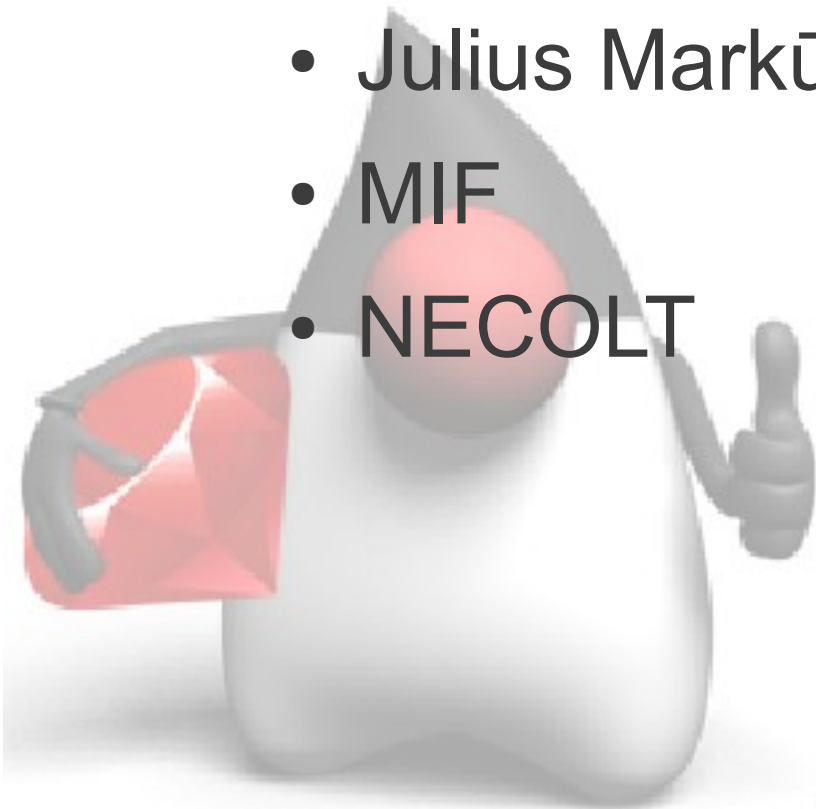
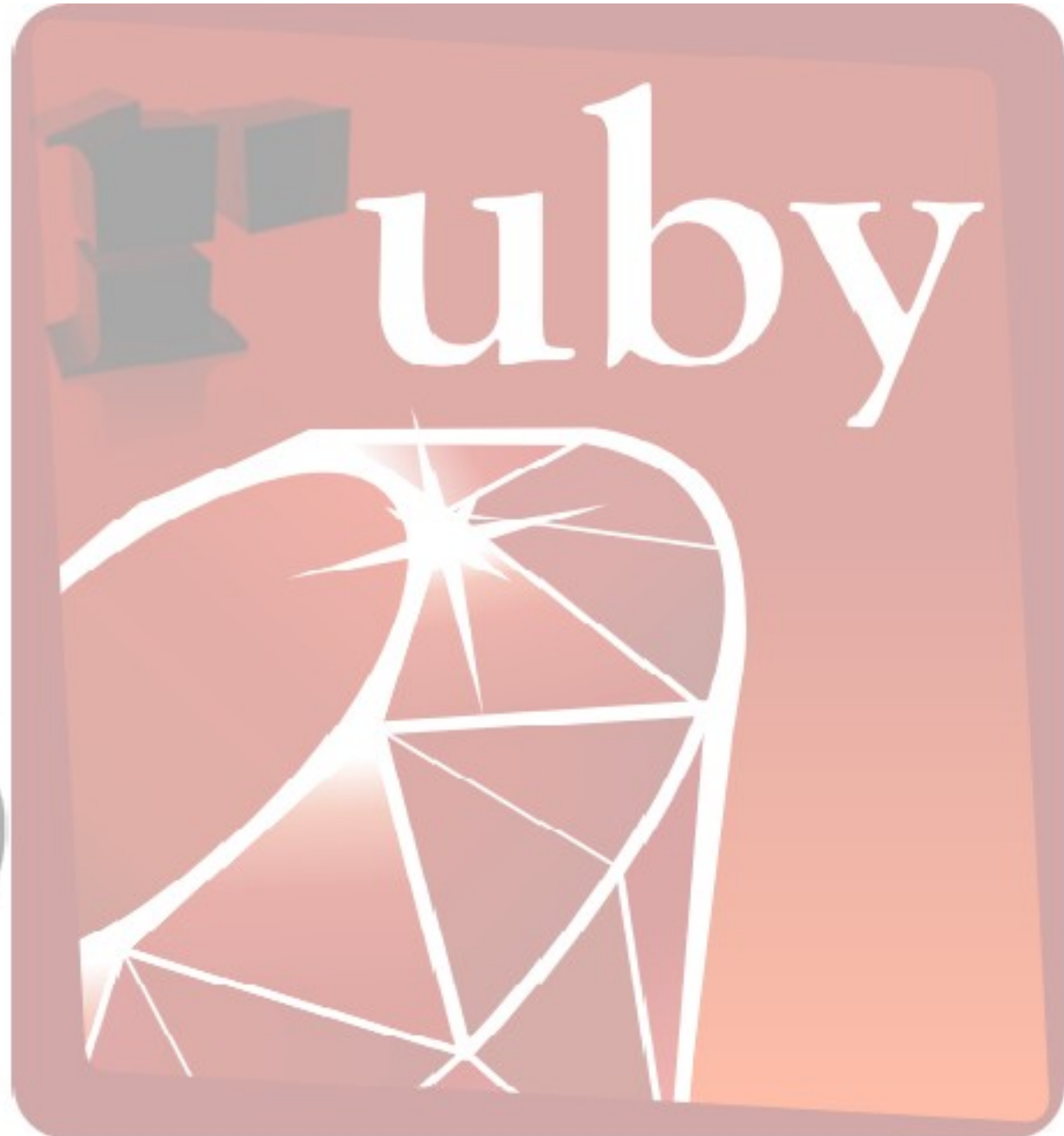


Ruby interpretatoriai

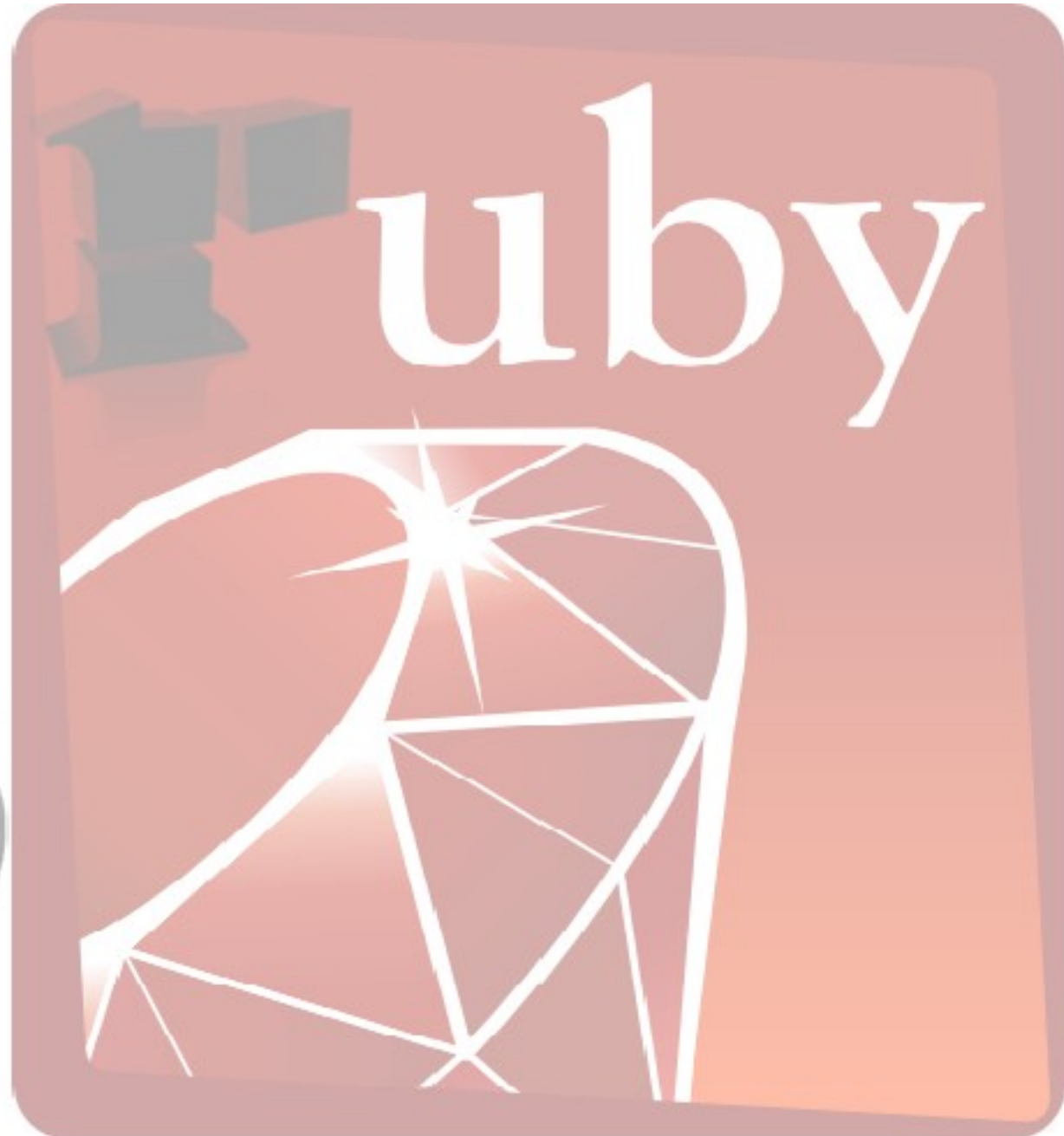
- Julius Markūnas
- MIF
- NECOLT



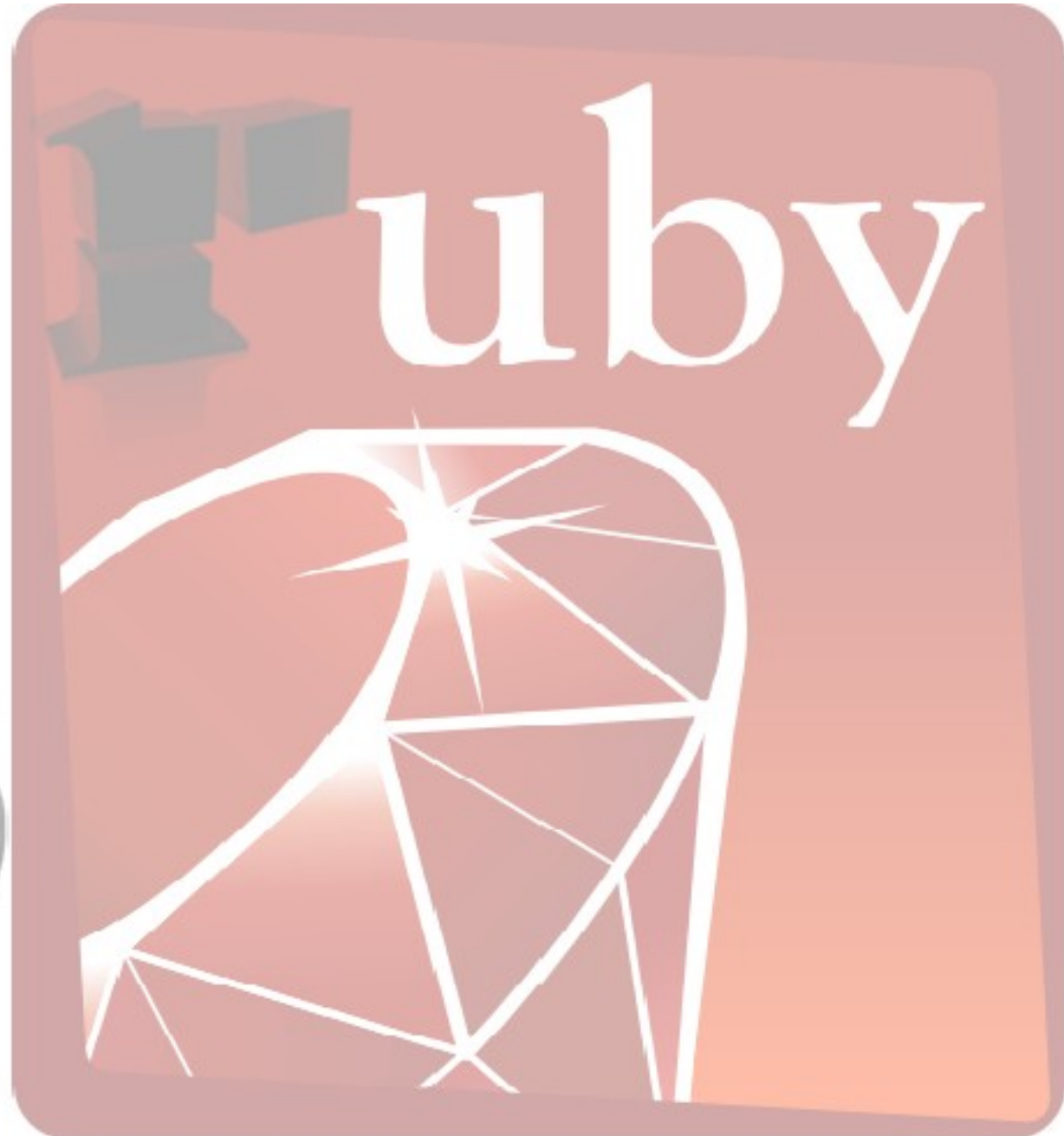
- MRI
- Rubinius
- JRuby
- MacRuby
- IronRuby
- MagLev
- Xruby



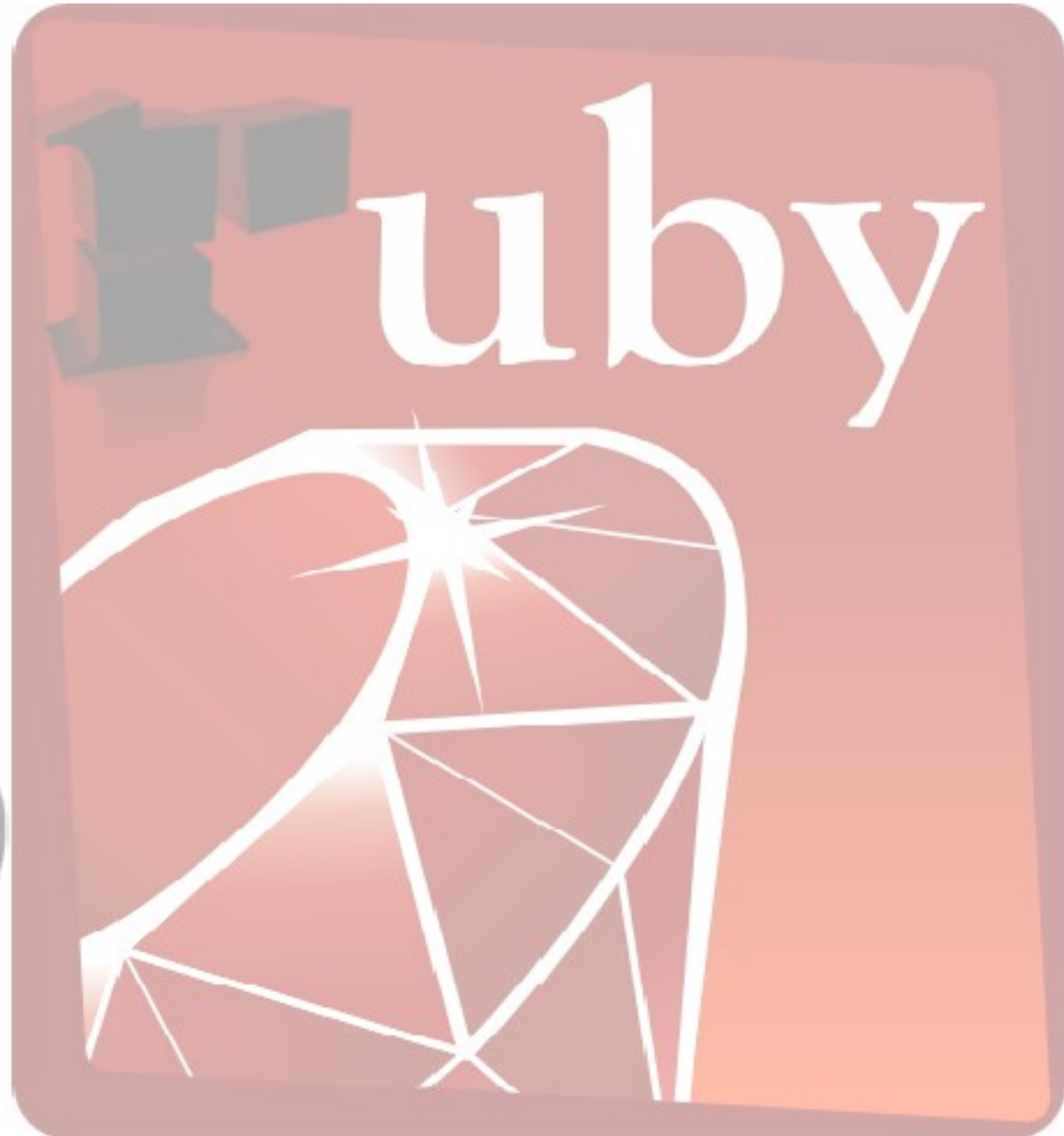
- **MRI**
- **Rubinius**
- **JRuby**
- **MacRuby**
- **IronRuby**
- **MagLev**
- **Xruby**



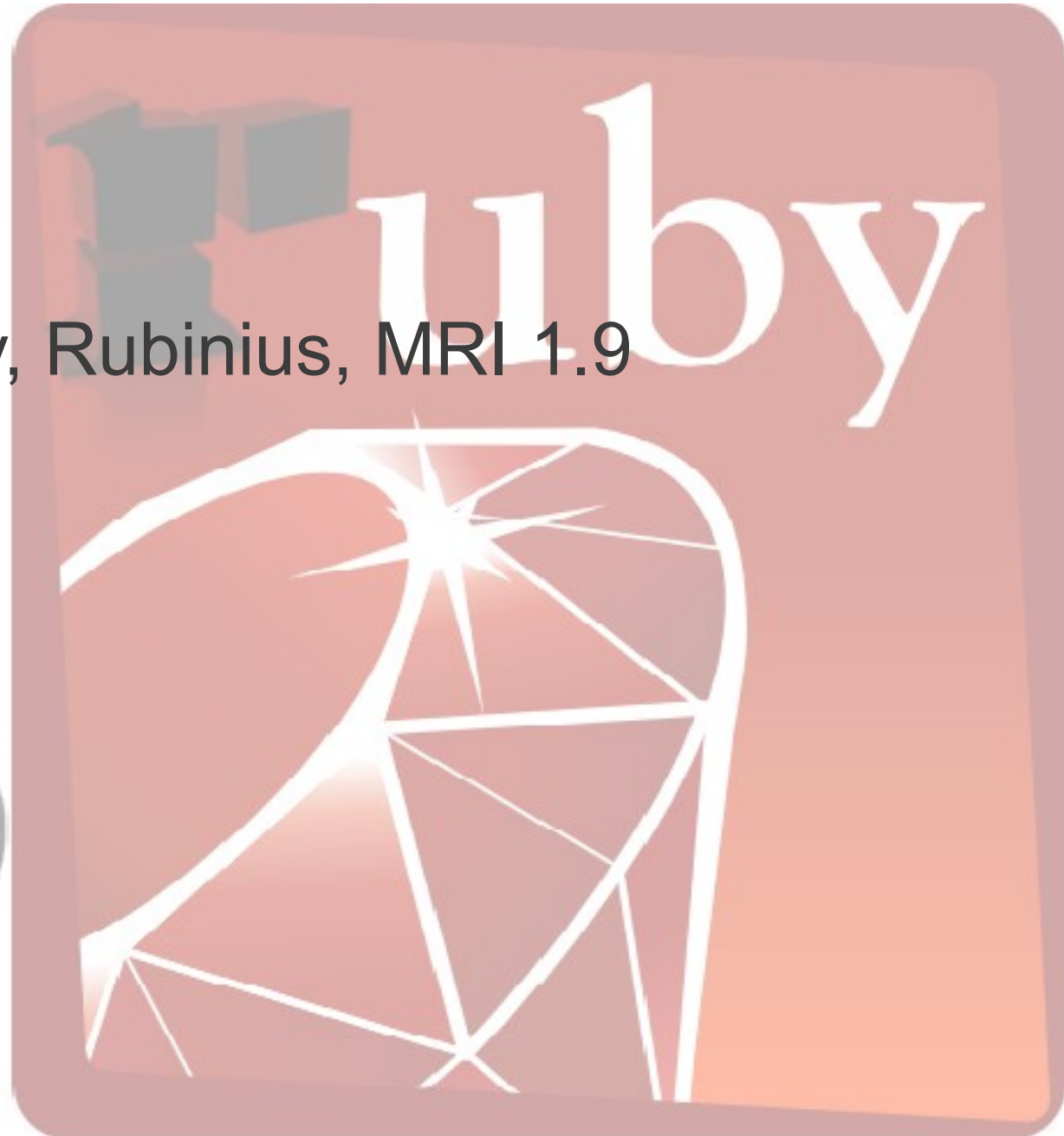
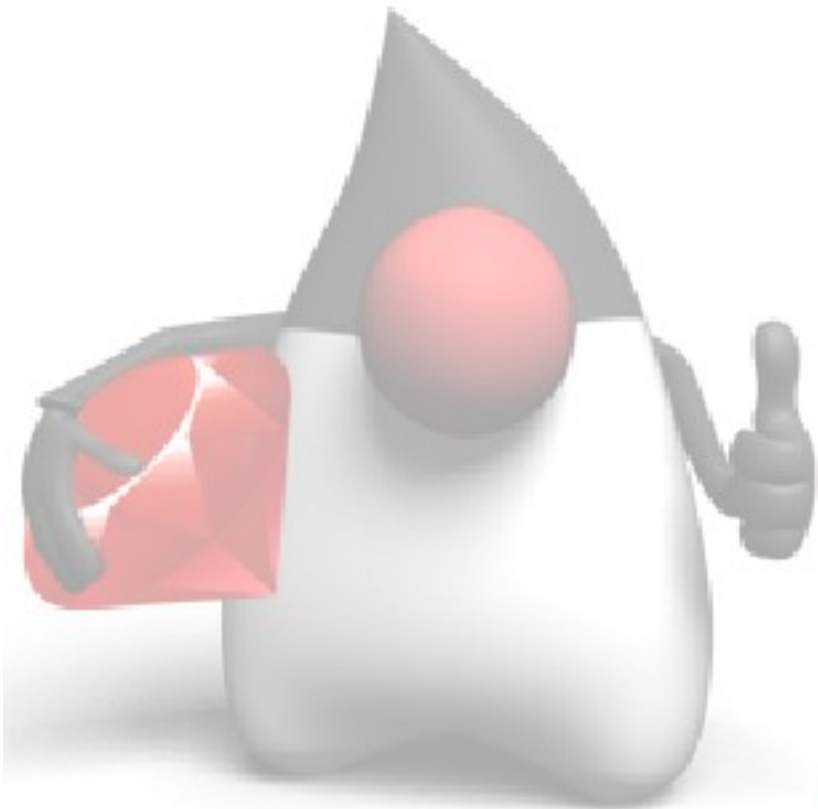
- Interpreter
- Optimizations
- GC
- Threading
- Extending ruby



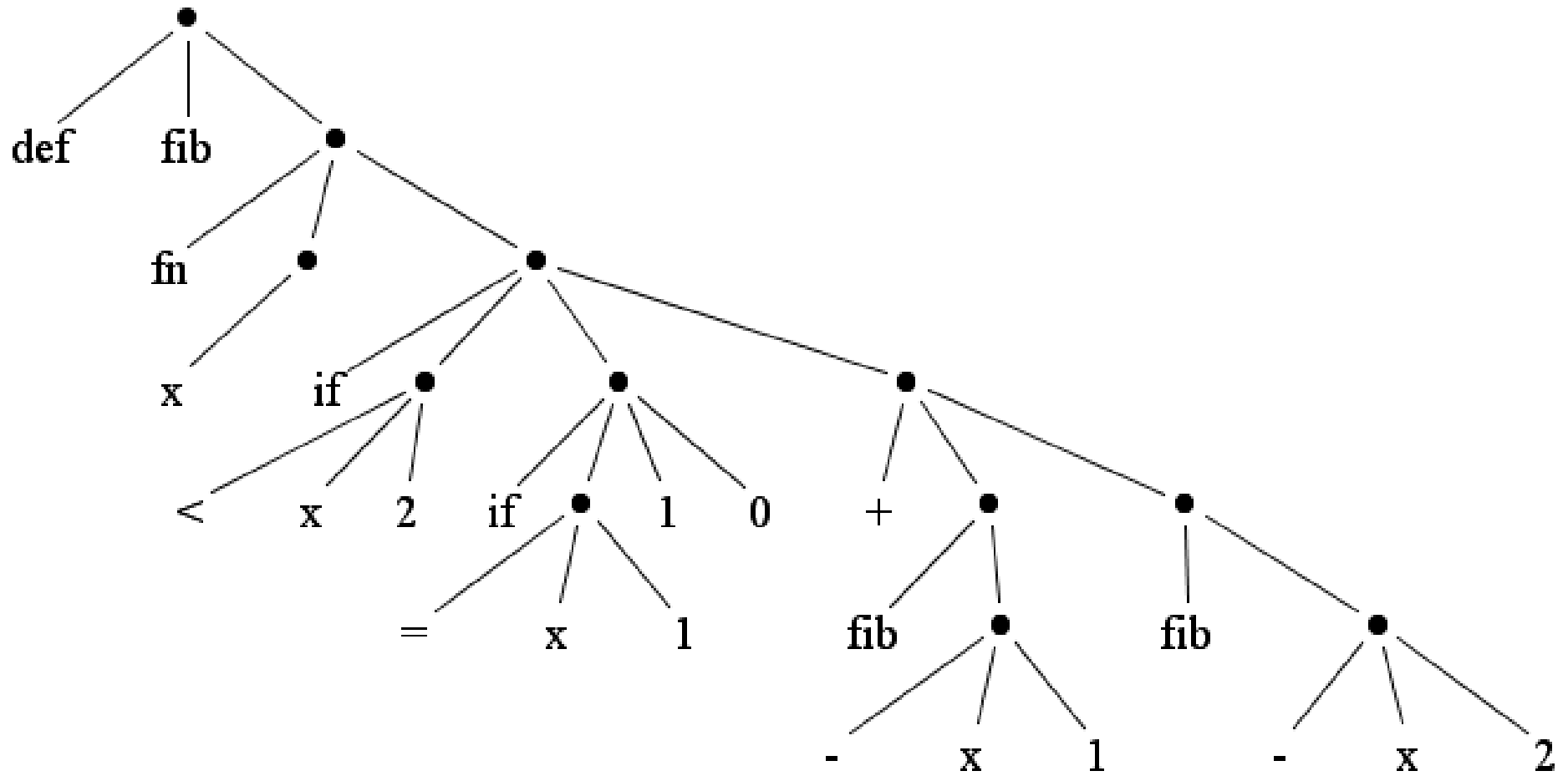
- **Interpreter**
- Optimizations
- GC
- Threading
- Extending ruby



- AST - MRI 1.8
- Bytecode - JRuby, Rubinius, MRI 1.9

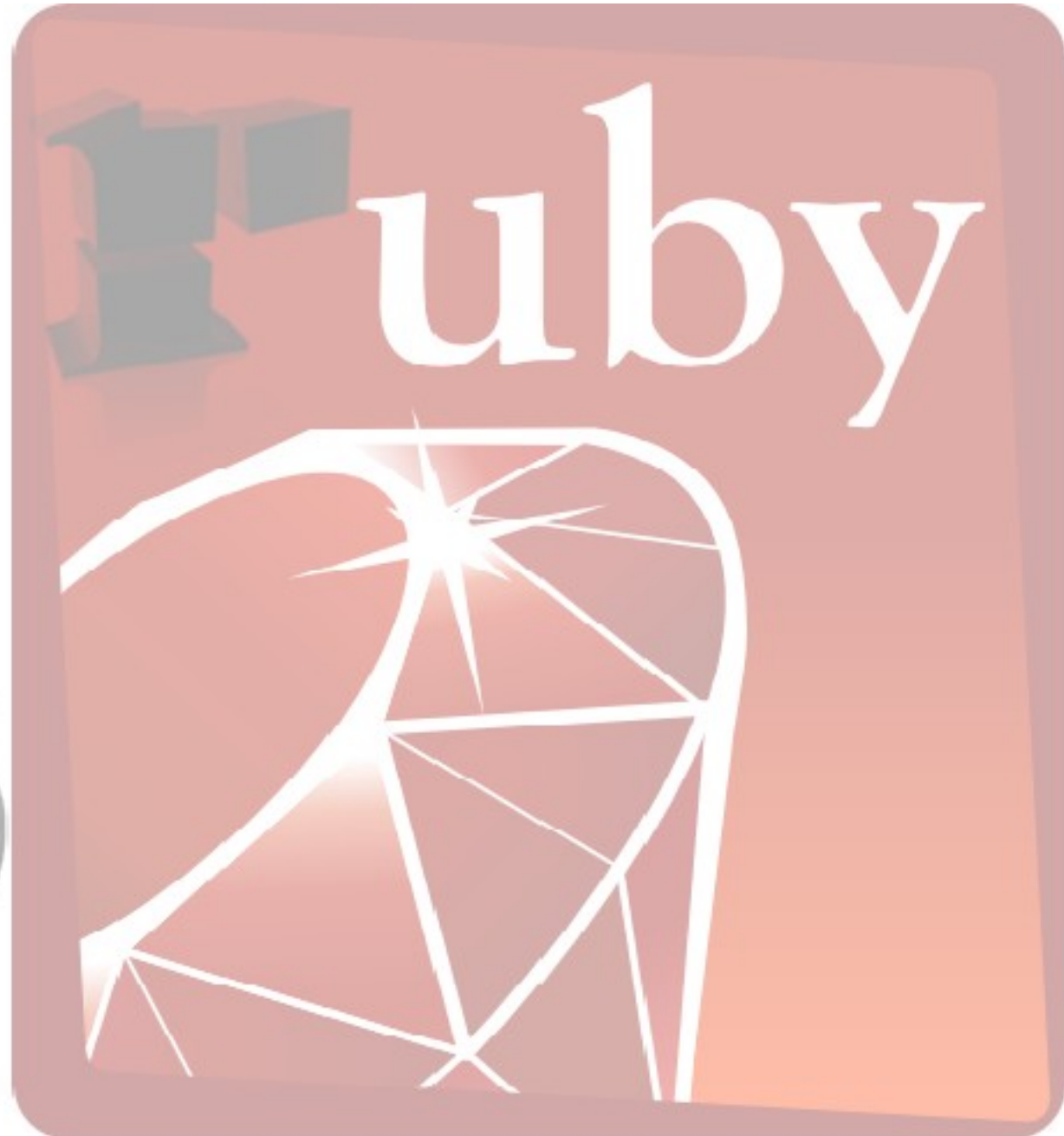


```
(def fib (fn (x) (if (< x 2) (if (= x 1) 1 0) (+ (fib (- x 1)) (fib (- x 2))))))
```

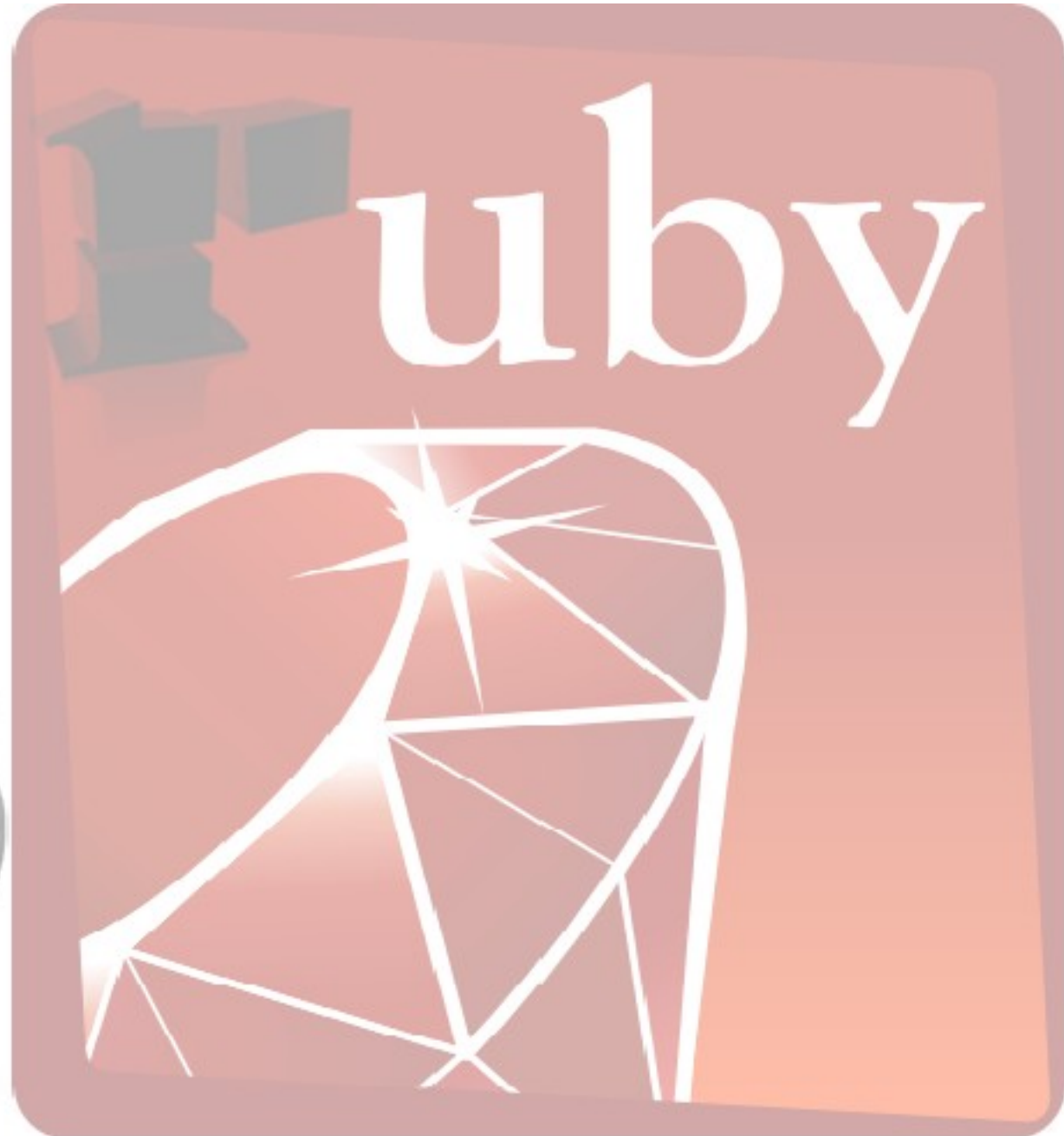
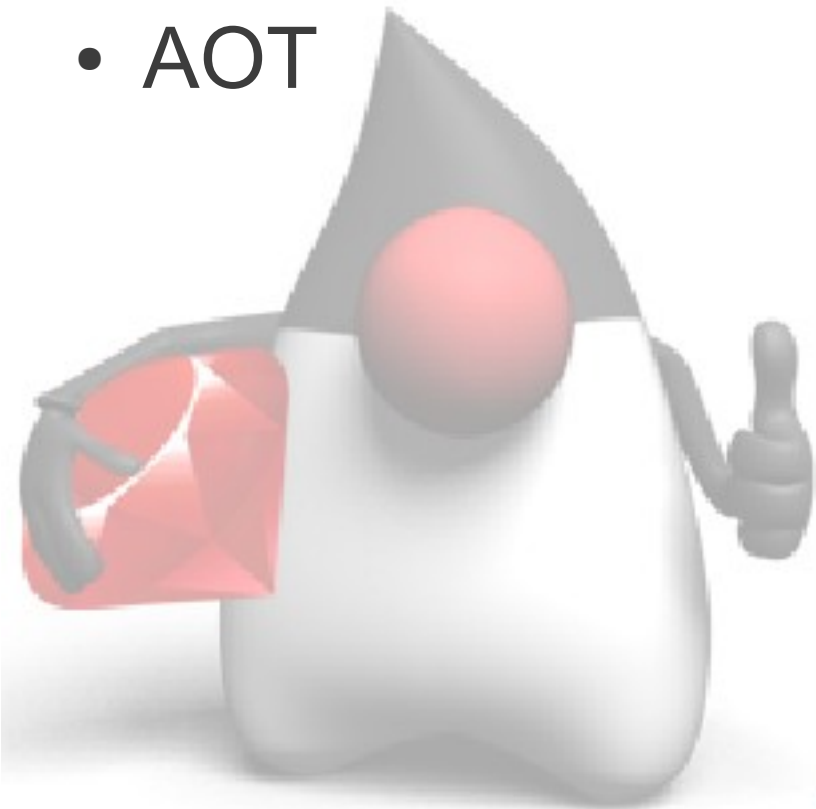



```
irb(main):009:0> puts cp.decode
0000: push_cpath_top
0001: find_const          0
0003: meta_push_0
0004: meta_push_1
0005: send_stack          :new, 2
0008: push_local         0 # n
0010: send_stack          :include?, 1
0013: goto_if_false      20
0015: push_local         0 # n
0017: ret
0018: goto                21
0020: push_nil
0021: pop
0022: push_local         0 # n
0024: meta_push_1
0025: meta_send_op_gt    :>
0027: goto_if_false      53
0029: push_self
0030: push_local         0 # n
0032: meta_push_1
0033: meta_send_op_minus :-
0035: allow_private
0036: send_stack          :fib, 1
0039: push_self
0040: push_local         0 # n
0042: meta_push_2
0043: meta_send_op_minus :-
0045: allow_private
0046: send_stack          :fib, 1
0049: meta_send_op_plus  :+
0051: goto                54
0053: push_nil
0054: ret
=> nil
```


- Interpreter
- **Optimizations**
- GC
- Threading
- Extending ruby



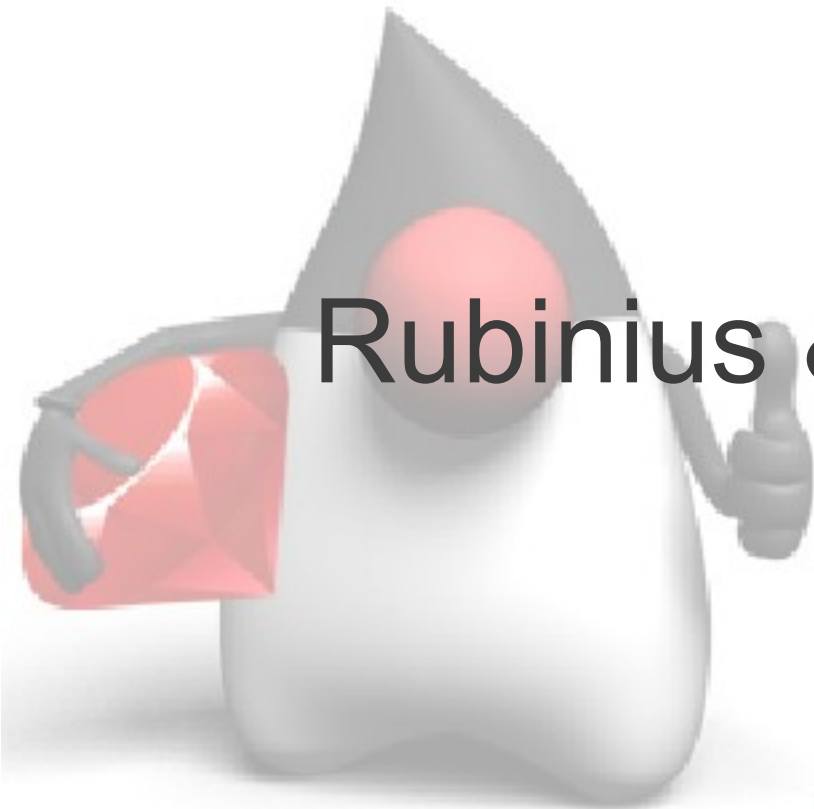
- JIT
- Method inline
- AOT



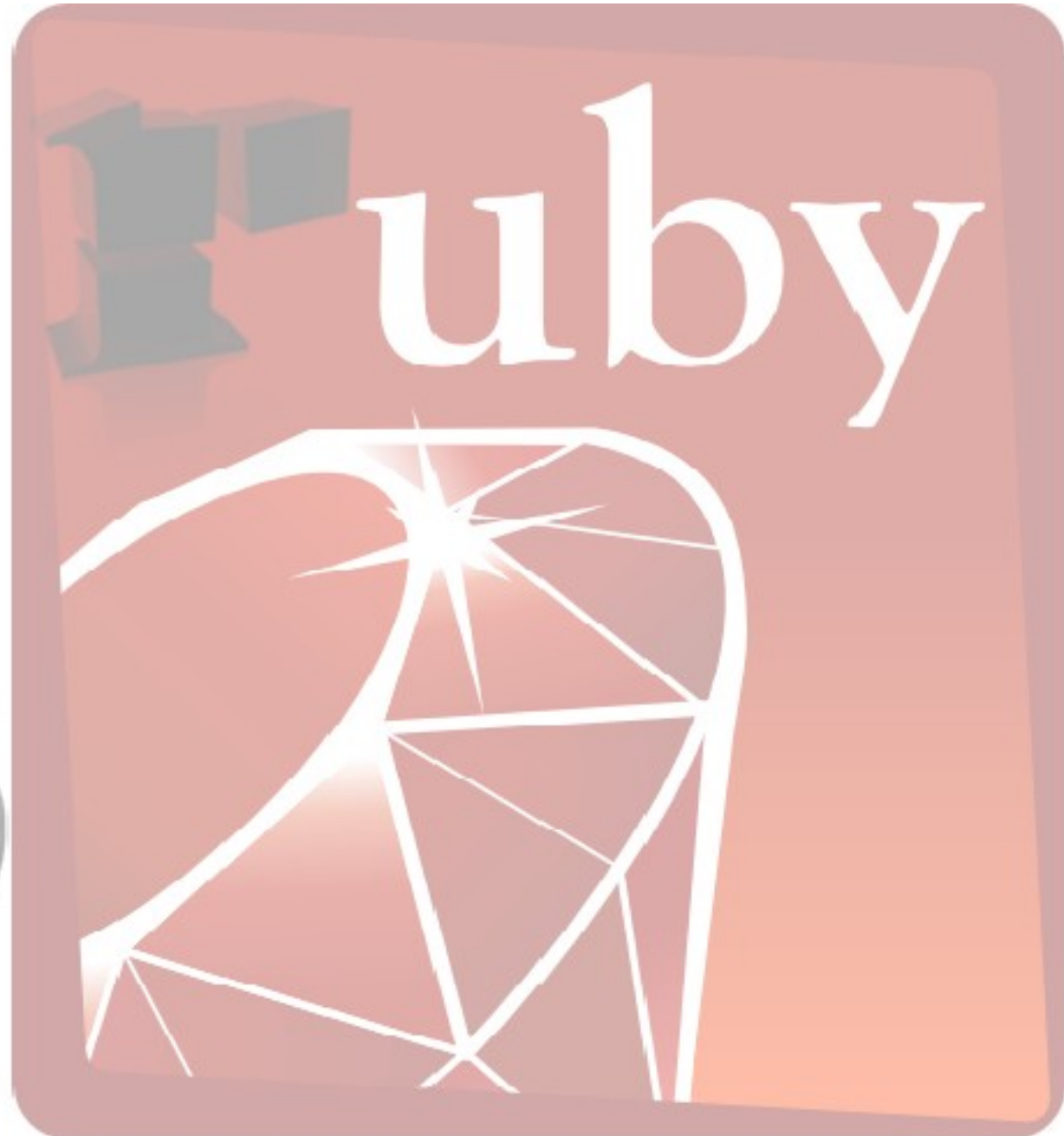
MRI :(

uby

Rubinius && JRuby (party)

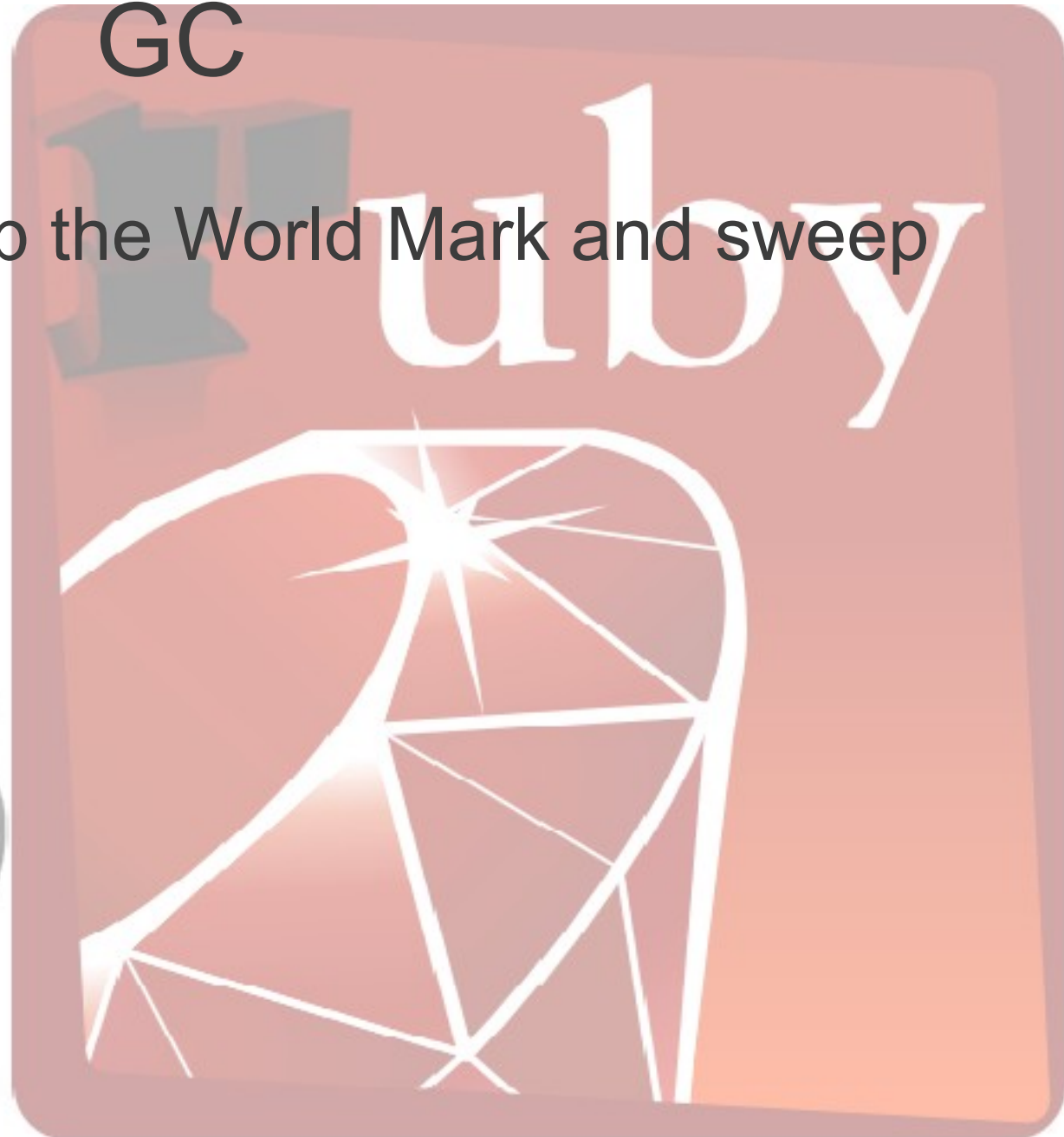
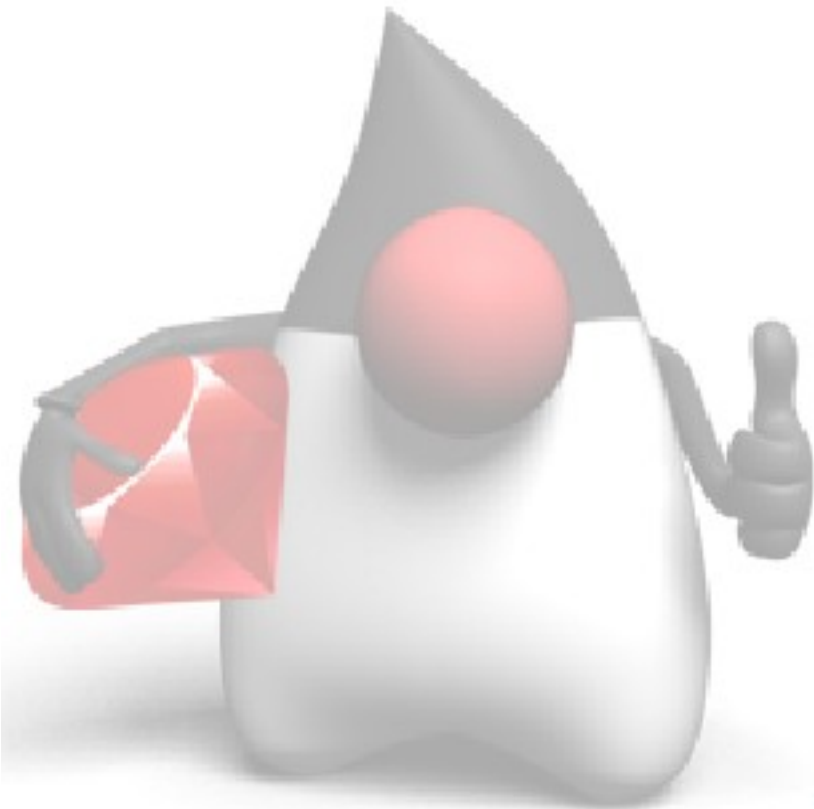


- Interpreter
- Optimizations
- **GC**
- Threading
- Extending ruby

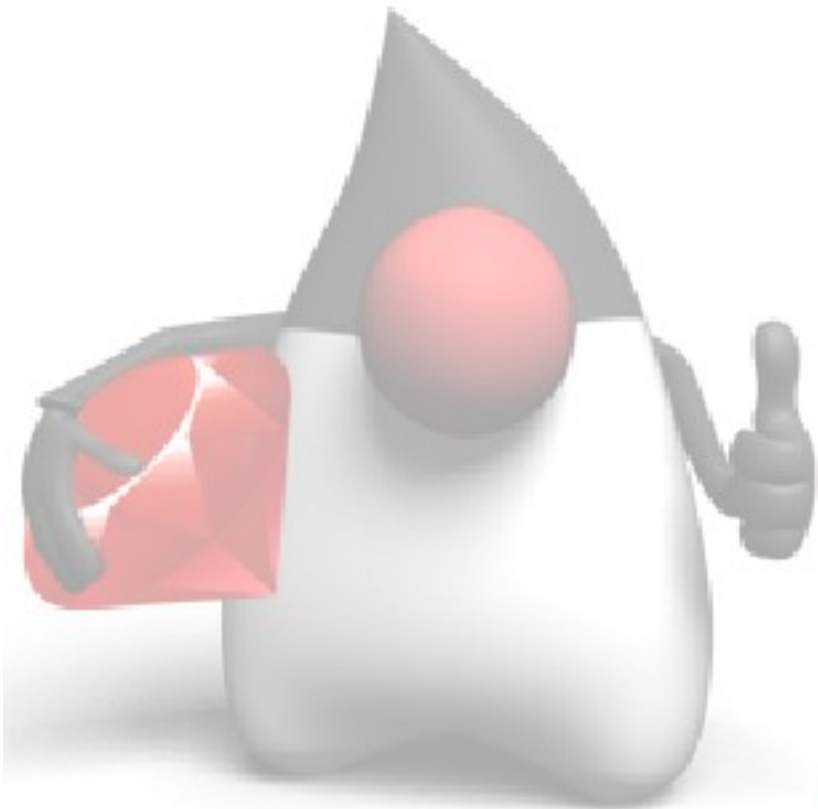


GC

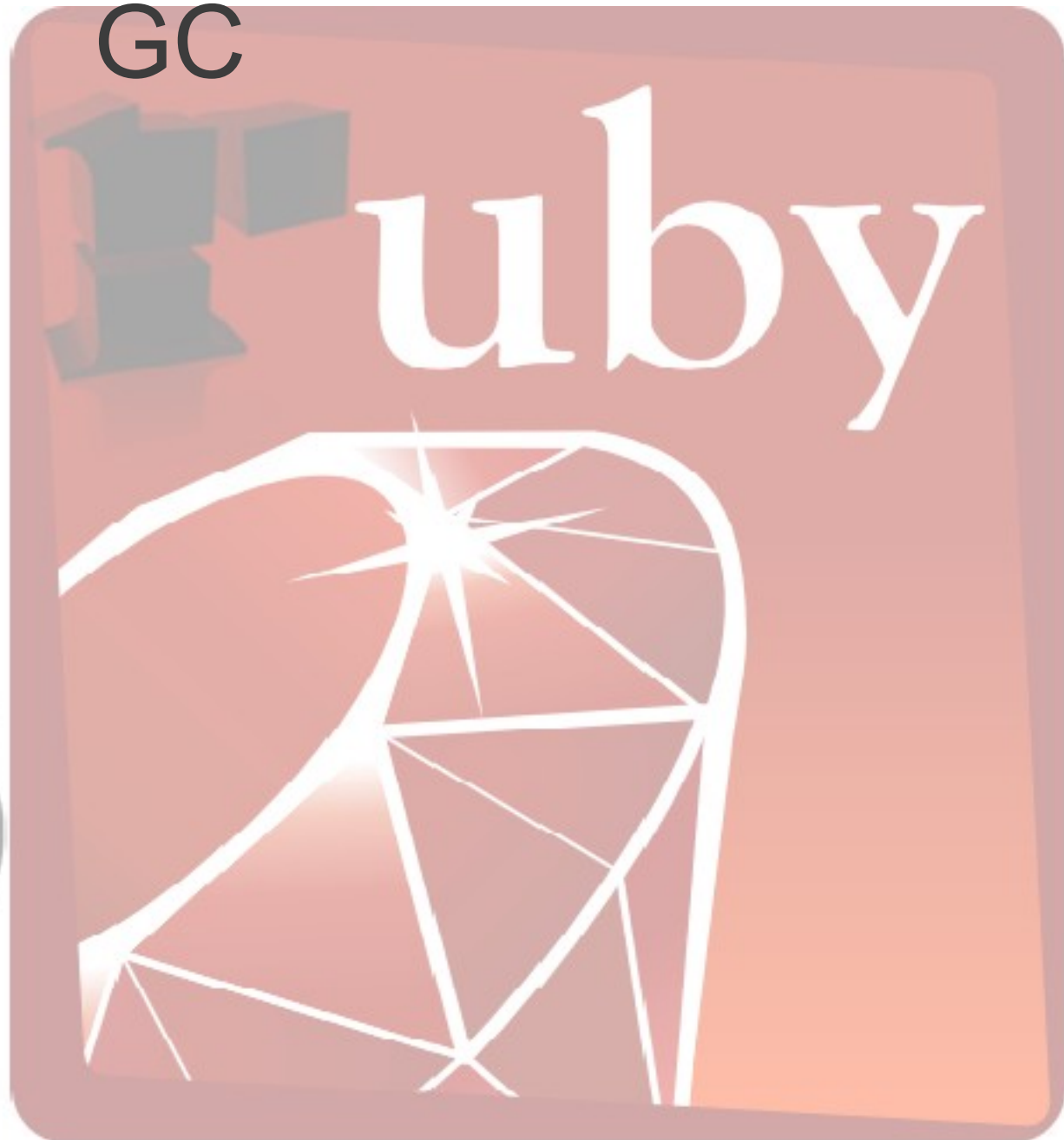
- Conservative Stop the World Mark and sweep
- Generational



- Ree
- Kiji

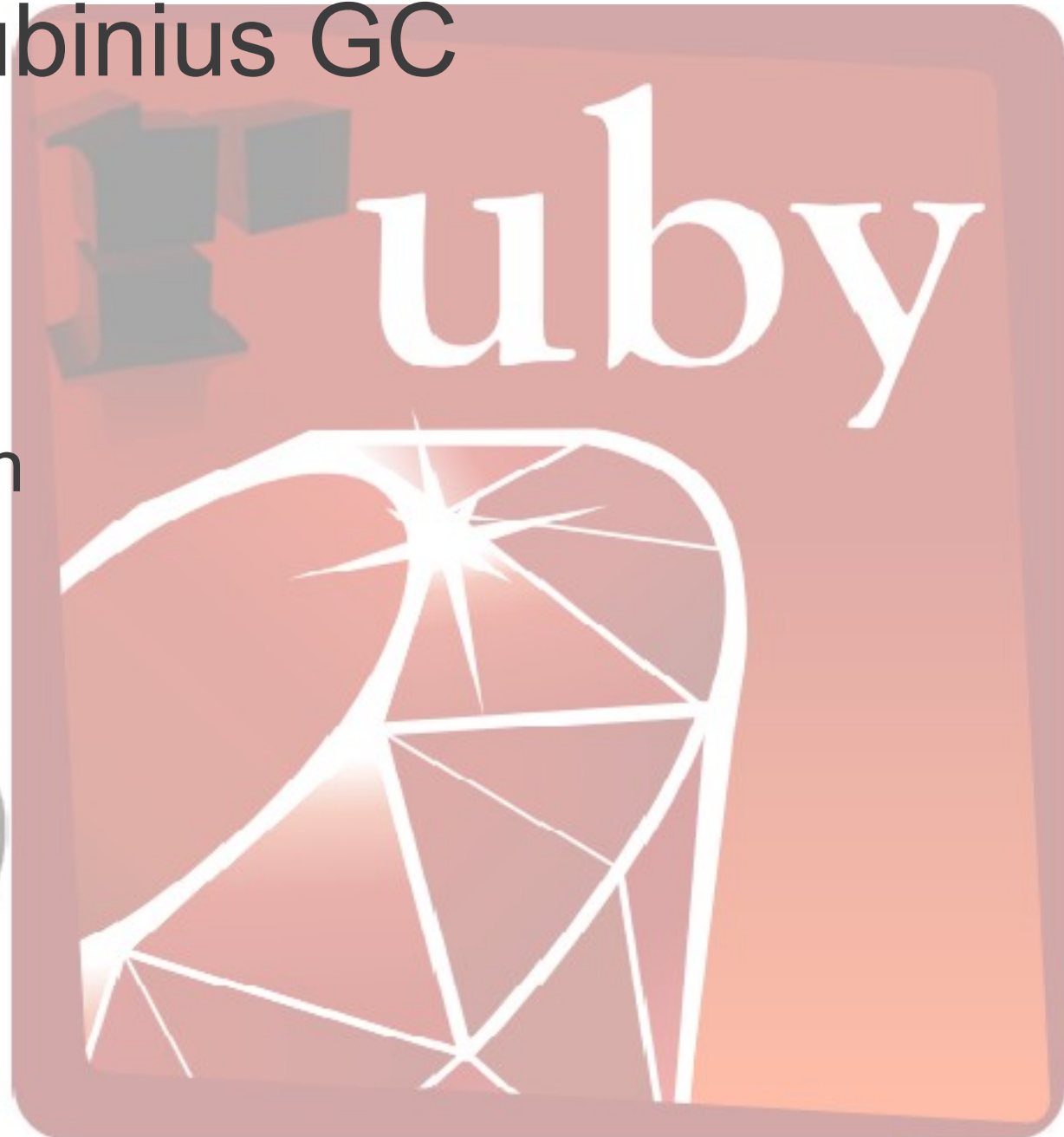
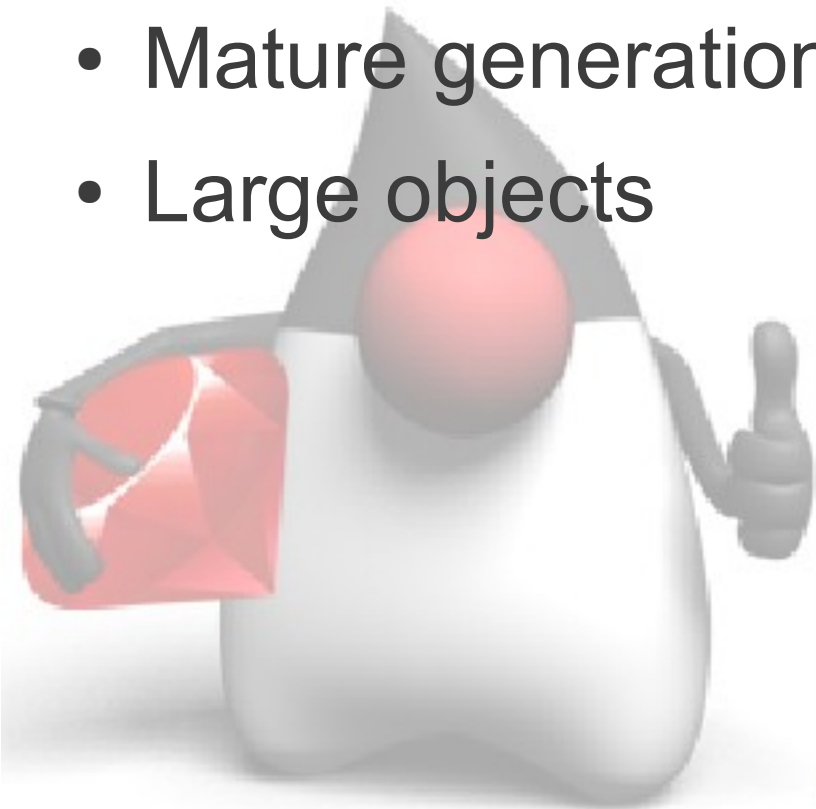


GC

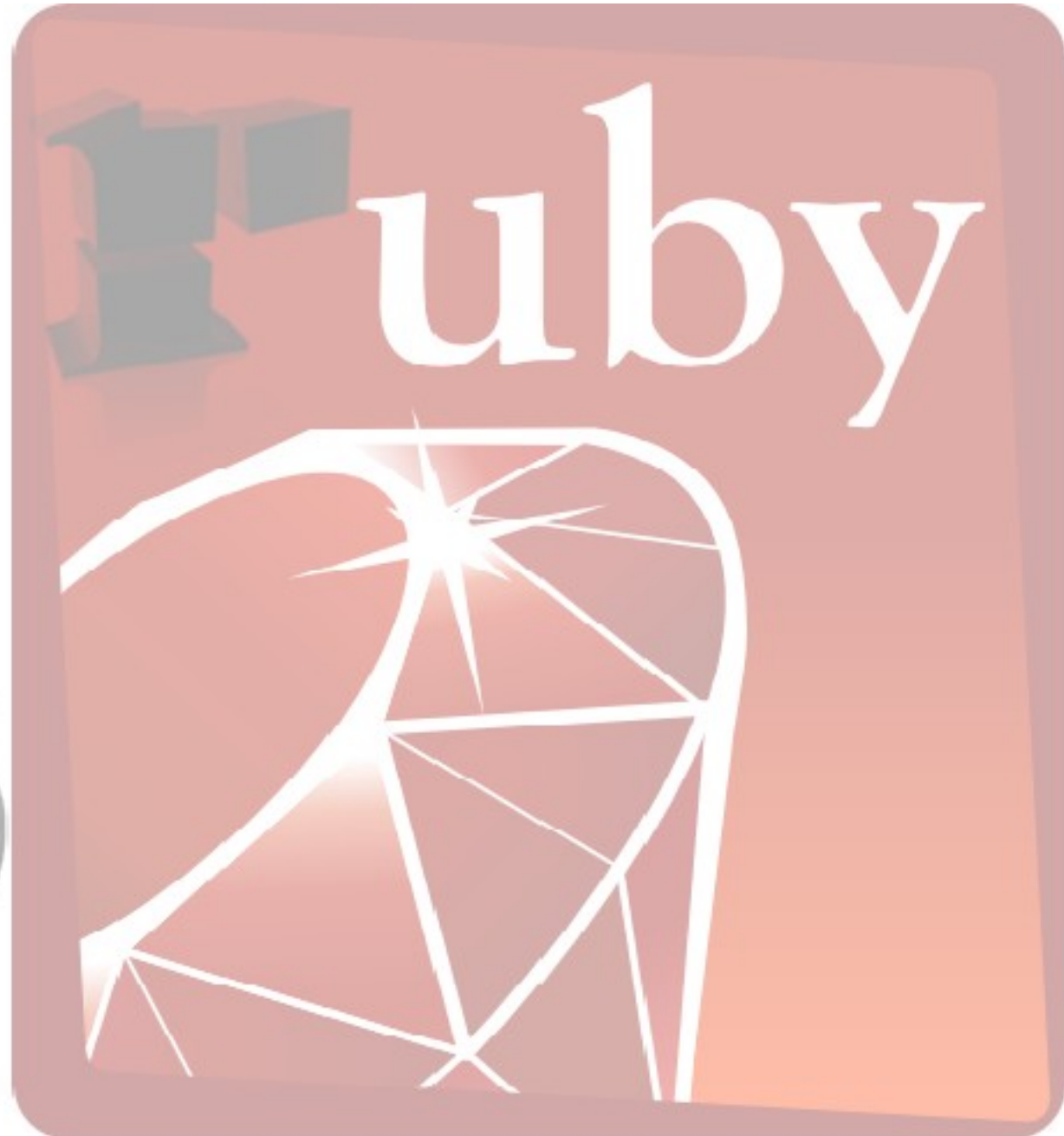


Rubinius GC

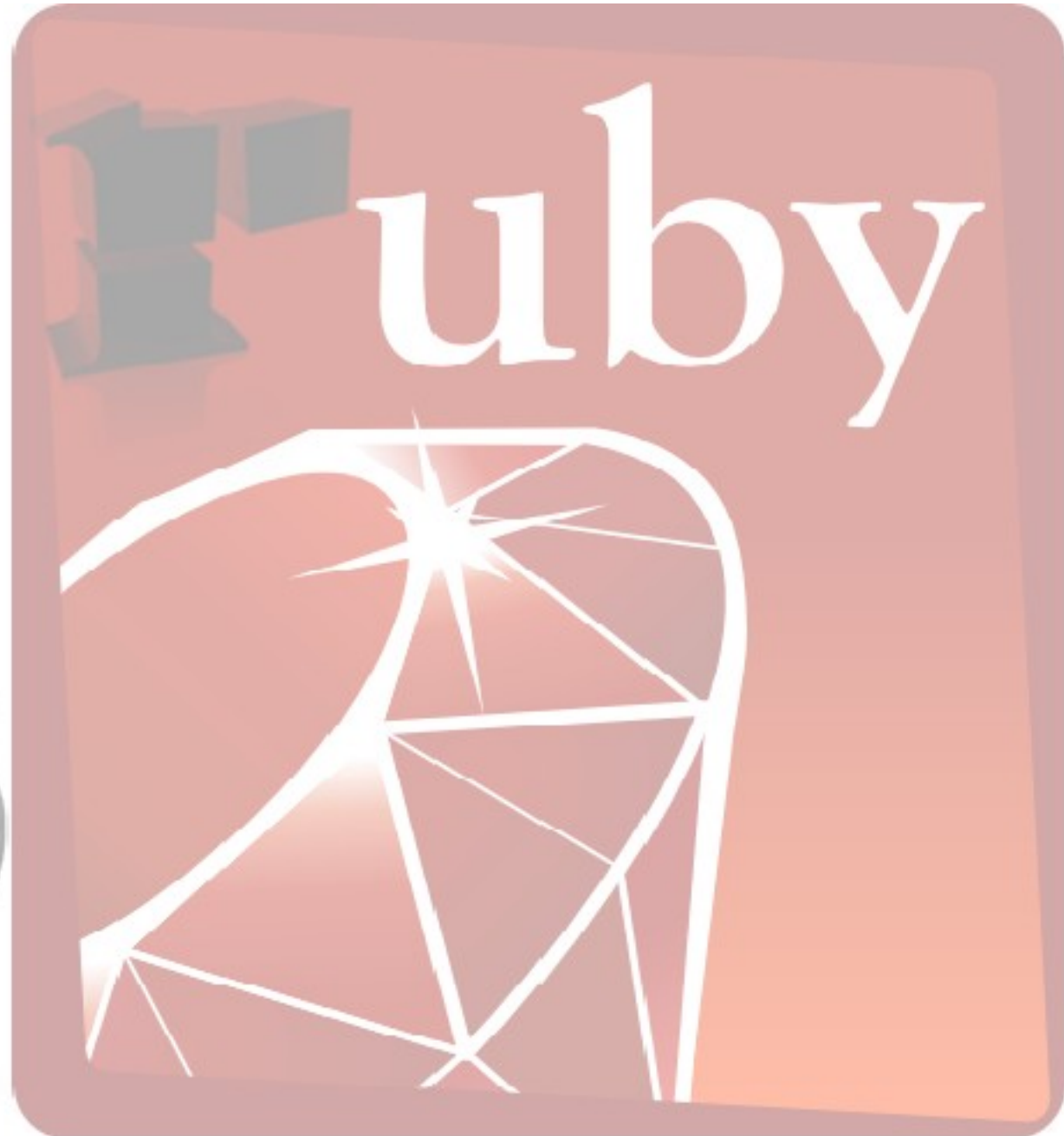
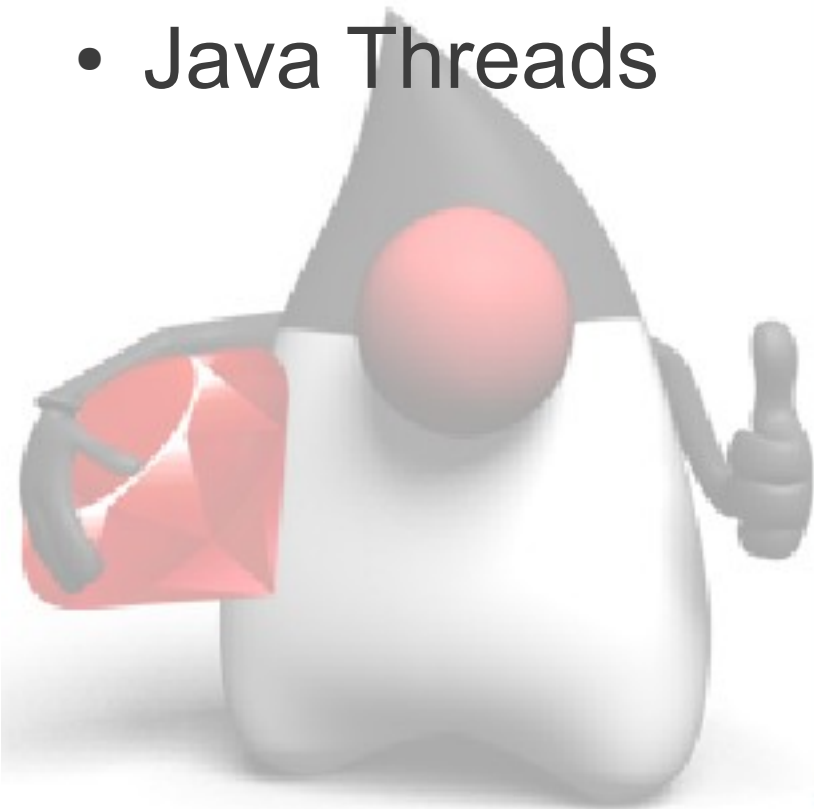
- Nursery
- Young generation
- Mature generation
- Large objects



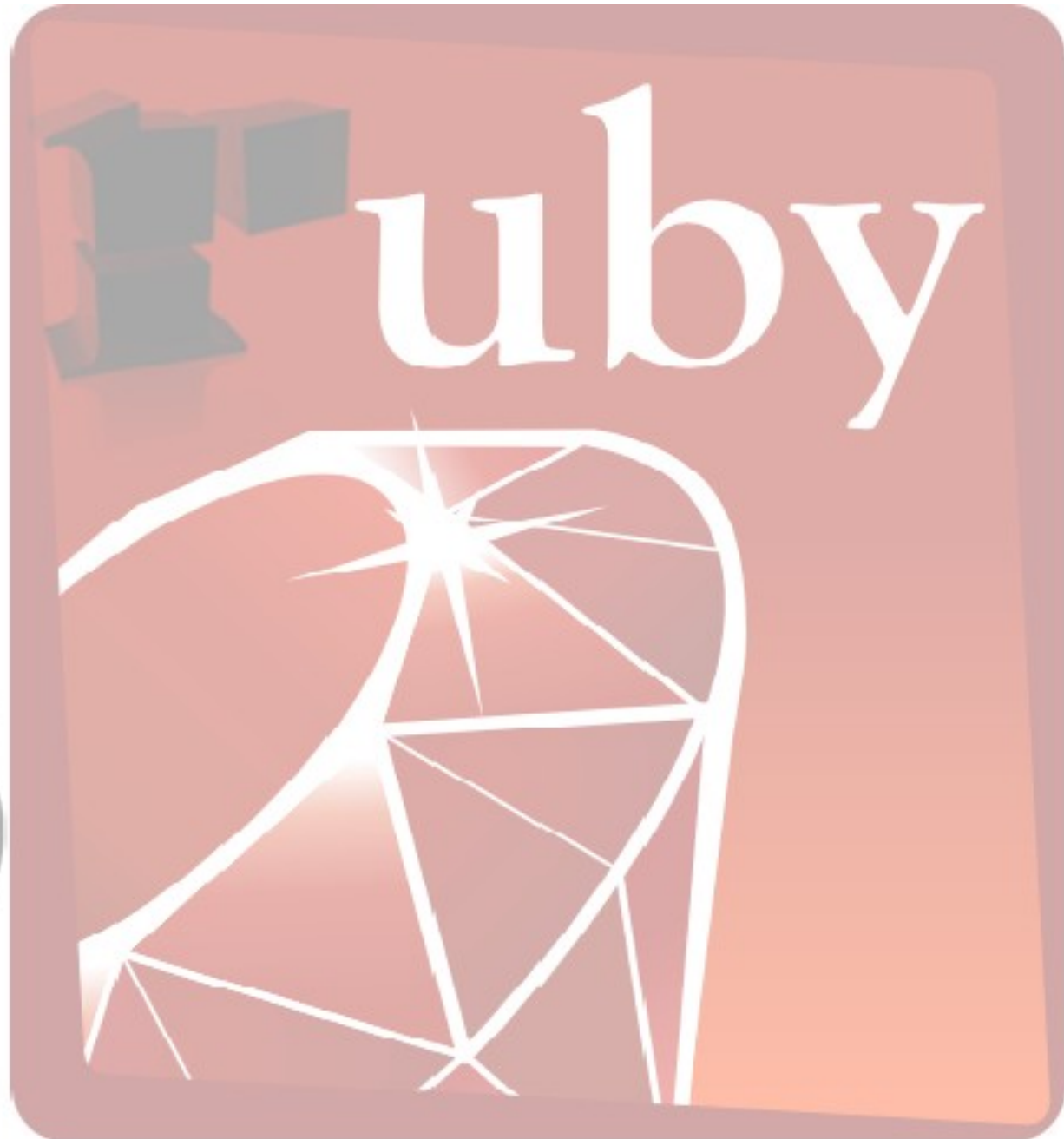
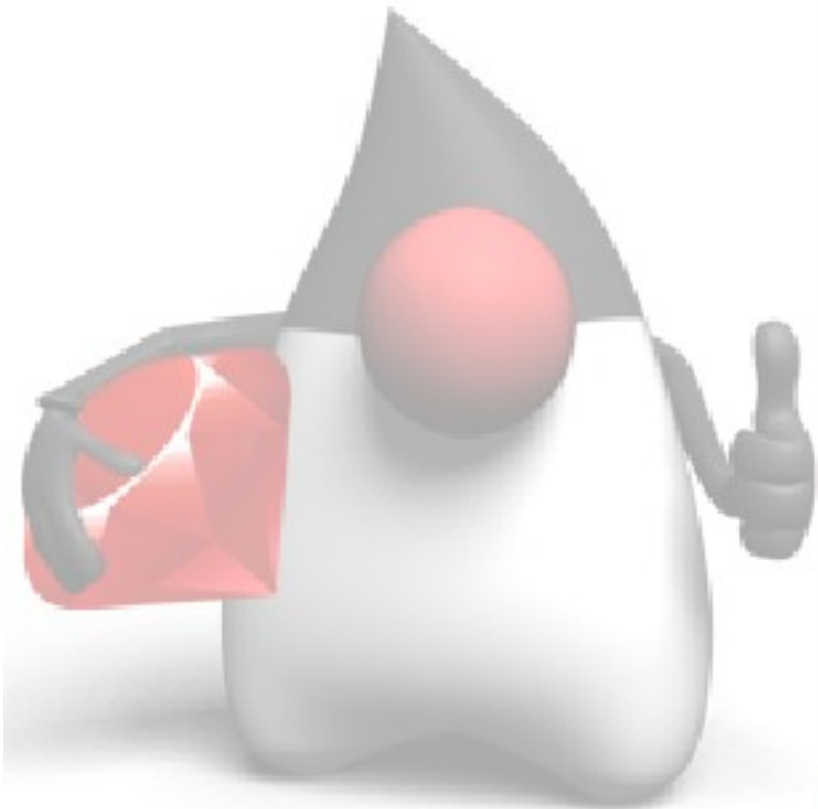
- Interpreter
- Optimizations
- GC
- **Threading**
- Extending ruby



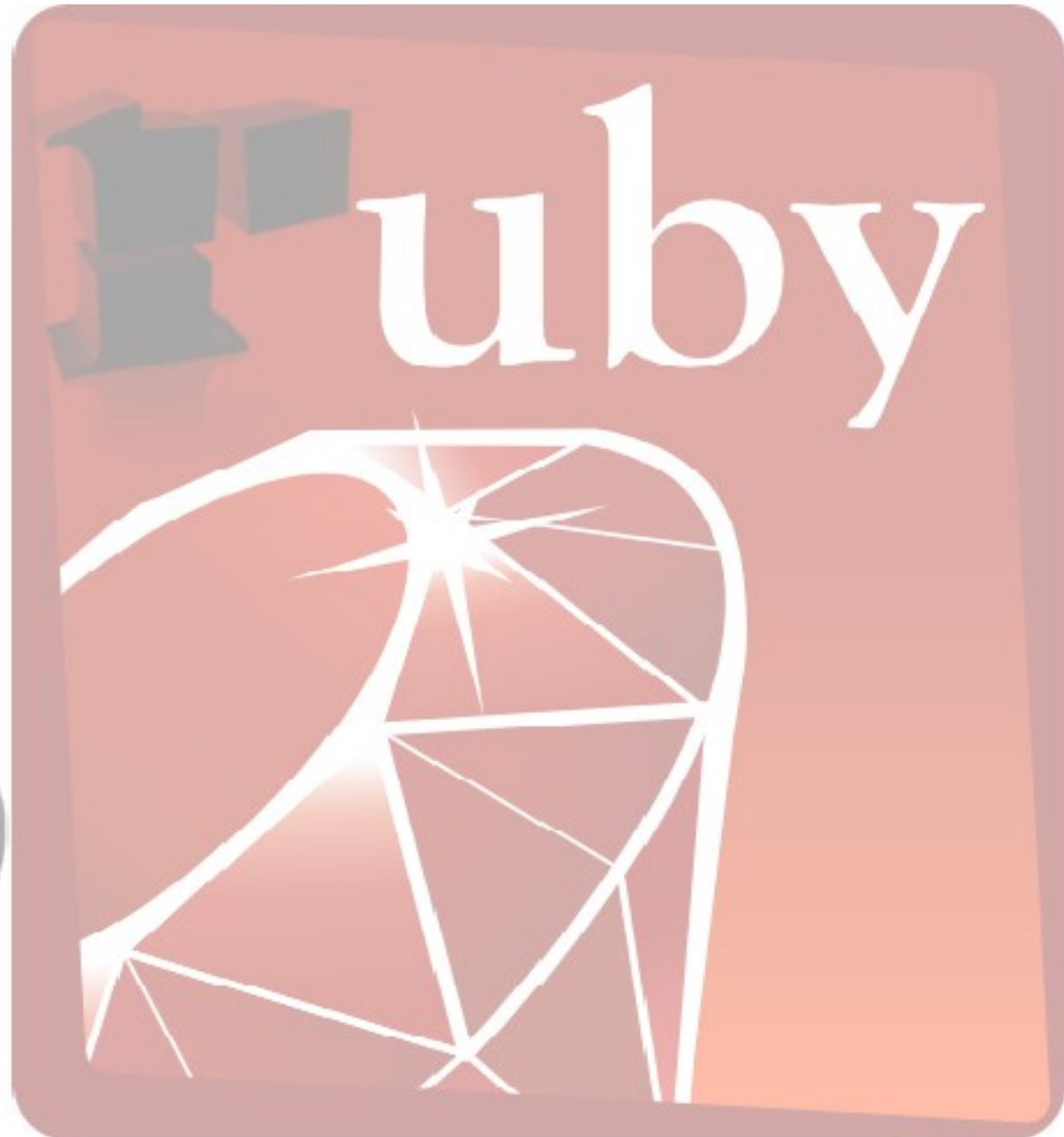
- Green threads
- System threads
- Java Threads



- JRuby
- MRI, Rubinius

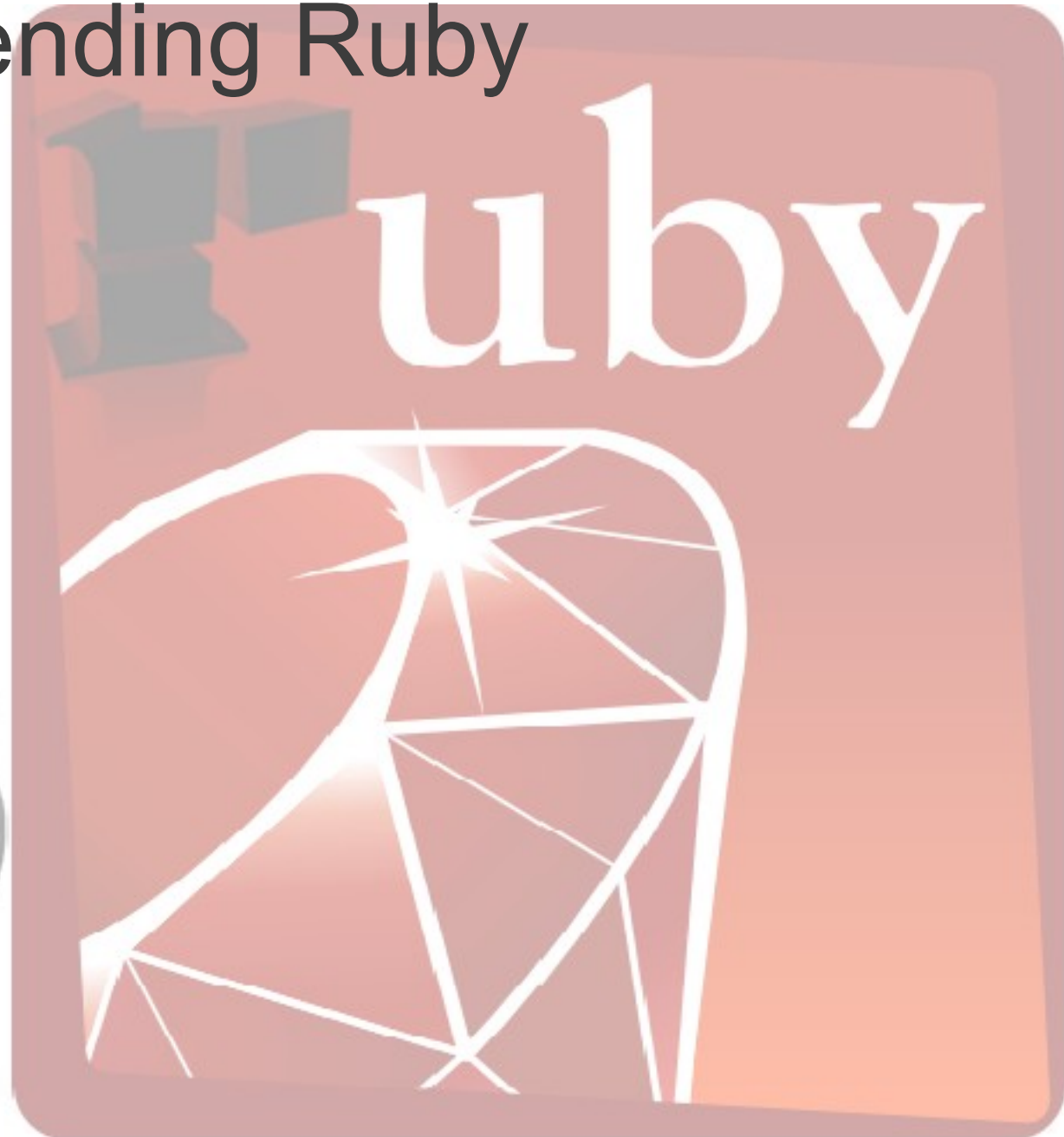
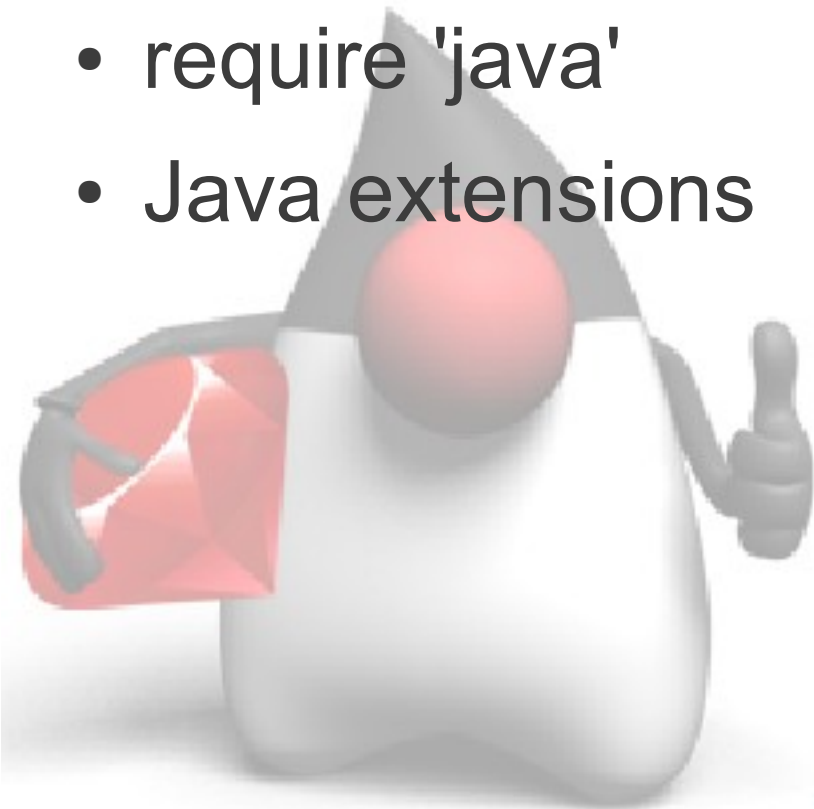


- Interpreter
- Optimizations
- GC
- Threading
- **Extending ruby**



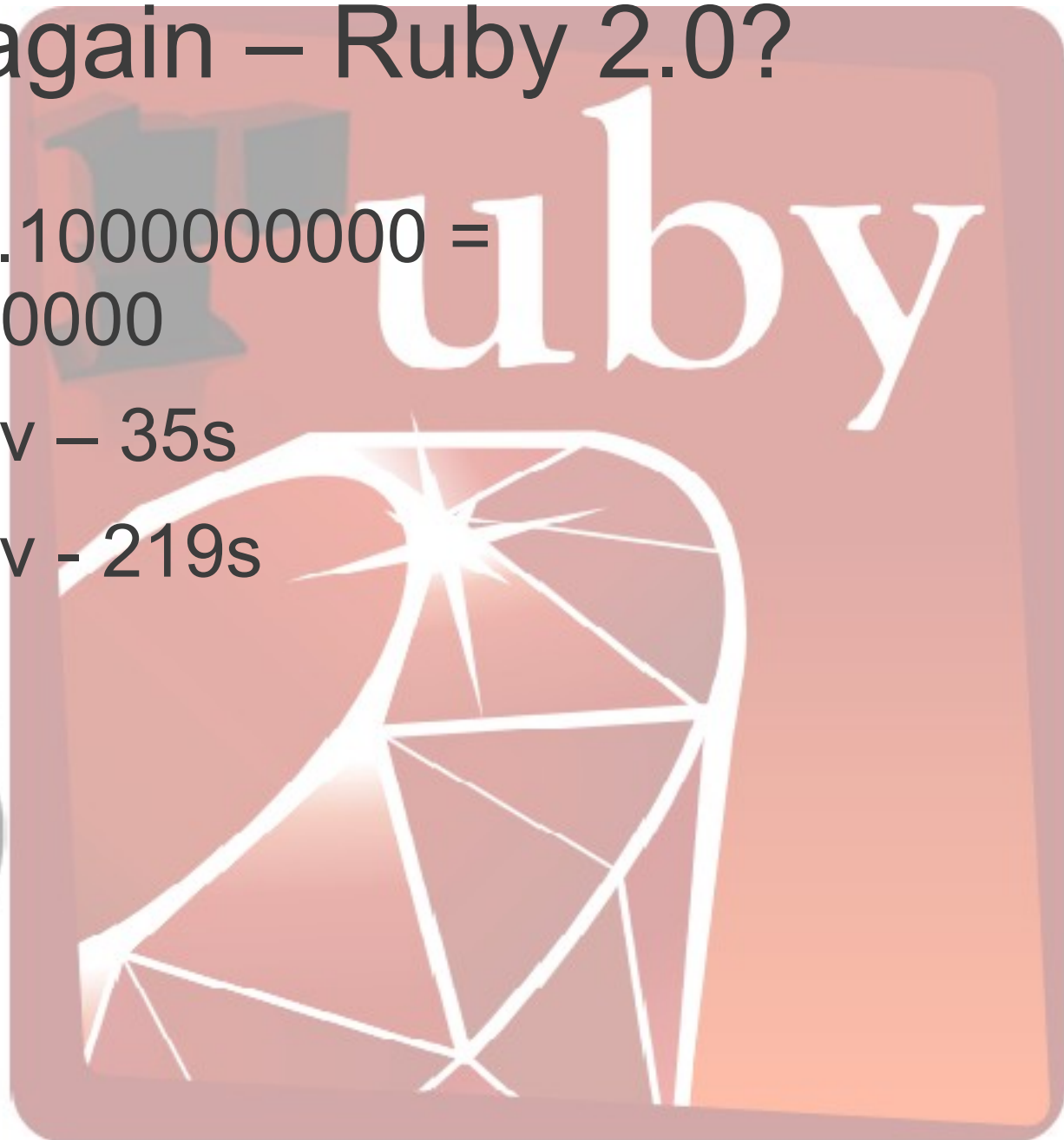
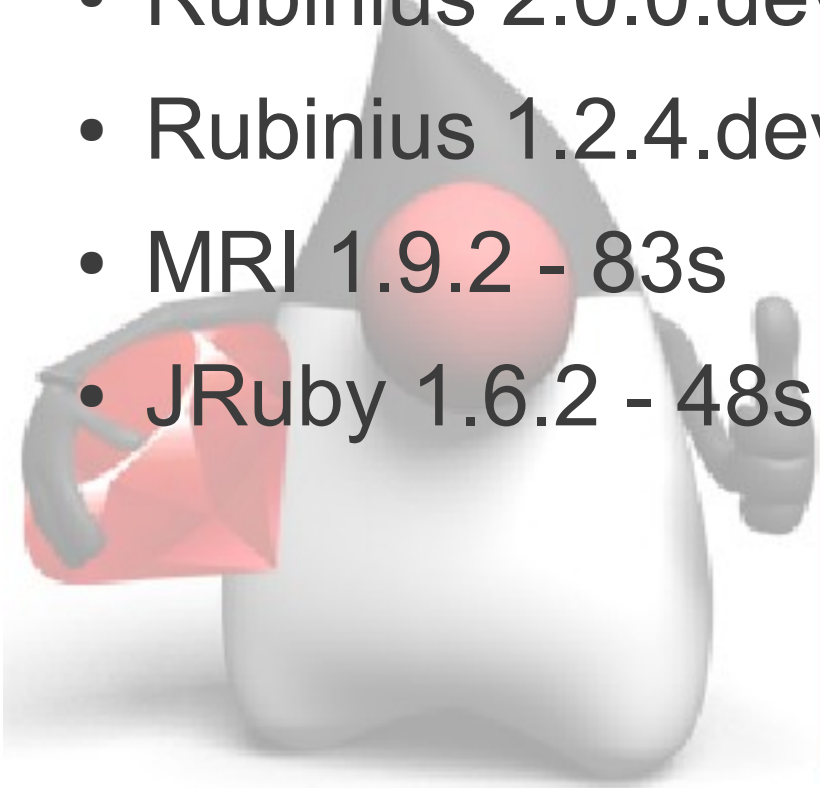
Extending Ruby

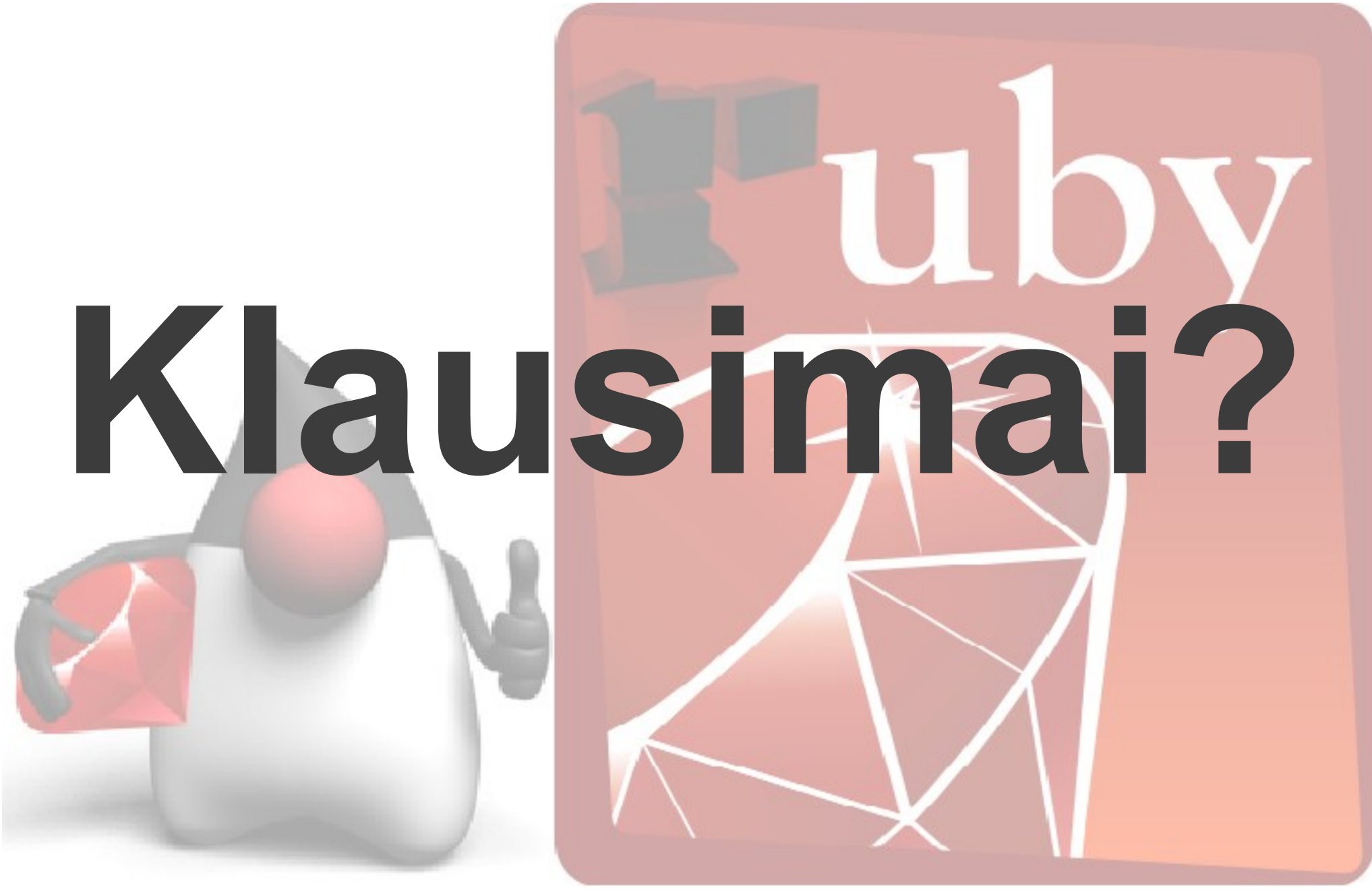
- C extensions
- FFI
- require 'java'
- Java extensions



Rubinius again – Ruby 2.0?

- 3 threads: $\text{sum } 1..10000000000 = 50000000005000000000$
- Rubinius 2.0.0.dev – 35s
- Rubinius 1.2.4.dev - 219s
- MRI 1.9.2 - 83s
- JRuby 1.6.2 - 48s





Klausimai?