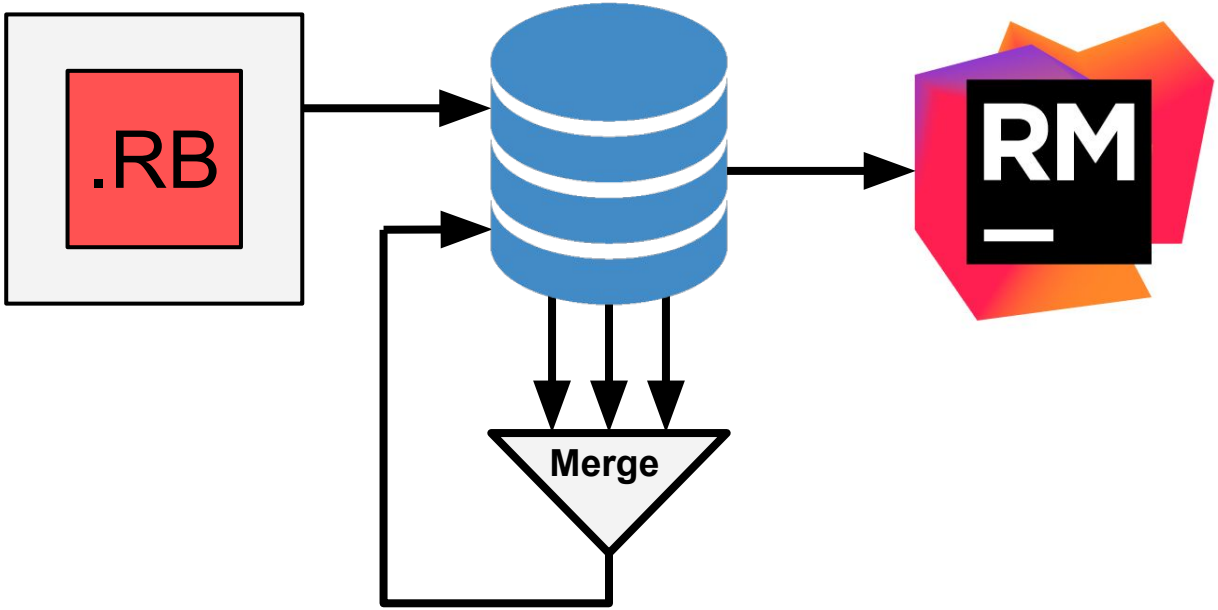
Динамический вывод типов на основе статистики запусков для языка Ruby - 2

Студент: Михаил Чернявский
Руководитель: Валентин Фондаратов

В целом о проекте

Хотим в runtime запоминать какую-то полезную информацию о вызовах методов и научить IDE использовать её для улучшения вывода типов при статическом анализе.

Что было



Простой пример

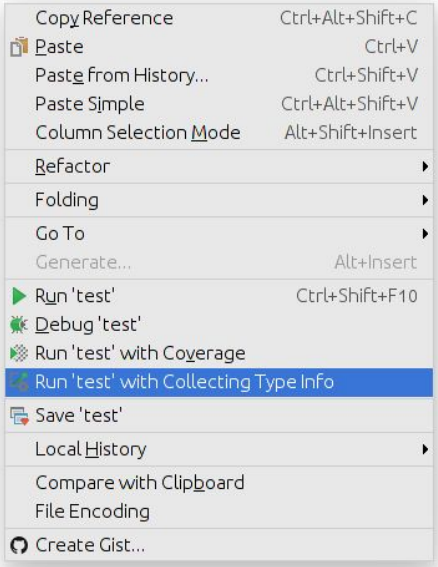
```ruby
my_class_def = <<-EOS
  class MyClass
    def foo(a, b=42, *c, d:, e: 42, **f, &blk)
      42
    end
  end
EOS

Object.class_eval(my_class_def)

my_class_instance = MyClass.new
my_class_instance.foo(42, d: 42)

my_class_instance.
```

```
__getobj__
__id__                  BasicObject
__send__()              BasicObject
__setobj__(obj)
equal?(other)           BasicObject
 foo                            call
instance_eval()         BasicObject
instance_exec(*arg…     BasicObject
method_missing(sym…     BasicObject
method(sym)                  Object
Press again to see all declared names >>
```



Простой пример

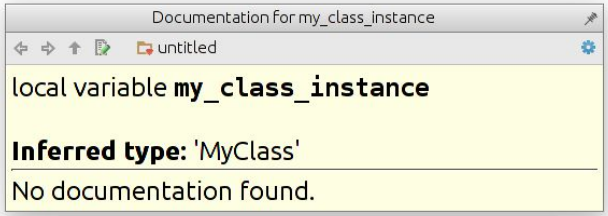
```
my_class_def = <<-EOS
  class MyClass
    def foo(a, b=42, *c, d:, e: 42, **f, &blk)
      42
    end
  end
EOS

Object.class_eval(my_class_def)

my_class_instance = MyClass.new
my_class_instance.foo(42, d: 42)

my_class_instance.
```

```
foo(a, b=…, *c, d, e=…… MyClass
__getobj__
__id__                  BasicObject
__send__()              BasicObject
__setobj__(obj)
equal?(other)           BasicObject
instance_eval()         BasicObject
instance_exec(*arg…     BasicObject
method_missing(sym…     BasicObject
method(sym)                  Object
pretty_print(q)  PP::ObjectMixin
Press again to see all declared names >>
```

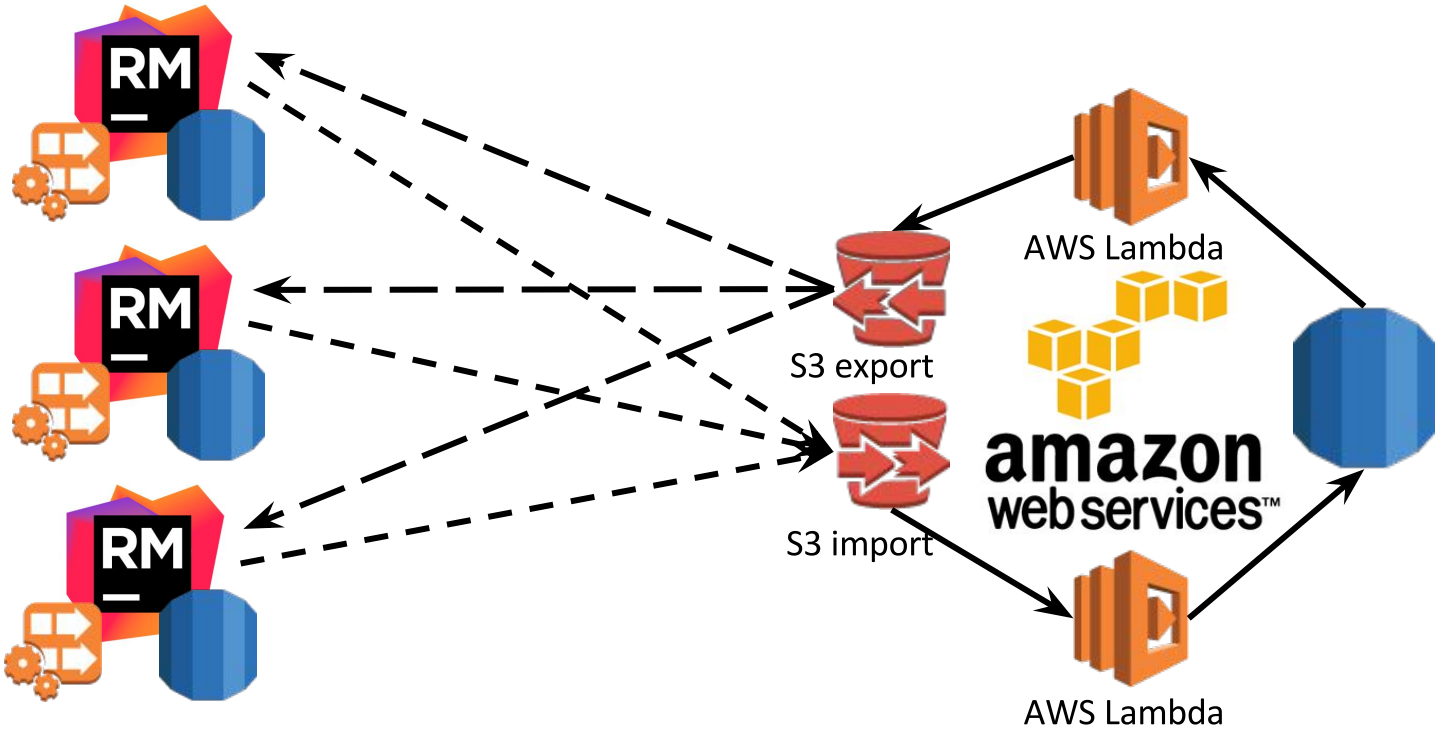


Как улучшить?

Будем шарить собранную информацию между юзерами:

- Сами проаннотируем популярные библиотеки
- Юзеры будут присылать нам свою статистику
- Выложим собранные данные всеобщее скачивание

Что получилось

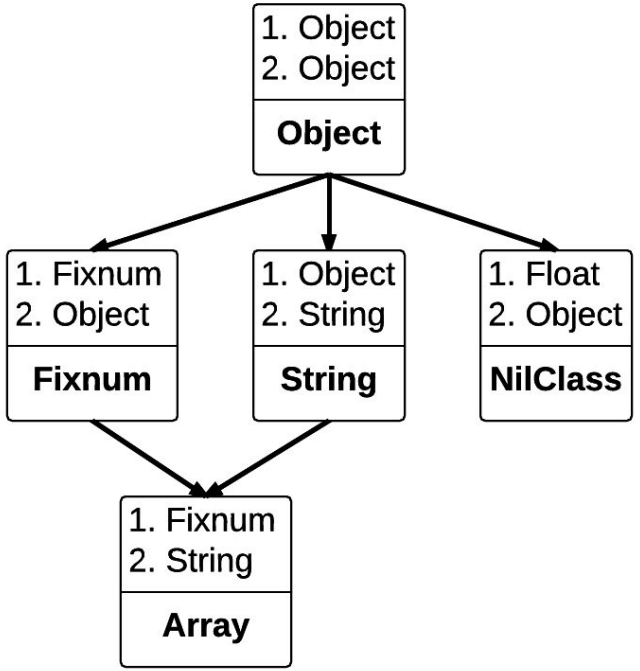


Кеширование запросов к статистике

При обращении к базе:

- Достаем всю инфомацию о методе
- Строим граф
- Кладем в кеш (Guava Cache)

Теперь храним множество таких графов в кеше и работаем с ними.



Спасибо за внимание!