

Ruby is an acceptable Lisp



Vitaly Kushner
astrails.com

Ruby is **better**
then Lisp

Ruby is better
then Lisp

sometimes ;-)

Language



<http://www.flickr.com/photos/jantik/254695220/>

Power



<http://www.flickr.com/photos/lrargerich/4187318085/>

Blub Language Paradox

Paul Graham

<http://paulgraham.com/avg.html>

Is **Lisp** the most
powerful?

NO

It depends

- *syntax*
- *linguistic power*
- *domain*
- *libraries*
- *community*

Ruby

History

- Yukihiro Matsumoto (aka "Matz")
- Released in 1994
- Got known in US about 2000
- Gained momentum around 2003-2005

more powerful
than Perl
more object-oriented
than Python.

- Perl
- Smalltalk
- Eiffel
- Ada
- Lisp

- Simple consistent syntax
- Dynamically typed
- Statically scoped
- Late binding
- Single Inheritance with Mixin support

- Everything is an object
- Closures
- Garbage Collection
- Multi platform

Ruby is **Awesome**

Clean & Consistent Syntax

`var` # variable

`$var` # global variable

`@var` # instance variable

`@@var` # class variable

`CONST` # constant

`Class` # class

```
attrs = {  
  :src => "foo.img",  
  :width => 100,  
  :height => 200,  
  :class => Avatar  
}
```

```
User.find params[:id],  
  :limit => 10,  
  :order => "name"
```

```
if @project.owned_by?(@user)
  return false unless @user.admin?
end
```

```
@project.complete!
```

Higher-order functions

useless without ...

Anonymous functions

```
def x_times(x, fun)
  for i in 1..x
    fun.call(i)
  end
end
x_times(10,
  lambda { |x| puts x })
```

```
def x_times(x)
  for i in 1..x
    yield i
  end
end
x_times(10) {|x| puts x}
```

```
def x_times(x)
  for i in 1..x
    yield i
  end
end

x_times(10) do |x|
  puts x
end
```

Everything is
an Object
that you can extend

```
class Fixnum
  def x_times
    for i in 1..self
      yield i
    end
  end
end

5.x_times { |x| puts x }
```

```
5.times { |x| puts x }
```

map	{ x ... }
collect	{ x ... }
select	{ x ... }
reject	{ x ... }
find	{ x ... }
any?	{ x ... }
all?	{ x ... }
sort	{ a, b ... }

3 . megabytes

=> 3145728

10.months.from_now

=> Thu Aug 12 03:25:40 0300 2010

5.minutes.ago

=> Mon Oct 12 03:21:02 0200 2009

Closures

```
def incrementor(increment)
  proc { |x| x + increment }
end
```

```
>> i5 = incrementor(5)
```

```
=> #<Proc:0x01874a78@(irb):46>
```

```
>> i5.call(3)
```

```
=> 8
```

Compact

```
# ruby
def paidMore(amount)
  proc { |e| e.salary > amount }
end
```

```
// C#
public Predicate<Employee>
    PaidMore(int amount) {
  return delegate(Employee e) {
    return e.Salary > amount;
  }
}
```

// ruby

```
def foo(n) lambda {|i| n+=i} end
```

; lisp

```
(defun foo (n) (lambda (i) (incf n i)))
```

ruby

```
[1,2,3].map {|n| n*n }.reject {|n| n%3==1 }
```

; lisp

```
(remove-if (lambda (n) (= (mod n 3) 1))  
          (mapcar (lambda (n) (* n n))  
                  '(1 2 3)))
```

Macros

Ruby **fakes** macros
pretty well

OO +

Monkey Patching

```
class Person
  def self.defsay(sound)
    define_method("say_#{sound}") do
      puts sound
    end
  end
end

defsay :hello
defsay :hi
end
```

```
>> bob = Person.new
=> #<Person:0x185cba8>
>> bob.say_hello
hello
=> nil
>> bob.say_hi
hi
=> nil
```

```
class Plugin < ActiveRecord::Base
  validates_presence_of :name
  validates_presence_of :description
  validates_presence_of :author_id
  belongs_to :author, :class_name => "User"
  has_many :plugin_versions,
    :dependent => :destroy
  belongs_to :default_version,
    :class_name => "PluginVersion"
  acts_as_commentable
  acts_as_taggable
  ...
end
```

Magic

NoMethodError

method_missing

```
User.find_by_name_and_company(  
    "Vitaly Kushner", "Astrails")
```

Ruby rewrite

```
require 'pp'  
require 'parse_tree'  
require 'parse_tree_extensions'
```

```
def print_ast(&block)  
  pp block.to_sexp  
end
```

```
print_ast do
  puts "hello"
end
```

```
s(:iter,
  s(:call, nil, :proc, s(:arglist)),
  nil,
  s(:call, nil, :puts, s(:arglist, s(:str, "hello"))))
```

Multiple inheritance
is **EVIL :-)**

Modules

a.k.a. Mixins

```
module FlyHome
  def set_home
    @home = position
  end

  def fly_home
    fly(@home)
  end
end
```

```
class Bird < Living
  include FlyHome
  def fly(direction) ...
  def position ...
end
```

```
class Airplane < Machine
  include FlyHome
  def fly(direction) ...
  def position ...
end
```

Libraries

Rubygems

```
→ ~ X sudo gem install astrails-safe
Successfully installed astrails-safe-0.2.7
1 gem installed
Installing ri documentation for astrails-
safe-0.2.7...
Building YARD (yri) index for astrails-safe-0.2.7...
Installing RDoc documentation for astrails-
safe-0.2.7...
→ ~ X
```

Rubygems.org
12,798 gems

GitHub.com

Community

attitude

Testing

TDD

TDD

Test Driven Development

BDD

BDD

Behavior Driven Development

Pressure

TATFT

TATFT

Test All The Fucking Time

- Test::Unit
- RSpec
- Shoulda
- Cucumber
- Webrat

```
it "should be able to show media" do
  @media = stub_media
  Media.stub!(:find).and_return(@media)
  get :show, :id => @media.id
  response.should be_success
end
```

Ruby is Better then Lisp

Q & A



astrails

@astrails

Vitaly Kushner

astrails.com

@vkushner

References

- [Why Ruby is an acceptable LISP](#)
- [Beating The Averages](#) (Paul Graham)
- [Israel.rb](#) (ruby group)
- [Astrails blog](#)