowed_routes('products', ...)
    assert_resource_allowed_routes('products', ...)

    assert_not_nil set.named_routes[:product]
  end
end
```

flay rails

72,947

flay --fuzzy rails

82,411!!

(+9,464)

Thank You