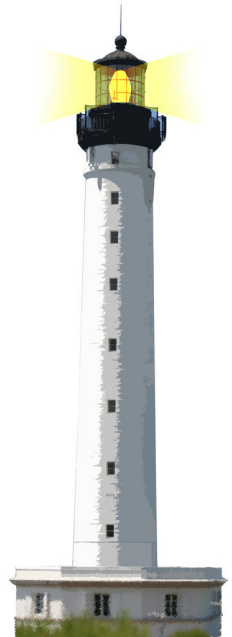
Messages: Composition and Precedence

Damien Cassou, Stéphane Ducasse and Luc Fabresse

W2S03

Pharo

http://www.pharo.org



Composition: from Left to Right!

What happens when we have two messages of the same kind?

- Execution from left to right

```
1000 factorial class name
> 'LargePositiveInteger'
```

is equivalent to

```
(((1000 factorial) class) name)
```

- Ease the composition of messages

Complete Message Precedence

- (Msg) > Unary > Binary > Keywords
- From left to right

Precedence Example

```
2 + 3 squared
> 2 + 9
> 11
```

- unary (squared) first
- then binary (+)

Precedence Example

```
2 raisedTo: 3 + 2
> 2 raisedTo: 5
> 32
```

- binary (+) first
- then keyword-based (raisedTo:)

Precedence Example

```
Color gray – Color white = Color black
> aGray – aWhite = aBlack
> aBlack = aBlack
> true
```

- unary messages
- then binary from left to right

Precedence Example

```
1 class maxVal + 1
> 1152921504606846976
```

- unary, unary and binary

```
1 class
> SmallInteger

1 class maxVal
> 1152921504606846975

1 class maxVal + 1
> 1152921504606846976

(1 class maxVal + 1) class
> LargePositiveInteger
```

Parentheses take Precedence!

```
0@0 extent: 100@100 bottomRight
> Message not understood
> 100 does not understand bottomRight
```

Should use ()

```
(0@0 extent: 100@100) bottomRight
> (aPoint extent: anotherPoint) bottomRight
> aRectangle bottomRight
> 100@100
```

The Price for Simplicity

Only messages:

- +
  - is a message, no precedence
  - can be redefined in domain classes
- Simple
- One limit: no mathematical precedence

No Mathematical Precedence

```
3 + 2 * 10
> 5 * 10
> 50
```

- should be rewritten using parentheses

```
3 + (2 * 10)
> 3 + 20
> 23
```

No Mathematical Precedence

```
1/3 + 2/3
> 7/3 /3
> 7/9
```

- should be rewritten using parentheses

```
(1/3) + (2/3)
> 1
```

Summary

- Three kinds of messages: unary, binary and keywords
- (...) > unary > binary > keywords
- Then from left to right
- There is no mathematical precedence because mathematical operations are plain messages
- Arguments are placed inside message structure:
  - 2 between: 0 and: 5 (the message is between:and:)

A course by

and

in collaboration with