There is a Ruby in your Browser
CREATE, COLLABORATE AND DEPLOY FROM IDEA TO PRODUCTION
We are going to talk a lot about languages today...
But first... let's go back in time...
1980 - 1988
CERN, John Poole's ICS

Hypertext
Real-time RPC
Computer Networking
March, 1989
Hypertext + TCP + DNS
World Wide Web
Dec, 1990
WorldWideWeb

First 'navigator'
GNU Operating System - Free Software Foundation

Welcome to the GNU Project web server, www.gnu.org. The GNU Project was launched in 1984 to develop a complete UNIX style operating system which is free software; the GNU system (GNU) is a recursive acronym for &ldquo;GNU's Not UNIX&rdquo; and its pronounced &ldquo;Jew-nux&rdquo;. Variants of the GNU operating system, which use the kernel Linux, are now widely used, though these systems are often referred to as &ldquo;Linux&rdquo; or &ldquo;Linux derivatives&rdquo;.

This is also the web site of the Free Software Foundation (FSF). FSF is the principal organizational sponsor of the GNU Project. FSF receives very little funding from corporations or grant-making foundations. We rely on support from individuals like you who support FSF’s mission to preserve, protect and promote the freedom to use, study, copy, modify, and redistribute computer software, and to defend the rights of Free
1991-1993
Spreading the web

Outside Europe
Welcome to the Mosaic for Microsoft Windows Home Page. Mosaic is a World Wide Web client that was developed at the National Center for Supercomputing Applications on the campus of The University of Illinois in Urbana-Champaign.

Search Our Space

News and Announcements

- Version 2.0.0
- New! Win32s Information
- World Wide Web Conferences Information
CERN Releases WorldWideWeb source code
Netscape Navigator 1.0
1995
Web needs to become more dynamic...
Netscape hired Brendan Eich
To implement Scheme* 
In the browser
"Why the web needed 2 programming languages?"
Java, Scripting Language
Requirement: Syntax similar to Java
Prototype language in 10 days...
"We aimed to provide a 'glue language' for the web designers and part-time programmers"
Alan Cooper
Created VBScript
Script language for the whole Windows ecosystem*
As we are talking about languages...
1995
Ruby was born...
"People want to express themselves when they program. They don't want to fight with the language" (2001, Matz)
"You can do everything in assembler, but no one wants to program in assembler anymore."

(2003, Matz)
We <3 Ruby

but...
"...no one wants to program in assembler anymore" (2016, Elia Schito)
Why Javascript

???
How to organize your code in javascript
Everything in JS is time-sensitive* (and could be incompatible)
Frameworks, libraries

Everything change from little time to time
Packaging and delivery
npm
(b) (gate)

bower

componentjs

jspsm

volojs

spmjs

cpm

jamjs
leffpad
Options are not but Noise and distraction
Why not stick with Ruby?
Decaf Browser:

WebKit fork that runs Ruby (native)

gh: timahoney/decaf
Not the best alternative
As it's hard to convince every browser...
Why not write Ruby in JS?
Opal is ~95% RubySpec ready
Ruby runtime

+ Transpiler
Native:
Arrays, numbers, strings*, boolean, self -> this
puts "hello rubyconf"

# becomes:
(function(Opal) {
    var self = Opal.top, $scope = Opal, nil = Opal.nil, $breaker = Opal.breaker, $slice = Opal.slice;

    Opal.add_stubs(['$puts']);
    return self.$puts("Hello RubyConfBr")
})(Opal);
Packaging and Dependency
$ gem install opal-*

# With rails, add to your Gemfile and:
$ bundle install

# In your code:
require 'my-awesome-framework'
Some libraries
To get started...
opal-jquery
# Gemfile

gem 'opal'
gem 'opal-jquery'

# Your code:
Document.ready? do
  alert "document is ready to go!"
end

# ...
Element.find('#header').on :click do
  puts "The header was clicked!"
end
opal-rspec
# Gemfile

gem 'opal'
gem 'opal-rspec'

# Your code:
describe User do
  it "can be created with a name" do
    expect(User.new).to_not be_persisted
  end
end
Frameworks
class Clock
  include Inesita::Component

  def initialize
    @time = Time.now
  end

  def render
    div class: 'clock' do
      text @time.strftime('%r')
    end
  end
end

$document.ready do
  Clock.new.mount_to($document['clock'])
end
React.rb
class Nav < React::Component::Base
  def render
    div{"Our Nav Bar Goes Here including a login box"}
  end
end
Perguntas?