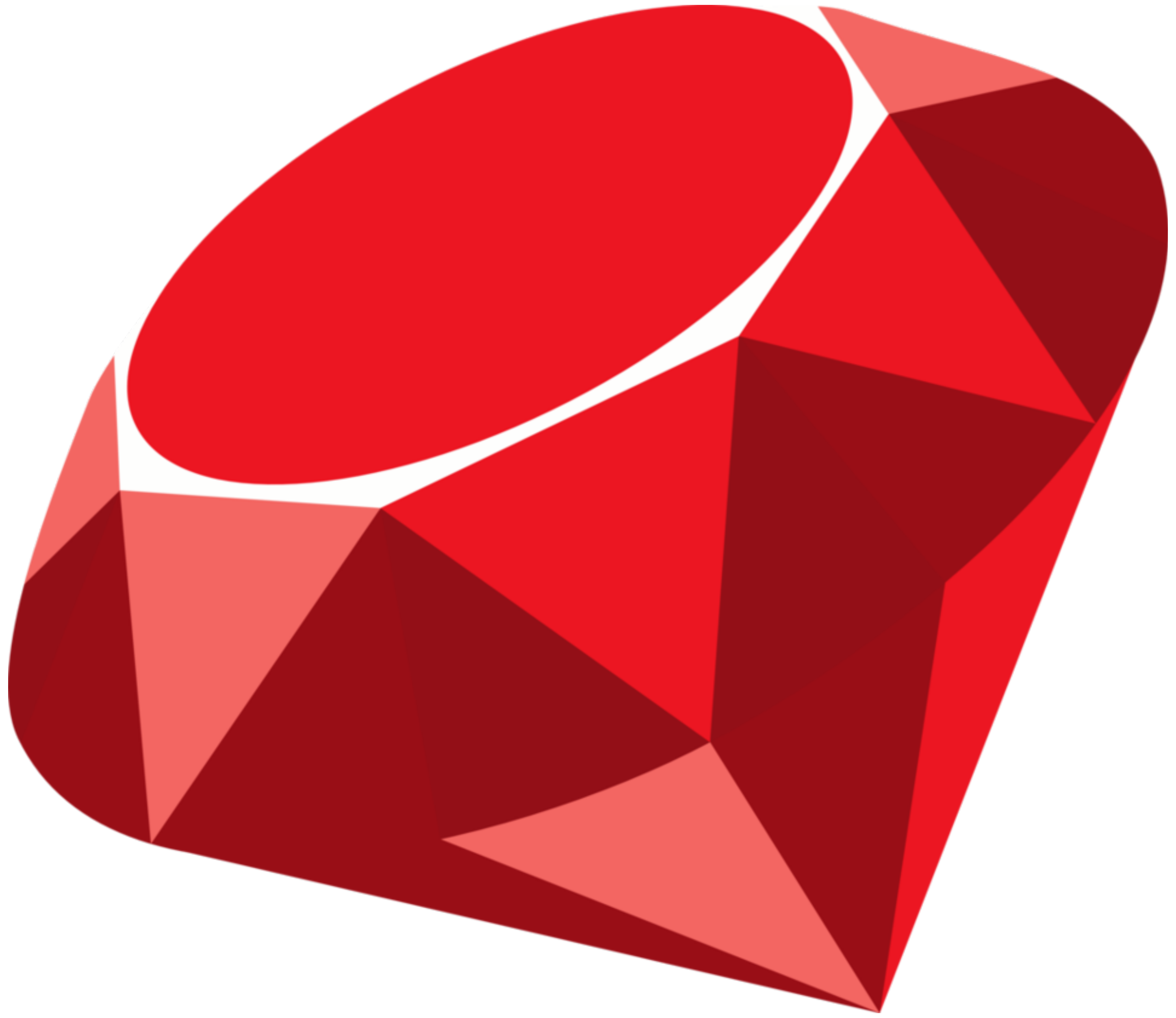
Ruby

Designing and Maintaining Software (DAMS)

Louis Rose

Ruby in a nutshell



Open-source

Influenced by Perl, Smalltalk, Eiffel, Ada and Lisp

Dynamic

Purely object-oriented

Some elements of functional programming

https://www.ruby-lang.org/en/about
https://en.wikipedia.org/wiki/Duck_typing

Ruby in a nutshell

Open-source

Influenced by Perl, Smalltalk, Eiffel, Ada and Lisp

Dynamic

Purely object-oriented

Some elements of functional programming

Duck-typed

```
class Numeric
  def plus(x)
    self.+(x)
  end
end

y = 5.plus(6)
```

https://www.ruby-lang.org/en/about
https://en.wikipedia.org/wiki/Duck_typing

Ruby in a nutshell

```ruby
5.times do
  print "Hello!"
end
```

Open-source

Influenced by Perl, Smalltalk, Eiffel, Ada and Lisp

Dynamic

Purely object-oriented

Some elements of functional programming

Duck-typed

https://www.ruby-lang.org/en/about
https://en.wikipedia.org/wiki/Duck_typing

Ruby in a nutshell

```
search_engines =
  %w[Google Bing].map do |e|
   "www." + e.downcase + ".com"
  end
```

Open-source

Influenced by Perl, Smalltalk, Eiffel, Ada and Lisp

Dynamic

Purely object-oriented

Some elements of functional programming

Duck-typed

https://www.ruby-lang.org/en/about
https://en.wikipedia.org/wiki/Duck_typing

Ruby in a nutshell

Open-source

Influenced by Perl, Smalltalk, Eiffel, Ada and Lisp

Dynamic

Purely object-oriented

Some elements of functional programming

Duck-typed

https://www.ruby-lang.org/en/about
https://en.wikipedia.org/wiki/Duck_typing

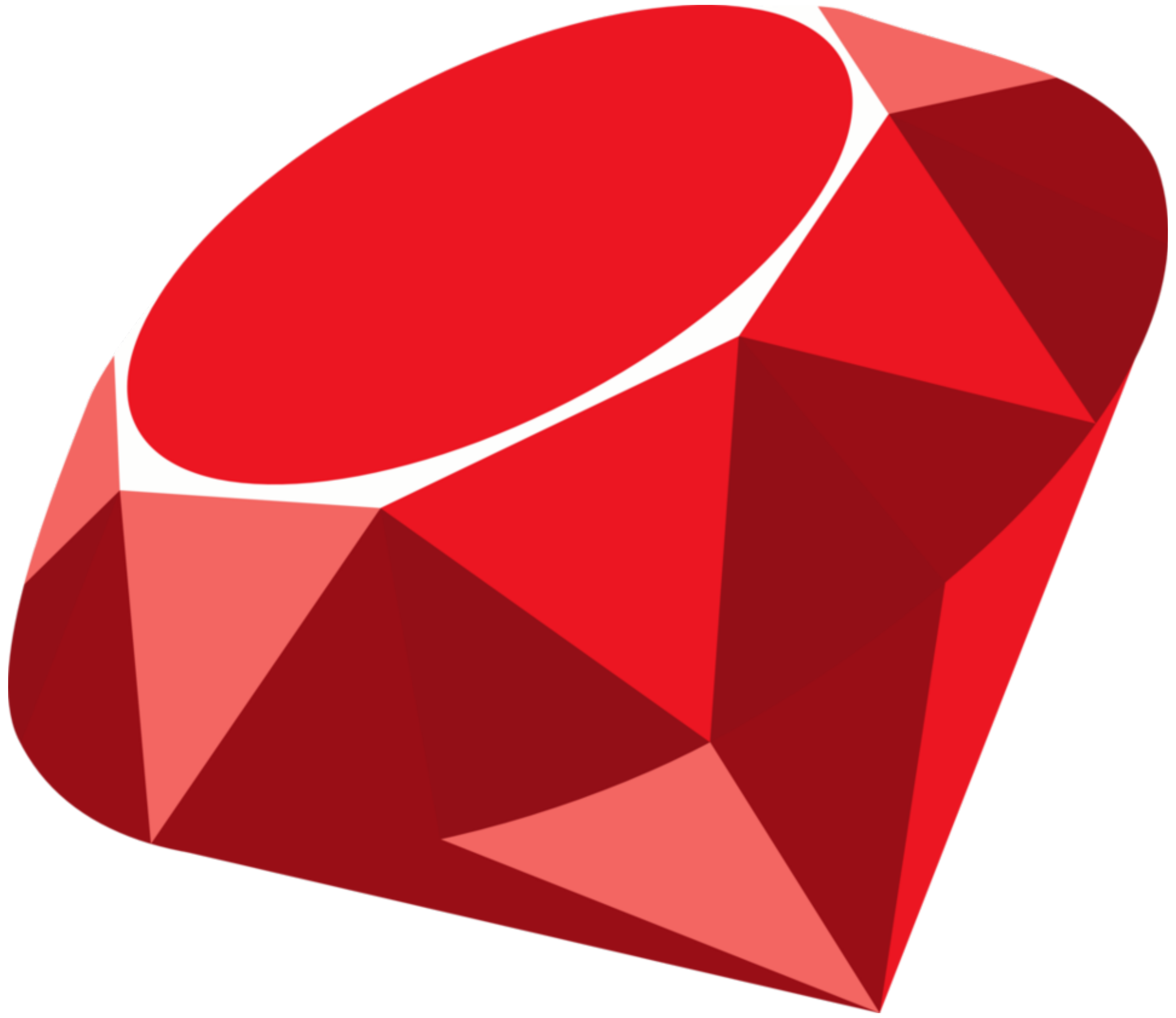
```
class Duck
  def quack
    "Quack!"
  end
end

class Person
  def quack
    "Hello!"
  end
end
```

```
def startle(duck)
  duck.quack
end

donald = Duck.new
john = Person.new
startle(donald)
startle(john)
```

Why Ruby for DAMS?



Open-source

Influenced by Perl, Smalltalk, Eiffel, Ada and Lisp

Dynamic

Purely object-oriented

Some elements of functional programming

Duck-typed

Ruby vs Java

- Both are memory-managed

Differences

- Ruby is dynamically typed. Java is statically typed
  - No casting
  - No type declarations
  - Variables are untyped
- Ruby is interpreted. Java is compiled
- Ruby is pure OO. Java is “sort of” OO

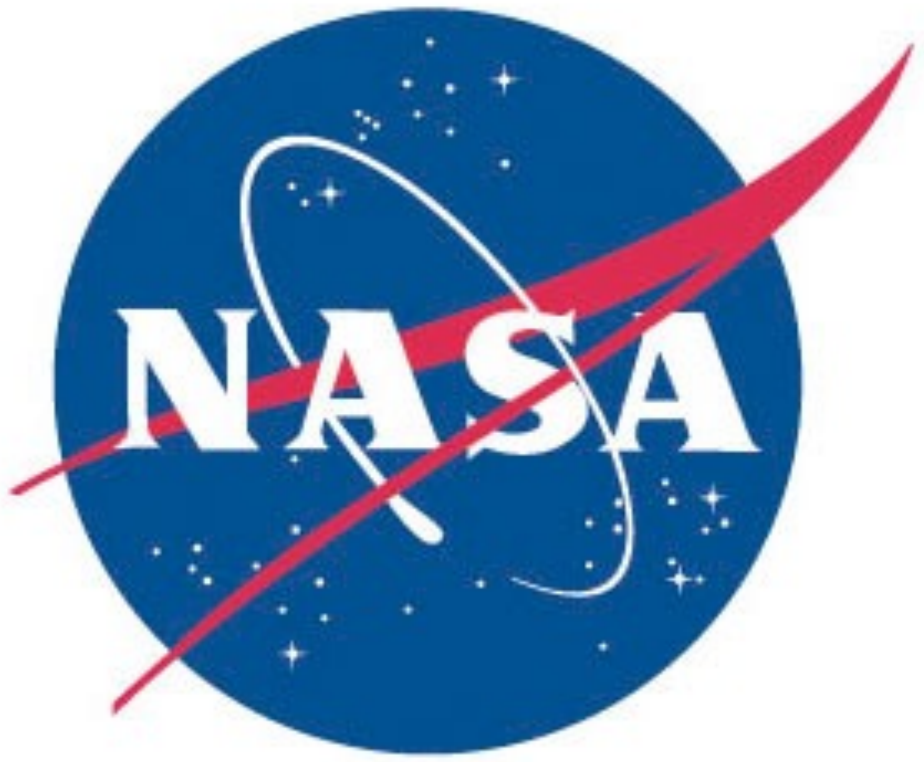
Ruby vs Python

- Both are pure OO
- Both are dynamically typed
- Both have a REPL (irb in Ruby)
- Syntax is similar

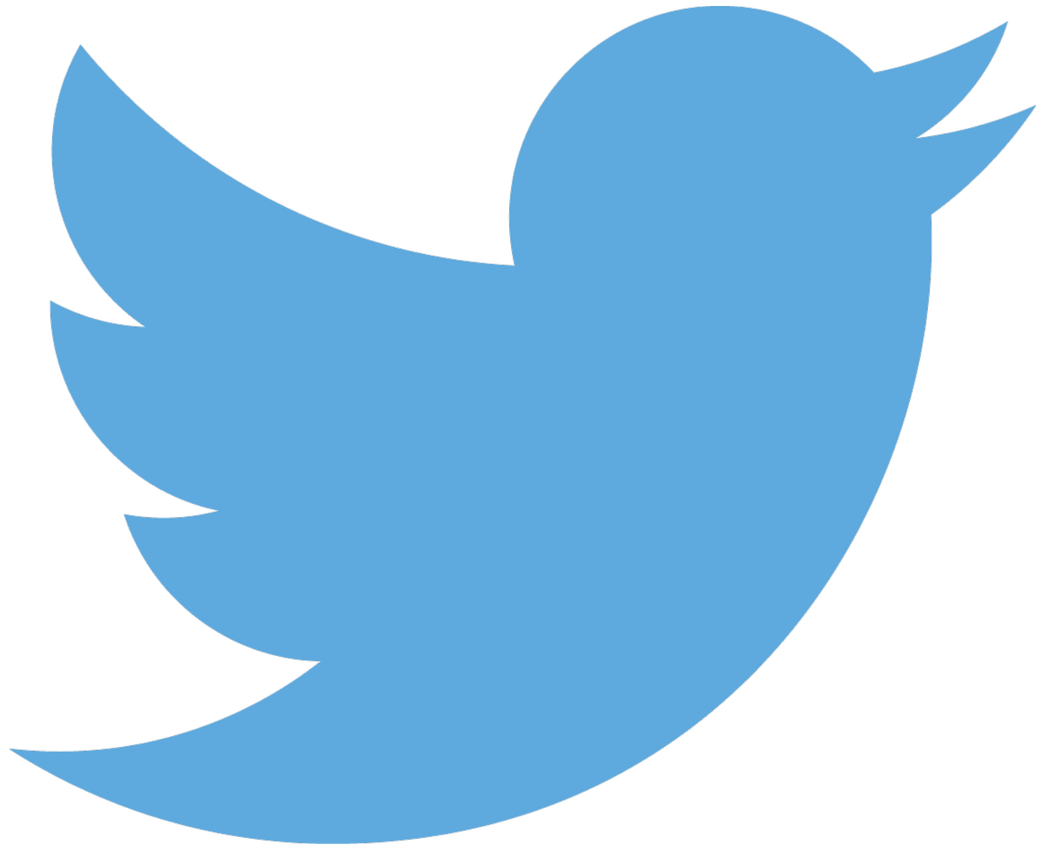
Differences

- Ruby built-in types can be extended
- Ruby makes many OO patterns very easy to implement
- Ruby has more natural function composition

Who uses Ruby?



Who uses Ruby (on Rails)?



Resources

The DAMS Ruby tutorial

Official website:
https://www.ruby-lang.org

Poignant Guide to Ruby:
http://mislav.uniqpath.com/poignant-guide/