

Fallacy of Composition and Ruby

RubyKaigi2016 After Party LT
Masafumi OKURA

pp myself

- ・ 大倉雅史 (Masafumi OKURA)
- ・ Rubyist (I hope you know this)
- ・ Vimmer
- ・ フリーランス (Freelancer)
- ・ 11月ぐらいからお仕事募集中 (Looking for a job)

Economics

This is  class





Fallacy of Composition

Correct in micro
Incorrect in macro

Combination of good
things become bad

In economics

(^o^) I'll save money and have better life!

$(\sim \ominus \sim)$ 。 o(Hey wait, what about others)

$$|\text{Bank}| \quad \mathcal{L}(\cup) \quad \neg \quad \equiv$$

(😊) ➡ Haha, others do that as well and
economy is getting bad because people
don't spend much money

うわああ

あああ

In Ruby

```
1 unless some_long_condition
2   some_long_name_object.
   some_long_name_method_with_ar
   gs(arg1, arg2)
3 end
```

RuboCop



- `LineLength < 80`
- `IfUnlessModifier = true`
- Notice that those two sometimes conflict

When you combine
those without thinking...

```
1 some_long_name_object.  
  some_long_name_method_with_ar  
  gs(arg1, arg2) \  
2    unless some_long_condition
```

```
1 some_long_name_object.  
  some_long_name_method_with_ar  
  gs(arg1, arg2) \
```

2 **unless** some_long_condition





Lessons

Don't apply multiple (correct)
rules at the same time
without thinking carefully.

Fallacy of Composition
exists in our daily lives