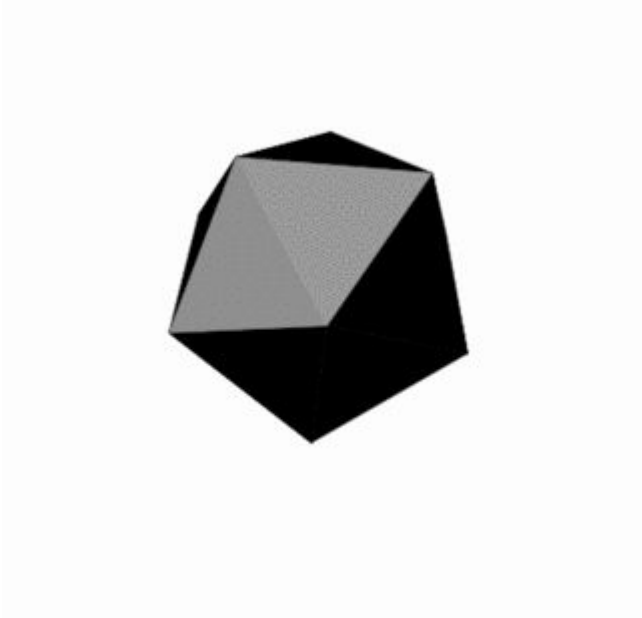
Getting to know Crystal: A language for humans and computers

Beta Ziliani - Manas.Tech | Crystal Core Team

Roadmap

CRYSTAL



Crystal is for humans

CRYSTAL

Humans like to write and read *clean* code

```
require "http/server"

class SimpleEcho
  PORT = 8080

  def self.start(port = PORT)
    server = HTTP::Server.new do |context|
      context.response.content_type = "text/plain"
      context.response.print "Hello world, got #{context.request.path}!"
    end

    Log.info { "Listening on http://localhost:#{port}/" }
    server.listen port
  end
end

SimpleEcho.start
```

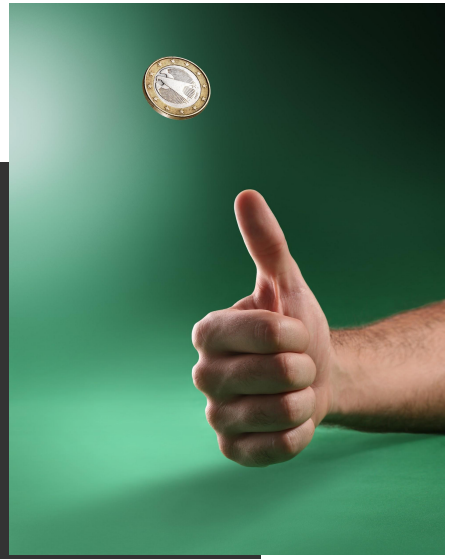
CRYSTAL

Humans like to avoid dependencies

```
require "http/server"

class SimpleEcho
  PORT = 8080

  def self.start(port = PORT)
    server = HTTP::Server.new do |context|
      context.response.content_type = "text/plain"
      context.response.print "Hello world, got #{context.request.path}!"
    end

    Log.info { "Listening on http://localhost:#{port}/" }
    server.listen port
  end
end

SimpleEcho.start
```

CRYSTAL

Humans like their programs to *not fail*



```
class Duck
  def quack
    puts "🦆 quacks!"
  end
end

if rand(2) >= 1
  duck = Duck.new
end

duck.quack
```

```
$ crystal example.cr
error in line 11
Error: undefined method 'quack' for Nil (compile-time type is (Duck | Nil))
```

CRYSTAL

Humans like to *avoid bureaucracy*

```
struct Nil
  def quack
    puts "🤫"
  end
end
```

```
$ crystal example.cr
🦆 quacks!
$ crystal example.cr
🤫
```

Duck typing + monkey patching like in Ruby

CRYSTAL

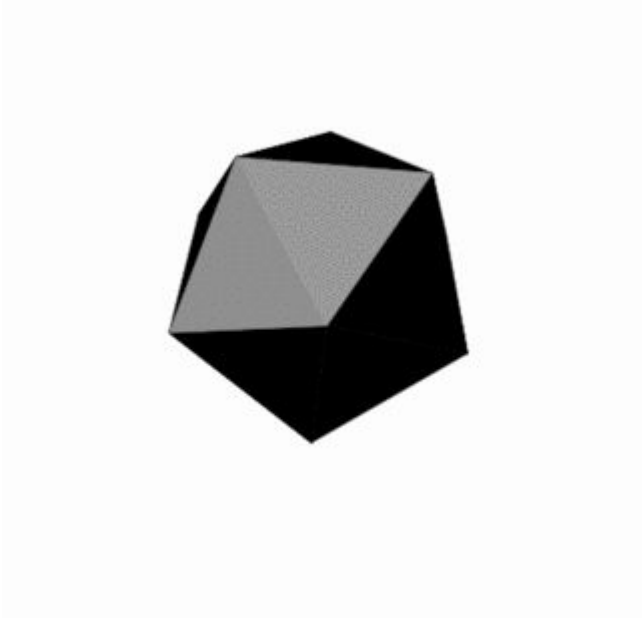
Summing up

- Pretty like Ruby
  - Similar syntaxis, but compatibility is not a goal
- Safer: checks types

- Has type inference
  - No need to write boilerplate types
- Duck typing and monkey patching

} bureaucracy free!

CRYSTAL

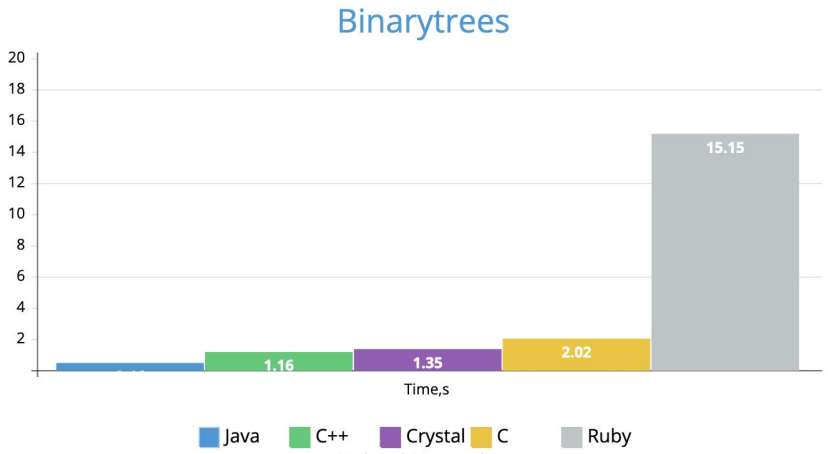


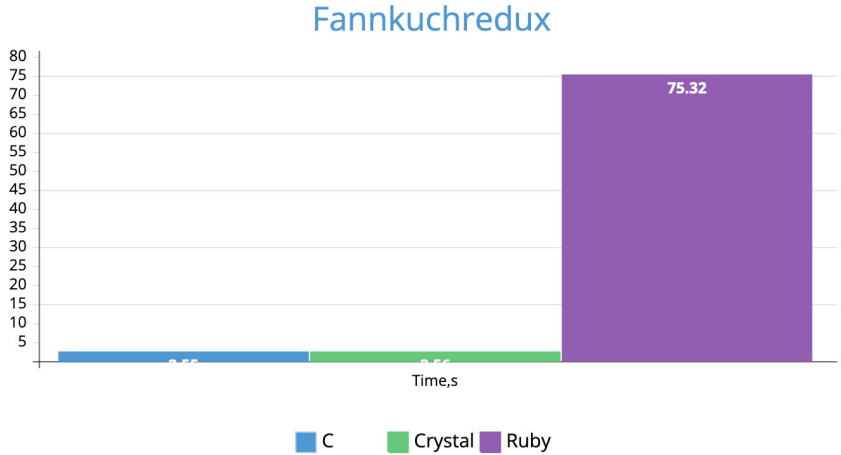
Crystal is for computers

CRYSTAL

Computers like native code

Performant execution



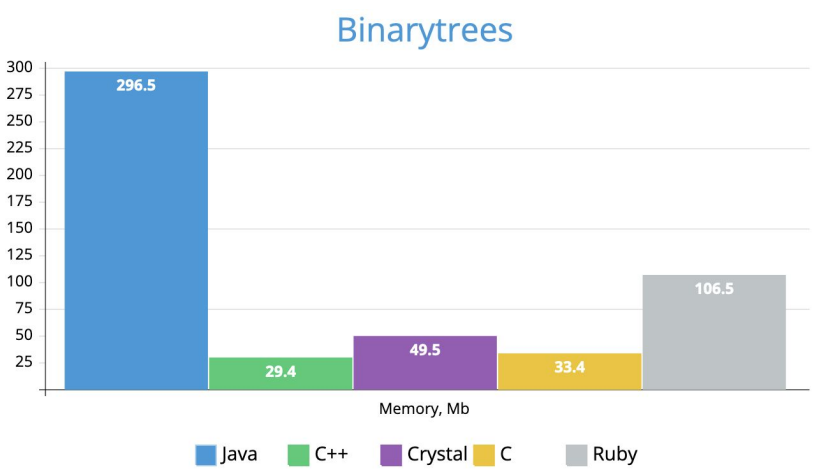


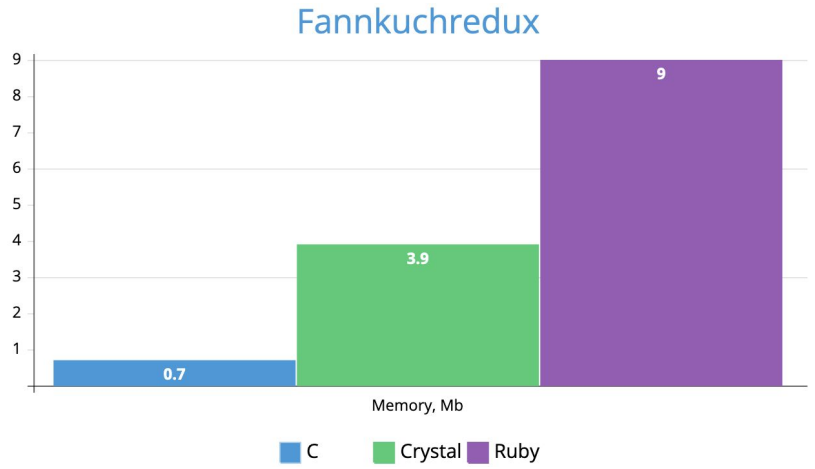
Source https://github.com/kostya/crystal-benchmarks-game
Note: Distrust benchmarks!

CRYSTAL

Computers like native code

Low memory footprint





Source https://github.com/kostya/crystal-benchmarks-game
Note: Distrust benchmarks!

CRYSTAL

Computers like operating systems


macOS


Ubuntu


Kubuntu


Debian


RedHat


CentOS


OpenSUSE


Arch Linux


elementary OS


Fedora


KDE Neon


Manjaro


Linux Mint


Gentoo Linux


Alpine Linux


FreeBSD


OpenBSD


Windows Subsystem for Linux


Windows (Preview)


from asdf

CRYSTAL

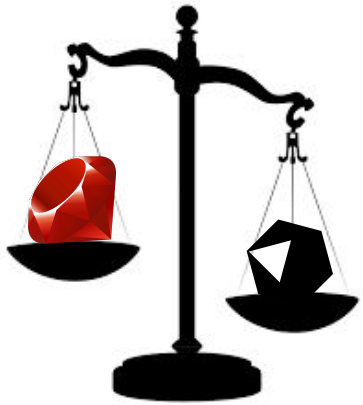
Summing up

- Competes with performant compiled languages
  - Speed
  - Memory consumption
- Supported in several OSs
  - Windows native not quite there *yet*

CRYSTAL

Why change Ruby for Crystal?

Crystal is better than Ruby



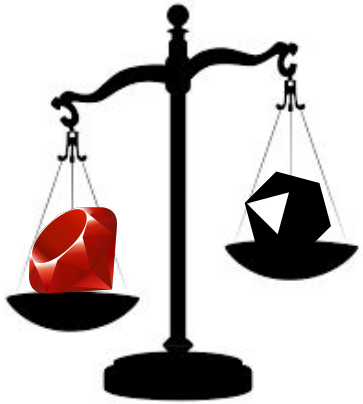
- Faster execution
- Smaller memory consumption

} 🌱 a *greener* language

- Easier to deploy
  - No interpreter needed, just the binaries!
- Safer

CRYSTAL

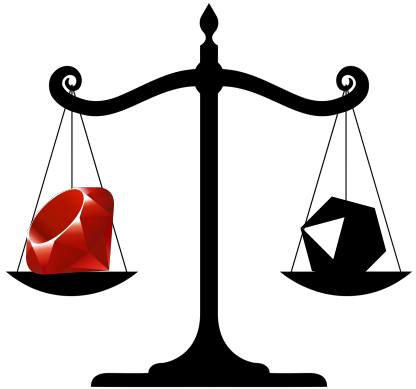
Ruby is better than Crystal



- Faster development
  - Need to wait for the compiler!
- Larger ecosystem
- Larger community

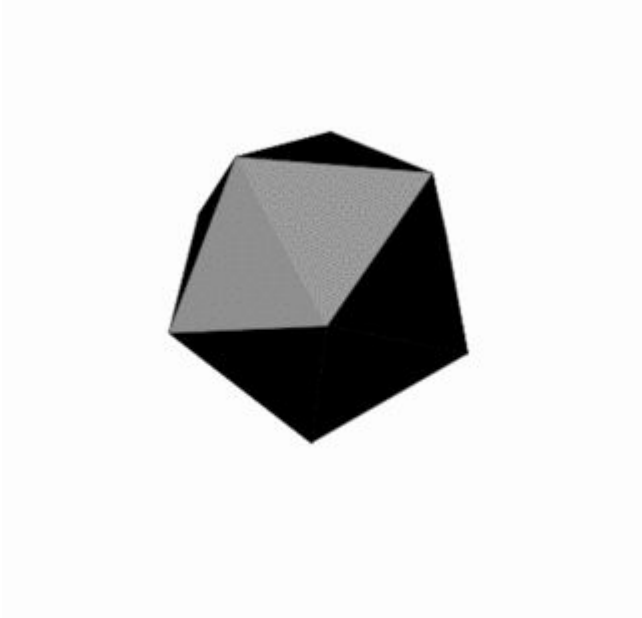
CRYSTAL

So...



- They are *incomparable*
- Simply use what works best for you...
- ... or use the two!
  - https://twinslash.com/services/ruby-on-rails-crystal
  - https://dev.to/seesethcode/dual-booting-rails-7-kemal-a-crystal-framework-1j2p

CRYSTAL

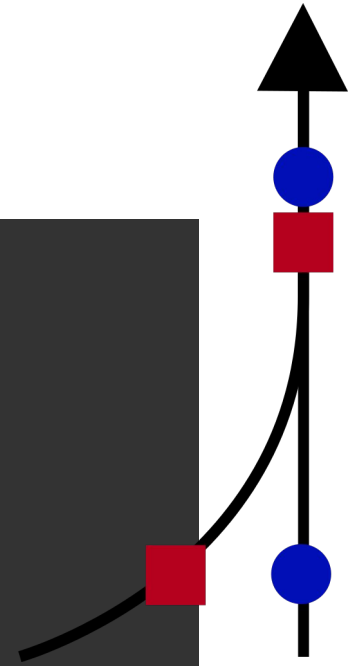


Perks to get you excited

CRYSTAL

Go style concurrency

```
require "http/client"

channel = Channel(Int32).new

SITES = ["www.example.com", "info.cern.ch"]

SITES.each do |site|
  spawn do
    response = HTTP::Client.new(site).get "/"
    channel.send response.body.each_line.size
  end
end

total_lines = 0

SITES.size.times do
  total_lines += channel.receive
end

puts "In total there are #{total_lines} lines"
```



CRYSTAL

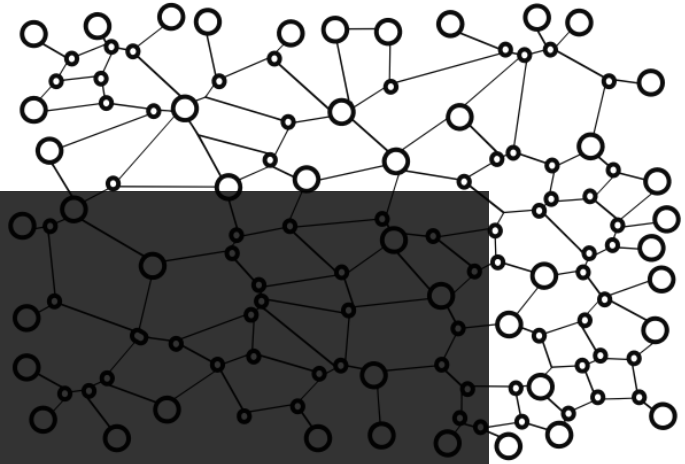
Macros


```
class Roman
  def roman_to_int(str)
    str = str.upcase
    # etc…
  end

  macro method_missing(call)
    roman_to_int({{call.name.id.stringify}})
  end
end

r = Roman.new
puts r.iv       # => 4
puts r.xxiii    # => 23
puts r.mm       # => 2000
```

Compile-time generation of code

CRYSTAL

Shards: Dependency manager



```
# file shard.yml
name: my-project

dependencies:
  mysql:
     github: crystal-lang/crystal-mysql
```

```
$ shards install
Resolving dependencies
Fetching https://github.com/crystal-lang/crystal-mysql.git
Fetching https://github.com/crystal-lang/crystal-db.git
Installing db (0.11.0)
Installing mysql (0.14.0)
Writing shard.lock
```

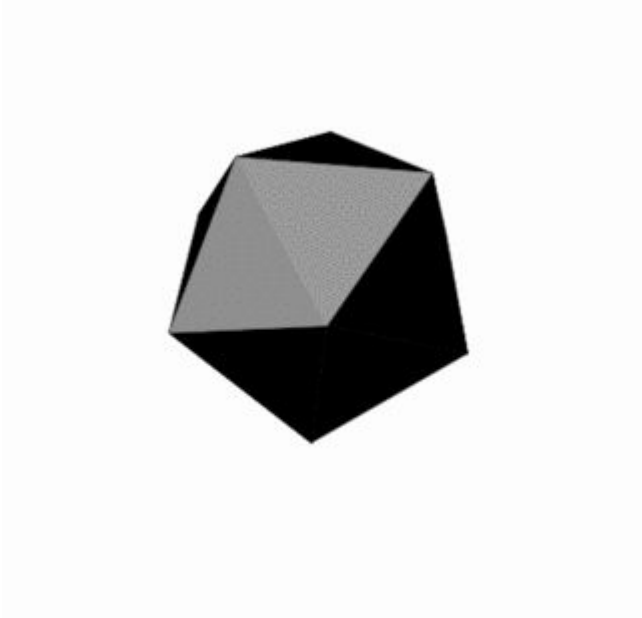
CRYSTAL

C-bindings without tears

```
@[Link("gmp")]
lib LibGMP
  struct MPZ
    _mp_alloc : LibC::Int
    _mp_size : LibC::Int
    _mp_d : LibC::ULong*
  end

  fun init_set_str =  __gmpz_init_set_str(rop : MPZ*, str : UInt8*, base : LibC::Int) : LibC::Int
  fun cmp = __gmpz_cmp(op1 : MPZ*, op2 : MPZ*) : LibC::Int
end

LibGMP.init_set_str(out left, "1230000000", 10)
LibGMP.init_set_str(out right, "456000000", 10)

puts LibGMP.cmp(pointerof(left), pointerof(right)) > 0
```

```
typedef struct {
  int _mp_alloc;
  int _mp_size;
  mp_limb_t *_mp_d;
} __mpz_struct;

int mpz_init_set_str(mpz_ptr, const char *, int);
int mpz_cmp(mpz_srcptr, mpz_srcptr);
```

CRYSTAL



There’s more to Crystal

Find out at
www.crystal-lang.org

CRYSTAL

THANK YOU

CRYSTAL