本当は怖い ObjectSpace.each_object

nari(中村)

ネットワーク応用通信研究所

自己紹介

自己紹介

- nariと申します
- 九州Ruby01会議でGCについて喋ったものです
- はてなで「I am Cruby」というブログ
- 福岡出身の島根県民
  - 島根は鳥取の左です！
  - （大事な事なので二回言いました）

ObjectSpace.each_object

超基本的な疑問

ObjectSpace.each_object って何？

Heap内にある
全てのオブジェクト(一部除く)に対して

```
ObjectSpace.each_object
{|o| puts o }
```

```
ObjectSpace.each_object
{|o| puts o }
```

引数に渡した
ブロックの処理を行う

例えば
こんな処理が書ける

Heap内のオブジェクトを全てpする

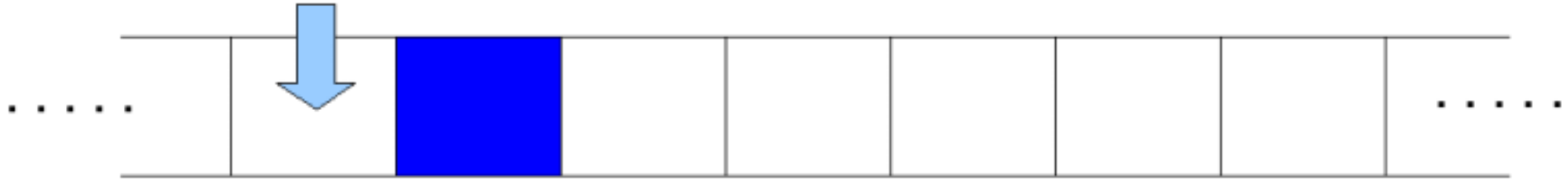
特定のクラスのみをダンプする.

Heap内のクラス数の統計を取る

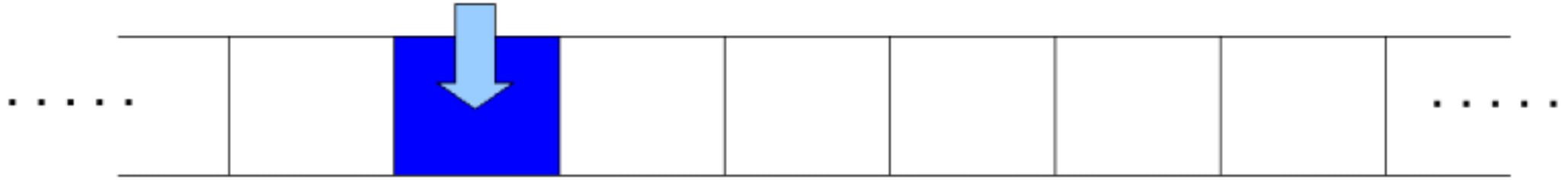
こ，これは便利！

仕組み

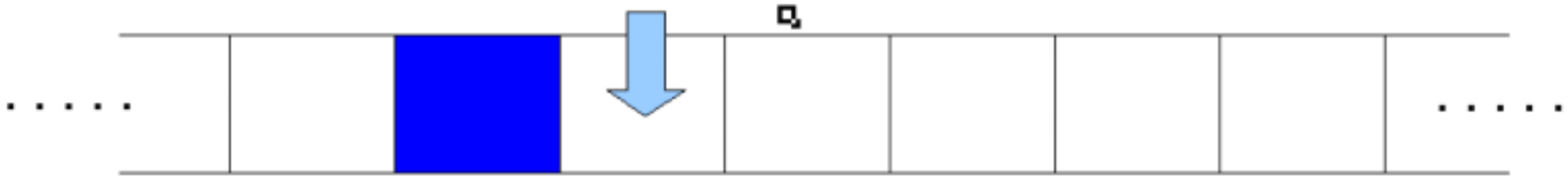
単純にHeap内を一つ一つなめていく



オブジェクトがあれば処理を評価



次へ



なるほど. . .

と言う事は. . .

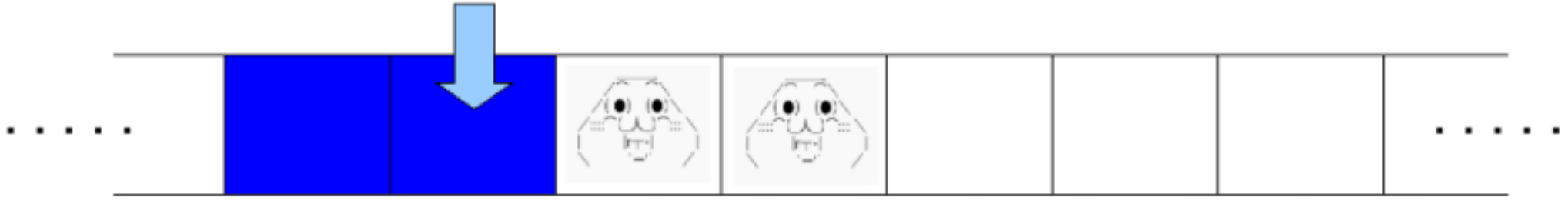
こ，こんなコードも動く？

デモ

Ruby内部では

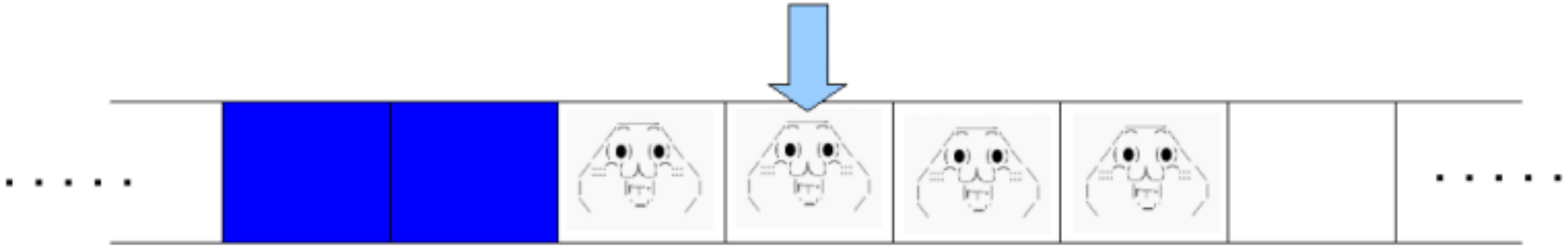
処理を評価＆やる夫生成

次へ，処理を評価＆やる夫生成

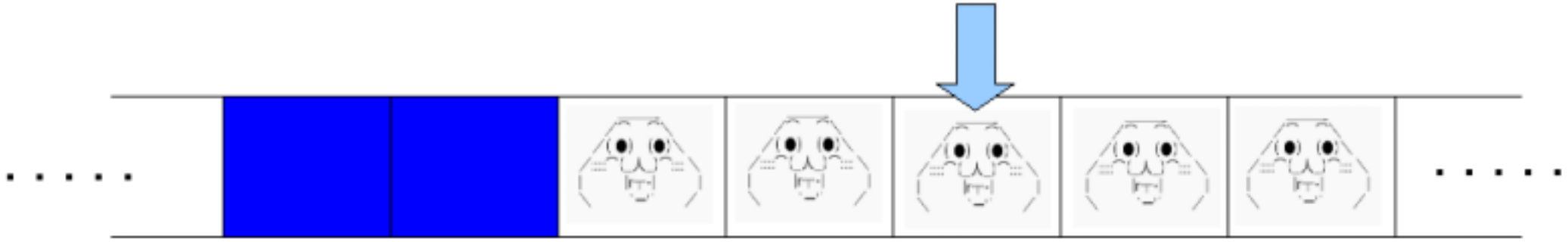


次へ，やる夫登場＆やる夫生成

次へ，やる夫登場＆やる夫生成

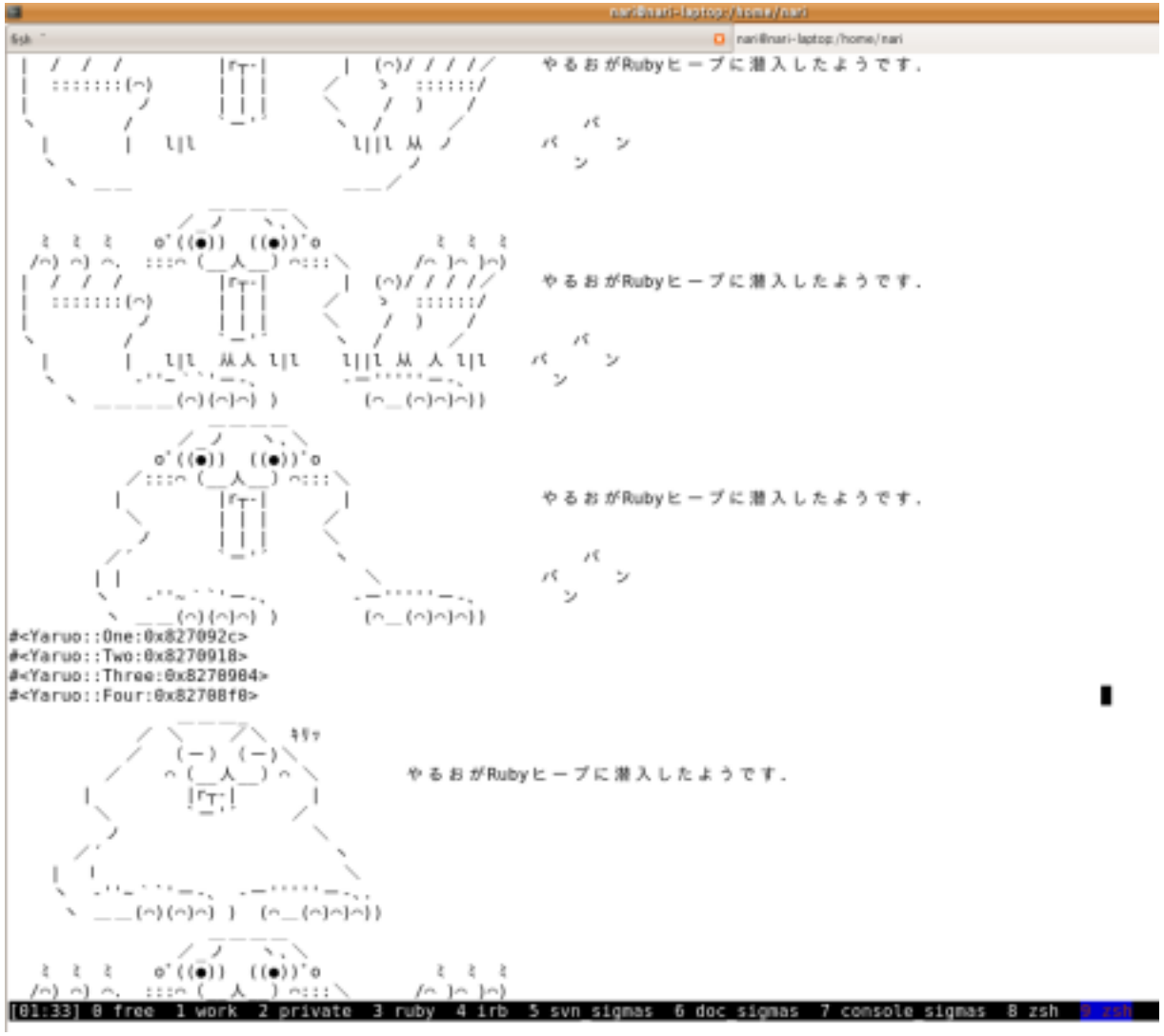


次へ，やる夫登場＆やる夫生成



結果

やる夫が増殖！


やるおがRubyヒープに潜入したようです．
#<Yaruo::One:0x827092c>
#<Yaruo::Two:0x8270918>
#<Yaruo::Three:0x8270904>
#<Yaruo::Four:0x82708f8>
[01:33] 0 free 1 work 2 private 3 ruby 4 irb 5 svn sigmas 6 doc sigmas 7 console sigmas 8 zsh 9 zsh

こっ，これは完全に
Ruby暗黒面ですね！

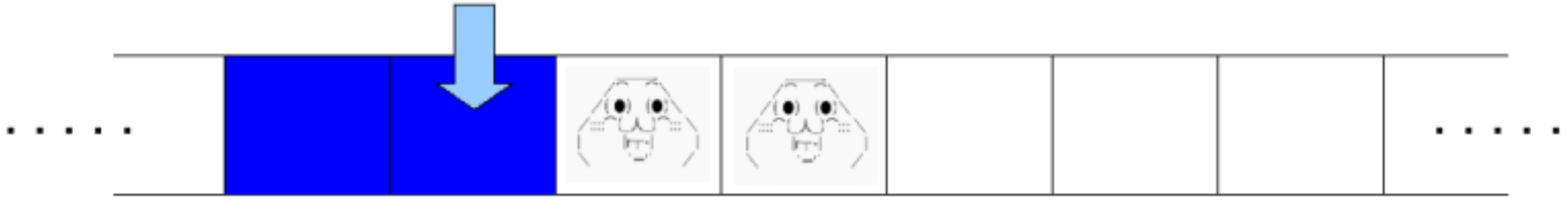
ここで疑問

『これって，処理中に
オブジェクトを生成しても何
でフリーズしないの？』

つまりさっきの様に

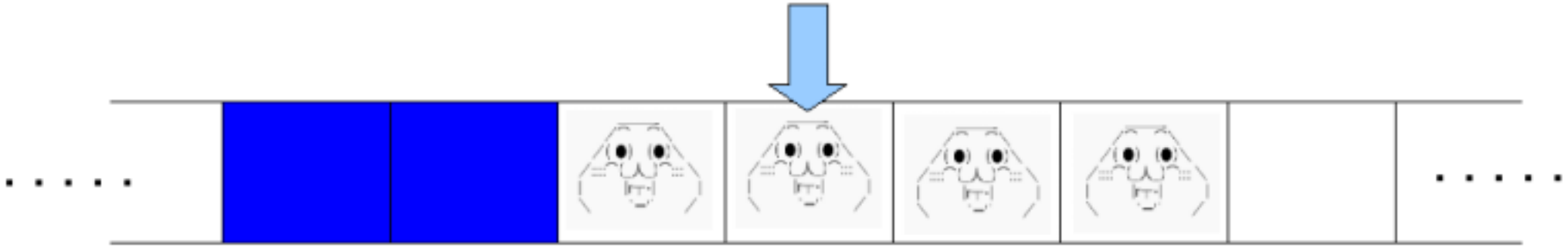
処理時にオブジェクト生成

処理時にオブジェクト生成

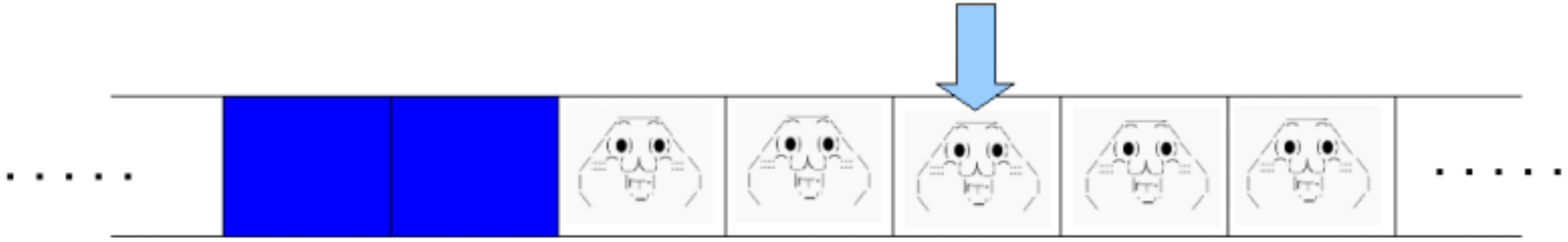


処理時にオブジェクト生成

処理時にオブジェクト生成



処理時にオブジェクト生成



無限ループへ．．．

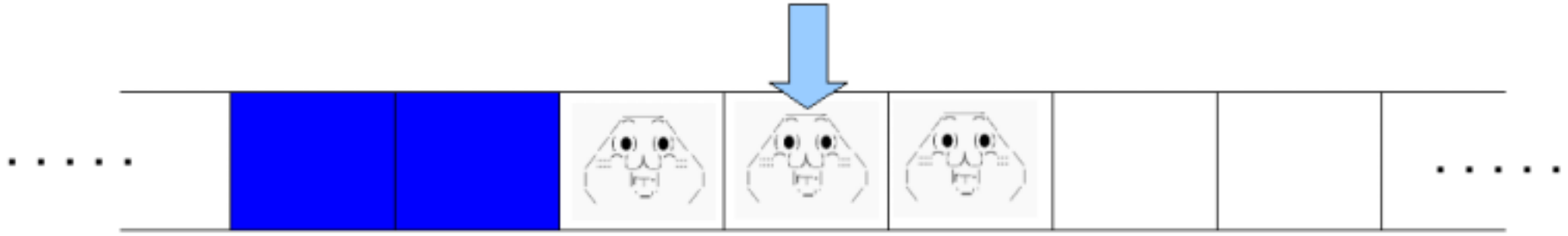
って事になぜならないのか？

理由

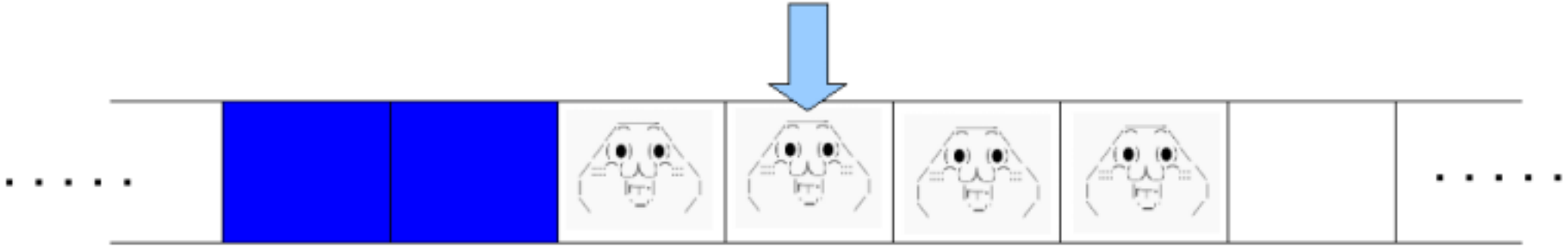
GC

可愛いよGC！

オブジェクトがあれば処理を評価

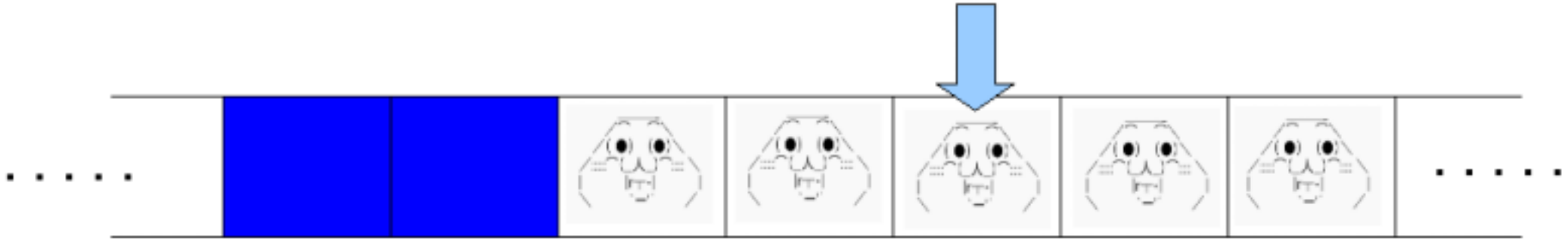


その処理内でオブジェクト生成



そのオブジェクトで処理を評価

その処理内でオブジェクト生成



ヒープがいっぱいになったらGC発動！

ゴミが回収されて無限ループ回避

GCによって無限ループを
回避していた.

なのでGC.disable=trueにすると

デモ

返ってこない！

GCが動かないので無限にオブジェクトが生成されてしまった

ちなみにさっきのHeap内の
統計を取ったものは

正確な値じゃない

デモ

このような場合，Ruby1.9では
ObjectSpace.count_objects
があります．

最後に

実際にある
悪用例

デンマーク出身のDHHさんの場合

あるクラスのサブクラスを取得する

```
class Object
  def subclasses_of(*superclasses)
    subclasses = []
    superclasses.each do |sup|
      ObjectSpace.each_object(class << sup; self; end) do |k|
        if k != sup &&
          (k.name.blank? ||
             eval("defined?(::#{k}) &&
               ::#{k}.object_id == k.object_id"))
          subclasses << k
        end
      end
    end
    subclasses
  end
end
```

使うときは

```
class Fizz; end
class Buzz < Fizz; end
class FizzBuzz < Fizz; end

subclasses_of(Fizz) #=> [Buzz, FizzBuzz]
```

これだけの為にヒープ全てをブン回す！！！

ActiveSupport
に入ってます.

まとめ

- ObjectSpace.each_objectは完全な暗黒面です.
- 大事な場面では使わないようにしましょう.
- どうでもいいような所で使いましょう.
  - 例えばデバッグとかデバッグとか

ご静聴
ありがとうございました.