Ruby 2.7-preview1

by @jimmynguyc

Hi

- Jimmy Ngu (@jimmynguyc)

- Engineering Team Lead @ RapidRiver Software

- KL Ruby Brigade, RubyConf MY (not anymore)
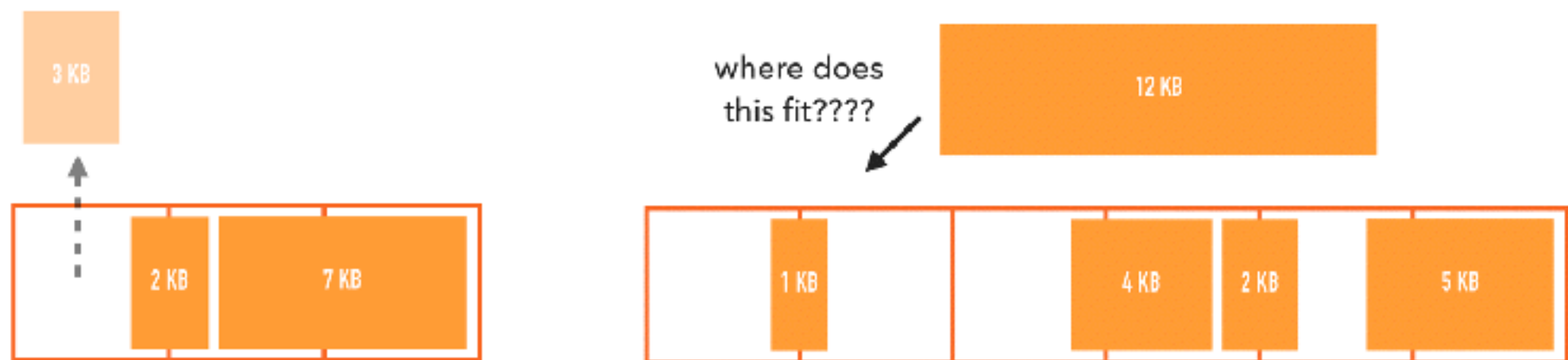
- We are hiring !! :D

New Features

- REPL improvement

- GC Compaction

- Pattern Matching [experimental]

- Numbered parameters

- Method reference operator (.:)

- Others

REPL Improvement

- Basically irb

- Syntax highlighting

- multi-line editing

GC Compaction

- Memory Fragmentation issue

- Hongli Lai's article - https://www.joyfulbikeshedding.com/blog/2019-03-14-what-causes-ruby-memory-bloat.html

GC Compaction

- Use jemalloc

- Set magical environment variable
  MALLOC_ARENA_MAX=2

- Hack GC to use malloc_trim()

GC Compaction

- GC.compact

- Reduce memory defragmentation

- Reduce heap page usage

- Reduce memory usage

Pattern Matching

- case .. when + multiple assignments

- case .. in .. end

```
case [0, [1, 2, 3]]
in [a, [b, *c]]
  puts a #=> 0
  puts b #=> 1
  puts c #=> [2,3]
end
```

- Syntax

```
case expr
in pattern [if|unless condition]
  ...
in pattern [if|unless condition]
  ...
else
  ...
end
```

Numbered Parameters

- Example

```
[1, 2, 3].map { @1 + 3 }
=> [4, 5, 6]
```

- Only valid in blocks

Method **R**eferencing Operator

- Better version of Symbol#to_proc

```
[1,2,3].map { |n| 2 * n }
# => [2, 4, 6]
[1,2,3].map(&2.:*)
# => [2, 4, 6]
```

Others

- Beginless/Endless range

```
ary[1..] # identical to ary[0..3]
```

- Enumerable#tally

```
["a", "b", "c", "b"].tally
#=> {"a"=>1, "b"=>2, "c"=>1}
```

- More JIT-ed code

- https://www.ruby-lang.org/en/news/2019/05/30/ruby-2-7-0-preview1-released/

What's Coming

- https://bugs.ruby-lang.org/projects/ruby-trunk/roadmap

- Pipeline Operator [experimental & highly controversial]

- Preview 2 (summer), Preview 3 (fall), December (RC + Release)

Q&A

Thank You