Hey ho, Io!

Mariusz Lusiak

applicake.com

May 11, 2011

2011-05-11

Io

Hey ho, Io!

Mariusz Lusiak

applicake.com

May 11, 2011

Io

2011-05-11

The origin and the usage of the Io language

2011-05-11

Io
└─The origin and the usage

The origin and the usage of the Io language

2002

2002

- rather new

Steve Dekorte

- written to help him understand how languages work
- looks like a good way to get more insight into programming languages

Io

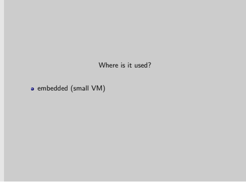
Io

Where is it used?

Where is it used?

Where is it used?

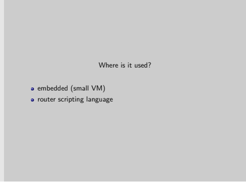
- embedded (small VM)


Where is it used?

- embedded (small VM)

Where is it used?

- embedded (small VM)
- router scripting language


Where is it used?

- embedded (small VM)
- router scripting language

Where is it used?

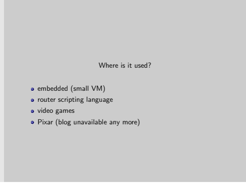
- embedded (small VM)
- router scripting language
- video games

Where is it used?

- embedded (small VM)
- router scripting language
- video games

Where is it used?

- embedded (small VM)
- router scripting language
- video games
- Pixar (blog unavailable any more)



Where is it used?

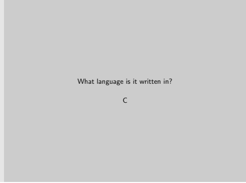
- embedded (small VM)
- router scripting language
- video games
- Pixar (blog unavailable any more)

What language is it written in?

What language is it written in?

What language is it written in?

C


What language is it written in?

C

The features of the Io language

The features of the Io language

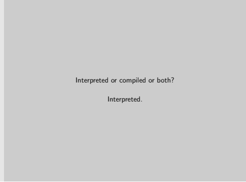
- many approaches to new language
- hello world tutorial
- sit down and hack away
- interactive console
- as professionals we should make an scientific approach
- ask general questions, good questions, that lead to meaningful answers
- that's what makes us different from high school kids
- what kind of questions can we ask?

Interpreted or compiled or both?

- determines interaction programmer - language

Interpreted or compiled or both?

Interpreted.


Interpreted or compiled or both?

Interpreted.

- determines interaction programmer - language

Programming paradigm?

- structural
- object-oriented
- functional

Programming paradigm?

Object oriented.

Programming paradigm?

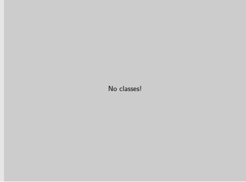
Object oriented.

- structural
- object-oriented
- functional

Prototype-based

- in Ruby we work with objects all the time
- but Io has no classes!

No classes!



- you might be wondering
- how objects are created?
- in Ruby object is an instance of a class
- how is inheritance implemented
- we'll get deeper in a moment

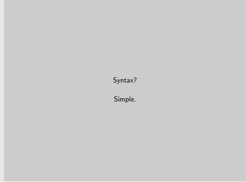
Syntax?

Syntax?

- syntax is next important thing
- how long will it take to learn a language?
- Io has simple syntax

Syntax?

Simple.


Syntax?

Simple.

- syntax is next important thing
- how long will it take to learn a language?
- Io has simple syntax

Expressive?

- little syntactic sugar
- in Ruby you can express complex thoughts in little writing

Expressive?

Not really.

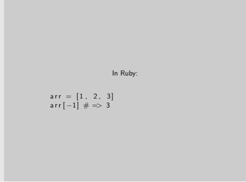
Expressive?

Not really.

- little syntactic sugar
- in Ruby you can express complex thoughts in little writing

In Ruby:

```
arr = [1, 2, 3]
arr[-1] # => 3
```


In Ruby:

```
arr = [1, 2, 3]
arr[-1] # => 3
```

- this is a nice syntax if you know it

Simple syntax.

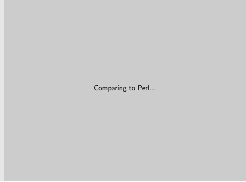
Simple syntax.

- makes it easy to write things
- makes it harder to understand complex thoughts

Clean syntax.

Clean syntax.

Comparing to Perl...


Comparing to Perl...

- write-only language

Clean syntax.

- little strange chars

Type system

Type system

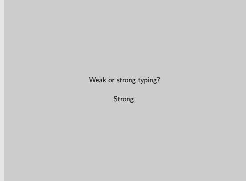
Weak or strong typing?

Weak or strong typing?

- var = 1, var = aaa
- 3 plus string in Ruby, in Javascript
- conditional statements (compare to Java)

Weak or strong typing?

Strong.


Weak or strong typing?

Strong.

- var = 1, var = aaa
- 3 plus string in Ruby, in Javascript
- conditional statements (compare to Java)

Static or dynamic typing?

Static or dynamic typing?

Static or dynamic typing?

Dynamic.

Static or dynamic typing?

Dynamic.

Support for concurrency?

Support for concurrency?

Support for concurrency?

Strong.

Support for concurrency?

Strong.

Not language features but important.

Not language features but important.

Community?

Community?

Community?

Small.

Community?

Small.

Testing frameworks?

Testing frameworks?

Testing frameworks?

UnitTest.

Testing frameworks?

UnitTest.

The object model and the semantics of the Io language

The object model and the semantics of the Io language

Prototype-based

- Javascript uses this, too

No class

No class

- how do we create objects?

Cloning objects

Cloning objects

```
Vehicle := Object clone
```

Vehicle := Object clone

- Object is provided by interpreter

```
Vehicle description := "Something"
```

Vehicle description := "Something"

- creating slots

```
Vehicle description # => "Something"
```

Vehicle description # => "Something"

- we sent a message with a slot name

```
Vehicle description = "Ble"
Vehicle otherSlot = "Ble" # => Error
```

Vehicle description = "Ble"
Vehicle otherSlot = "Ble" # => Error

```
Vehicle slotNames
# => list("type", "description")
```

Vehicle slotNames
=> list("type", "description")

```
Vehicle type # => Vehicle
```

Vehicle type # => Vehicle

Object type # => Object

Object type # => Object

```
Car := Vehicle clone
```

```
Car := Vehicle clone
Car slotNames # => list("type")
```

Car slotNames # => list("type")

```
Car description # => "Ble"
```

Car description # => "Ble"

```
Car type # => Car
```

Car type # => Car

```
ferrari := Car clone
```

ferrari := Car clone

```
ferrari slotNames # => list()
```

```
ferrari slotNames # => list()
```

- convention - small letter, no type
- better code organization

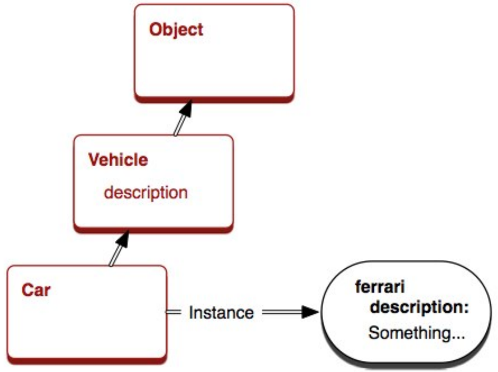
```
ferrari type # => Car
```

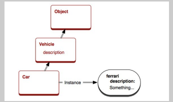
ferrari type # => Car

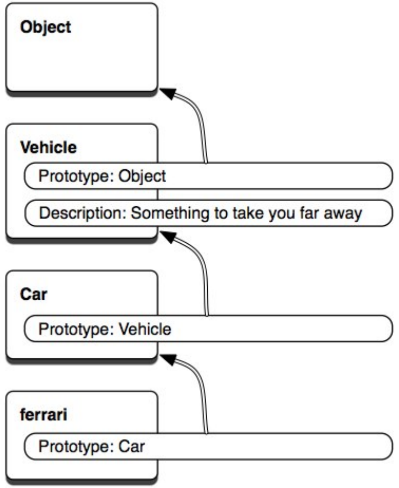
Objects are collections of slots.

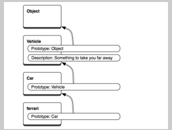
Objects are collections of slots.

- if slot is not found it is sent to parent


Object
Vehicle
description
Car
Instance
ferrari
description:
Something...




Object
Vehicle
Prototype: Object
Description: Something to take you far away
Car
Prototype: Vehicle
ferrari
Prototype: Car



Methods

Methods

```
method("Something" println)
```

method("Something" println)

```
Car drive := method("Vroom" println)

ferrari drive # => Vroom
```

Car drive := method("Vroom" println)
ferrari drive # => Vroom

```
ferrari getSlot("drive")
# => method("Vroom" println)
```

ferrari getSlot("drive")
=> method("Vroom" println)

```
ferrari proto # => Car
Car proto # => Vehicle
```

ferrari proto # => Car
Car proto # => Vehicle

true, false, nil

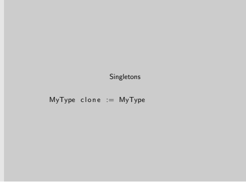
true, false, nil

Singletons

- how to create a singleton?

Singletons

```
MyType clone := MyType
```



Singletons

MyType clone := MyType

- how to create a singleton?

```
Object clone := "dupa"
```

Object clone := "dupa"

Messages

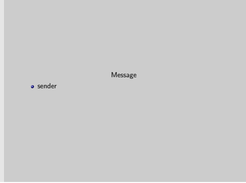
- all interactions are done with messages
- everything is a message (and message is an object)

Message

Message

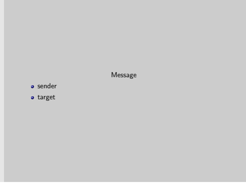
Message

- sender



Message

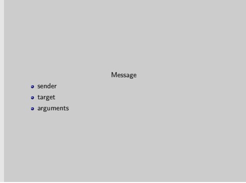
- sender
- target


Message

- sender
- target

Message

- sender
- target
- arguments


Message
- sender
- target
- arguments

```
for(i, 1, 10, i println)
a := if(b == 0, c + 1, d)
```

for(i, 1, 10, i println)
a := if(b == 0, c + 1, d)

Reflection

Reflection

The concurrency in the Io language

The concurrency in the Io language

Coroutines, Actors, Futures

Coroutines, Actors, Futures

Coroutines

Coroutines

- methods that voluntarily pass execution to other process

Live coding time!

Live coding time!

Actors

Actors

Futures

Futures

```
futureResult :=
  URL with("http://google.com/") @fetch
writeln("something")
writeln(
  "fetched",
  futureResult size,
  " bytes")
```

```
futureResult :=
  URL with("http://google.com/") @fetch
writeln("something")
writeln(
  "fetched",
  futureResult size,
  " bytes")
```

The summary

The summary

Interesting stuff...

2011-05-11

Io
└─The summary

Interesting stuff...

But not extremely interesting...

But not extremely interesting...

Strengths

Strengths

Small size

2011-05-11

Io
└The summary

Small size

Simplicity

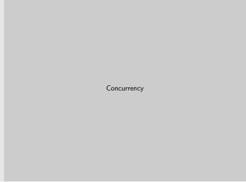
Simplicity

Flexiility

2011-05-11

Flexiility

Concurrency



- SIMD

Weaknesses

Weaknesses

Syntax

Syntax

Community

Community

Performance

Performance

No stable version

No stable version

- problems with addons
- URL