Go语言在小米抢购系统中的应用

——无用之用，兼谈一些小众编程语言

主讲：韩祝鹏 (hanzhupeng@gmail.com)

开场与自我介绍

- 韩祝鹏 (albertlee)
- 小米网 （2010年7月 — 今）
- 小众编程语言业余爱好者

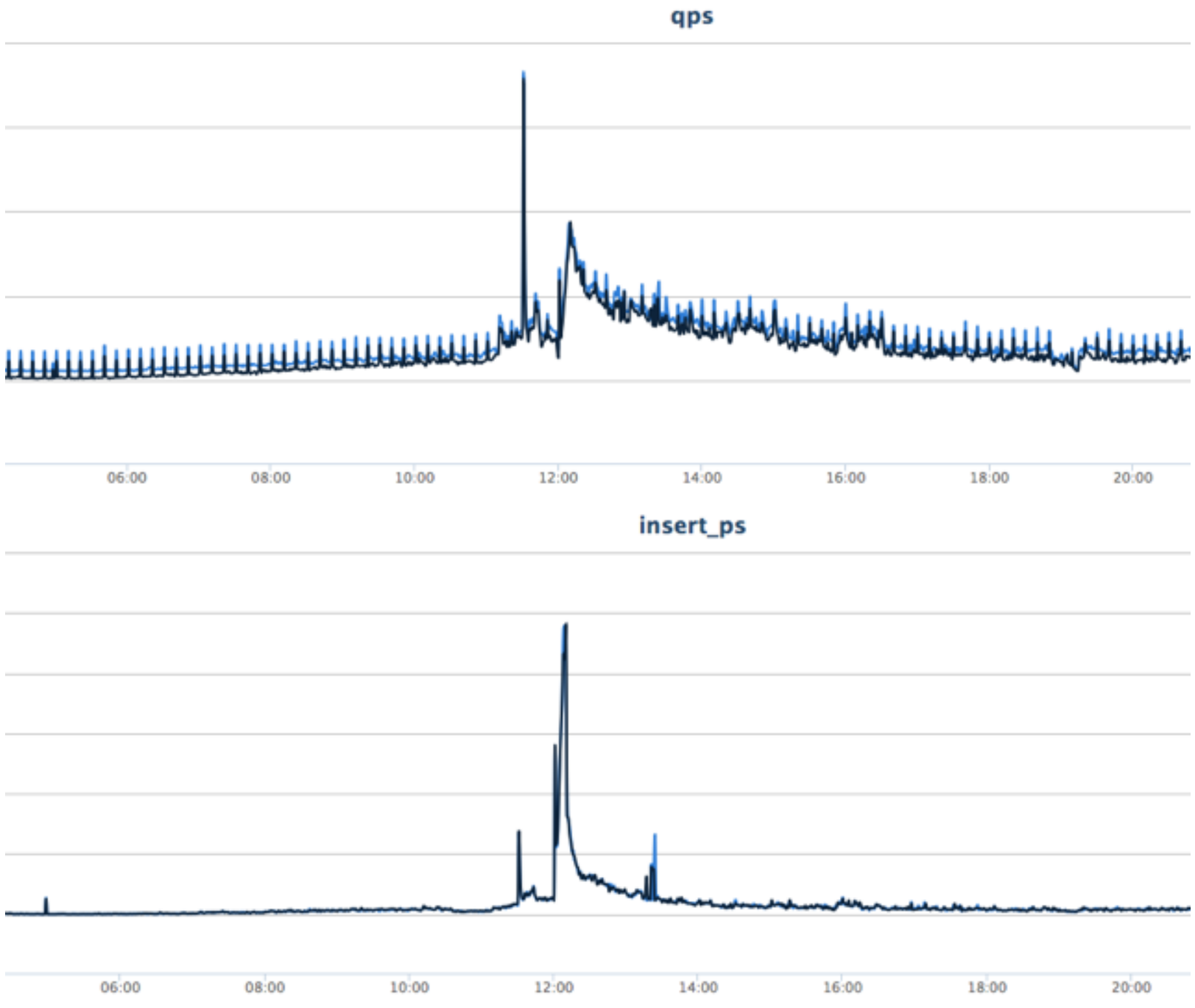
内容提纲

- 小米网大型秒杀系统设计
- Go语言在大秒系统中的使用与优化
- 高并发系统设计的思路
- 介绍几个小众的编程语言
- Just for fun

抢购应用的场景

- 开发时间限制
- 突发海量并发请求
- 数据准确性要求：不能超卖、购买数量限制
- 对失败的容忍基本为零

瞬时高并发


qps
06:00
08:00
10:00
12:00
14:00
16:00
18:00
20:00
insert_ps
06:00
08:00
10:00
12:00
14:00
16:00
18:00
20:00

抢购系统历史

- 基于PHP+Mysql同步请求的系统
- 基于PHP和日志异步处理的系统
- 基于Go的大秒系统

基于PHP+Mysql

- 数据一致性好
- 难于扩展，在突发大流量下，会很惨烈
- 只在2011年第一次预约时用过一次

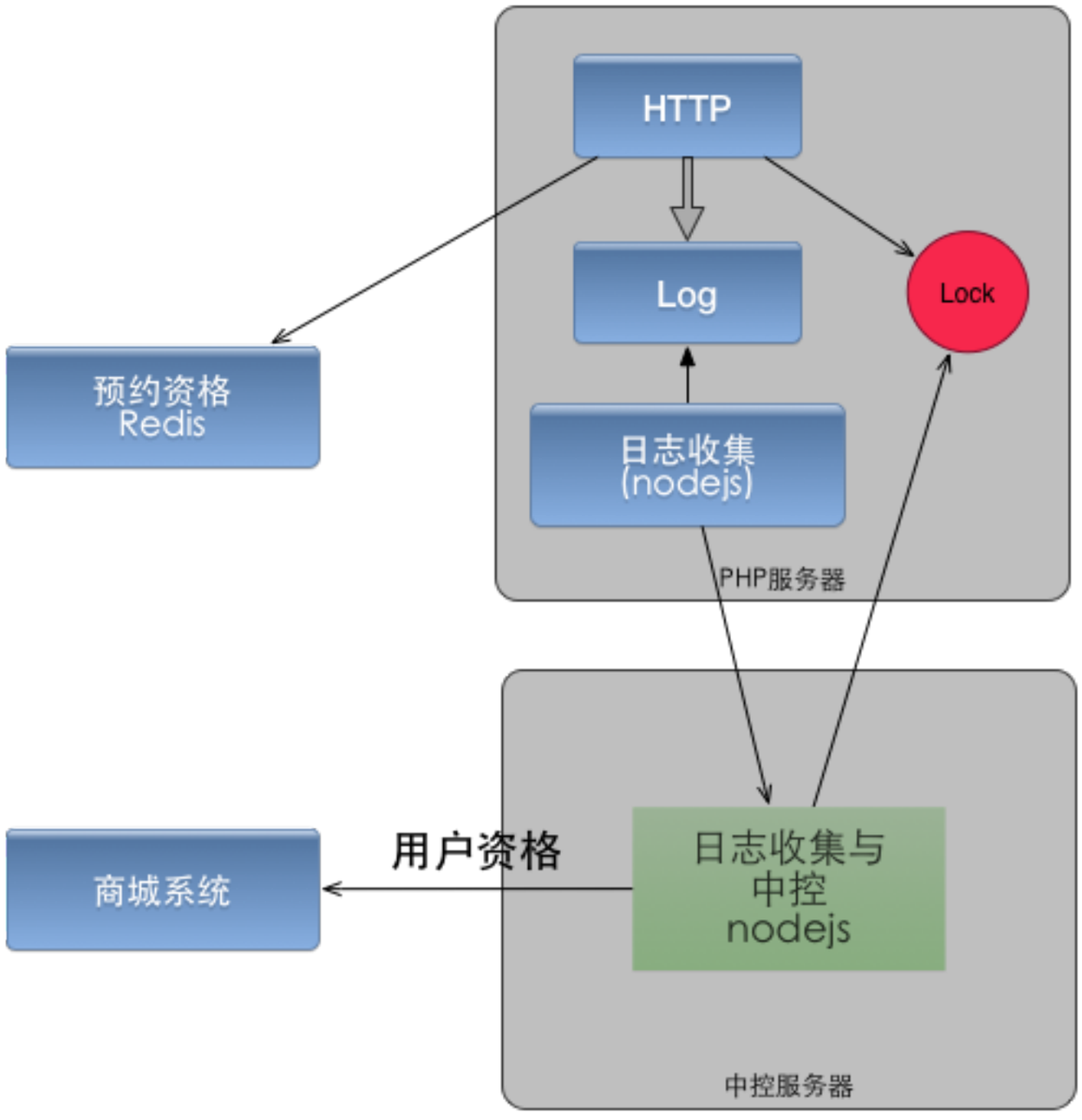
基于PHP+文件锁的系统

- PHP层通过文件锁判断结果，写日志，返回结果
- 日志异步集中处理
- 集中记数，操作缺货开关
- 抢到资格的用户列表通过消息队列传入商城系统
- 数量控制不精确，放弃部分数据一致性，获取性能

基于PHP+文件锁的系统

一周内能快速搞定!

基于PHP+文件锁的系统



基于Go语言的大秒系统

- 重写!
- 业务逻辑越来越复杂，急需管理
- 对更精细控制能力的需求

为什么选择Go?

- 语言偏好
  - 语法标准，相对简单
  - 编译型，强类型，GC
- Goroutine 并发能力
- 编译成可执行文件，部署简单
- 系统的消息通道结构与Go语言风格内在相似性

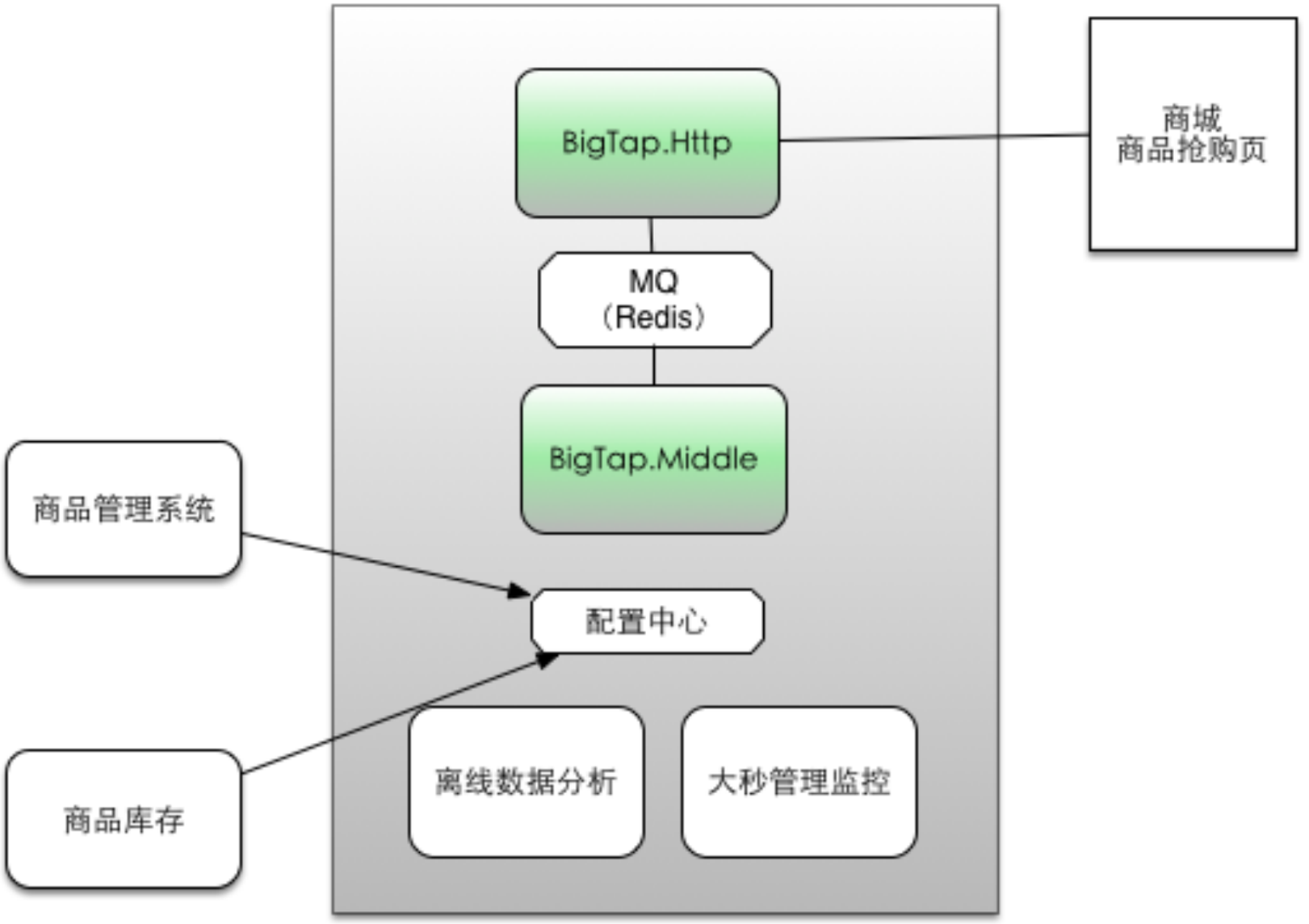
为什么选择Go?

- 非技术因素
  - 应用核心逻辑较为固定，不需要频繁改动
    - 在正式项目开始前半年，已经做过尝试
- 开发人员上手容易
- 老同事许式伟的推广与榜样作用

why not X?

- Erlang: 学习成本大，不确定性大。抱持开放态度
- PHP：我希望有一个简单的基于内存的数据管理
- C/C++：没信心快速搞定
- Python：对性能有所担忧，开发快速的收益长期看难以抵消服务器成本

BigTap系统框架关系



大秒系统的设计

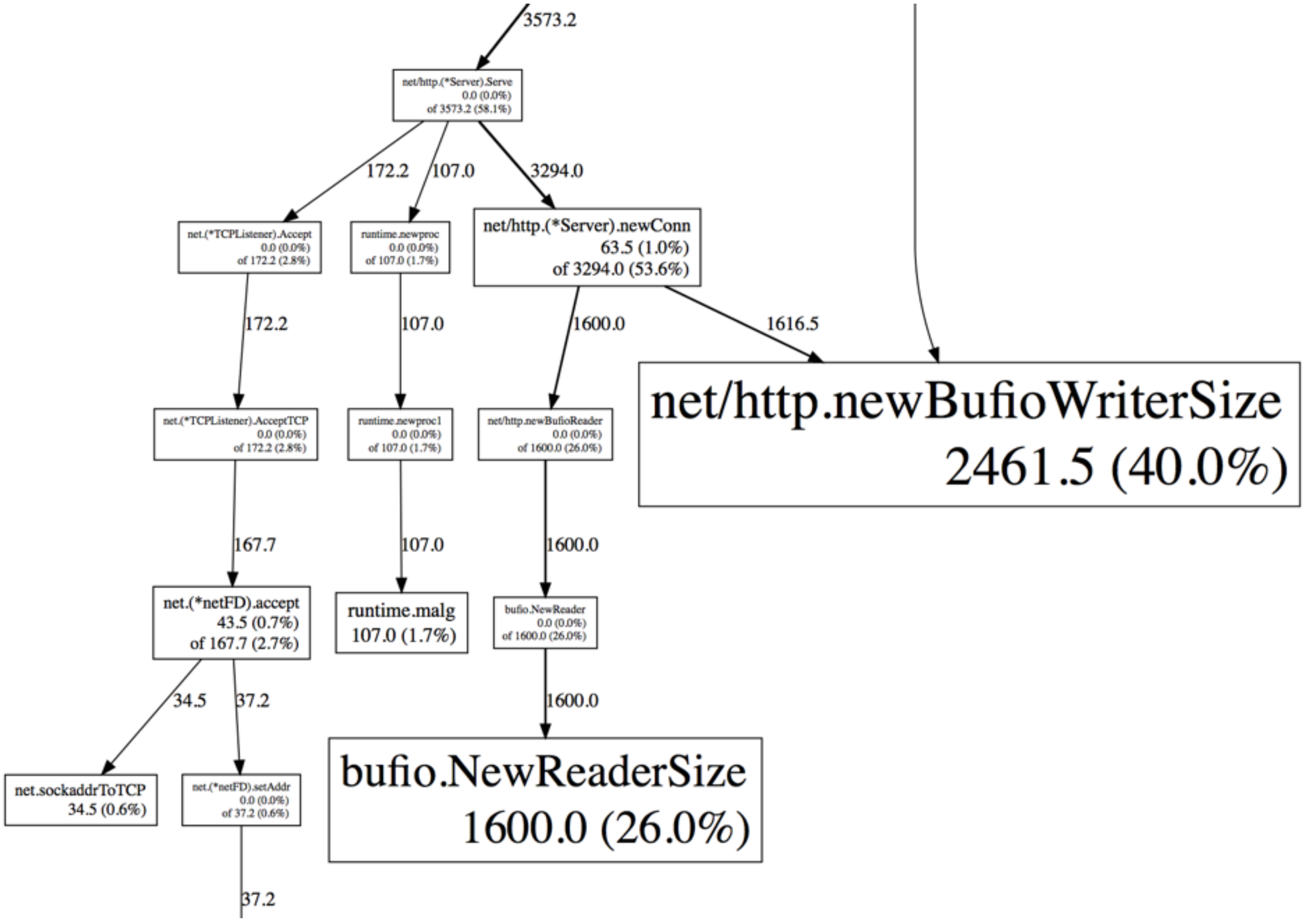
- Go语言开发前端HTTP层及middle层
- http层与middle层之间通过Redis做消息队列
- http层用于维持大量连接，保护middle层
- http层通过维护大量长连接，减少带宽使用
- middle层处理放行逻辑判断

大秒系统的设计

- 用户的请求随机发送到HTTP服务器上
- 通过消息队列将用户请求汇聚到同一台middle服务器
- 系统的消息通道结构与Go语言风格内在相似性
- 系统结构的设计与语言风格的相互影响

针对Go语言的性能优化

- 使用 Go 1.3 ， 提升GC性能
- 使用 Go 1.2 定制HTTP包
  - 增加写缓存和读缓存的通道长度
  - 创建连接时，减少读写缓存的长度
- 尽量减少heap上的短生命周期对象（避免频繁申请内存，引起GC问题）
- 实际效果：单机最大连接数增长了一倍


3573.2
net/http.(*Server).Serve
0.0 (0.0%)
of 3573.2 (58.1%)
172.2
107.0
3294.0
net.(*TCPListener).Accept
0.0 (0.0%)
of 172.2 (2.8%)
runtime.newproc
0.0 (0.0%)
of 107.0 (1.7%)
net/http.(*Server).newConn
63.5 (1.0%)
of 3294.0 (53.6%)
172.2
107.0
1600.0
1616.5
net.(*TCPListener).AcceptTCP
0.0 (0.0%)
of 172.2 (2.8%)
runtime.newproc1
0.0 (0.0%)
of 107.0 (1.7%)
net/http.newBufioReader
0.0 (0.0%)
of 1600.0 (26.0%)
net/http.newBufioWriterSize
2461.5 (40.0%)
167.7
107.0
1600.0
net.(*netFD).accept
43.5 (0.7%)
of 167.7 (2.7%)
runtime.malg
107.0 (1.7%)
bufio.NewReader
0.0 (0.0%)
of 1600.0 (26.0%)
34.5
37.2
1600.0
net.sockaddrToTCP
34.5 (0.6%)
net.(*netFD).setAddr
0.0 (0.0%)
of 37.2 (0.6%)
bufio.NewReaderSize
1600.0 (26.0%)
37.2

增加写缓存和读缓存的通道长度

- 使用chan作为资源池

修改后：

```
// TODO: use a sync.Cache instead
var (
        bufioReaderCache    = make(chan *bufio.Reader, 1000000)
        bufioWriterCache2k  = make(chan *bufio.Writer, 4)
        bufioWriterCache32B = make(chan *bufio.Writer, 1000000)
)
```

修改前：

```
// TODO: use a sync.Cache instead
var (
        bufioReaderCache    = make(chan *bufio.Reader, 4)
        bufioWriterCache2k = make(chan *bufio.Writer, 4)
        bufioWriterCache4k = make(chan *bufio.Writer, 4)
)
```

bufioWriter由默认4k减少为32字节

- 减少内存使用
- 减少写buffer操作

```
// Create new connection from rwc.
func (srv *Server) newConn(rwc net.Conn) (c *conn, err error) {
        c = new(conn)
        c.remoteAddr = rwc.RemoteAddr().String()
        c.server = srv
        c.rwc = rwc
        if debugServerConnections {
                c.rwc = newLoggingConn("server", c.rwc)
        }
        c.sr = liveSwitchReader{r: c.rwc}
        c.lr = io.LimitReader(&c.sr, noLimit).(*io.LimitedReader)
        br := newBufioReader(c.lr)
        // bw := newBufioWriterSize(c.rwc, 4<<10)
        bw := newBufioWriterSize(c.rwc, 32)
        c.buf = bufio.NewReadWriter(br, bw)
        return c, nil
}
```

bufioReader 由默认4k减少为1k

```go
func newBufioReader(r io.Reader) *bufio.Reader {
        select {
        case p := <-bufioReaderCache:
                p.Reset(r)
                return p
        default:
                //return bufio.NewReader(r)
                return bufio.NewReaderSize(r, 1<<10)
        }
}
```

- 连接结束时，将buf放回资源池
- 无法将资源放回资源池时，显式设置为nil
- 优化GC （具体效果待考）

```go
// default: make br = nil good for GC
func putBufioReader(br *bufio.Reader) {
        br.Reset(nil)
        select {
        case bufioReaderCache <- br:
        default:
                br = nil
        }
}
```

```go
func putBufioWriter(bw *bufio.Writer) {
        bw.Reset(nil)
        select {
        case bufioWriterCache(bw.Available()) <- bw:
        default:
                bw = nil
        }
}
```

改小buf，减少写buf的操作

```go
func (b *Writer) Write(p []byte) (nn int, err error) {
        for len(p) > b.Available() && b.err == nil {
                var n int
                if b.Buffered() == 0 {
                        // Large write, empty buffer.
                        // Write directly from p to avoid copy.
                        n, b.err = b.wr.Write(p)
                } else {
                        n = copy(b.buf[b.n:], p)
                        b.n += n
                        b.flush()
                }
                nn += n
                p = p[n:]
        }
        if b.err != nil {
                return nn, b.err
        }
        n := copy(b.buf[b.n:], p)
        b.n += n
        nn += n
        return nn, nil
}
```

bufio.go

对突发大流量系统设计的思考

- 独立小系统
- 分层
- 多用内存，少用IO，数据的局部性
- 小心防止雪崩效应
- 对队列的长度、生产消费速度要格外小心
- 对数据一致性的考虑：是否需要严格、实时

编程要从娃娃抓起

图图，你都三天了，Hello World都还不会写，不能输在起跑线上啊。

图图，你都三天了，Hello world都还不会写，不能输在起跑线上啊。说你两句还不理我了，我这是为了你好啊。离高考只剩六千多天了。

收起 | 查看大图 | 向左旋转 | 向右旋转

1月26日 13:08 来自 小米手机4

阅读 116.2万 推广 | 转发 1971 | 评论 431 | 1758

几个有意思的小众语言

- LOGO
- Smalltalk Squeak
- Scratch
- LISP，Prolog
- Haskell
- Forth

我对语言的偏好

- 基本语法足够简单，正交性
- 语言可扩展
- 明确的价值观
- 如果能有交互式的环境就更好了

LOGO

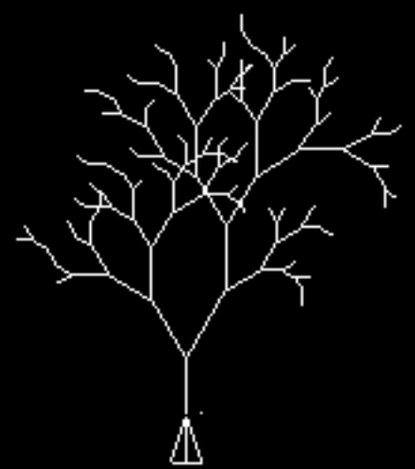
- 最开始就是设计成LISP的方言
- 不只是学习编程，更是学习如何学习

LOGO

Berkeley Logo

```
Welcome to Berkeley Logo version 6.0(exported)
? to tree :size
>   if :size < 5 [stop]
>   fd :size
>   lt 30 tree :size * (((random 5)+5)/10)
>   rt 60 tree :size * (((random 5)+5)/10)
>   lt 30 bk :size
> end
tree defined
?
```

Berkeley Logo



```
? cs bk 20
? tree 40
?
```

```
? show filter [? > 0] [3 -5 2 4 -1 -7 9]
[3 2 4 9]
? show filter [(remainder ? 2) = 1] [1 2 3 1]
[1 3 1]
```

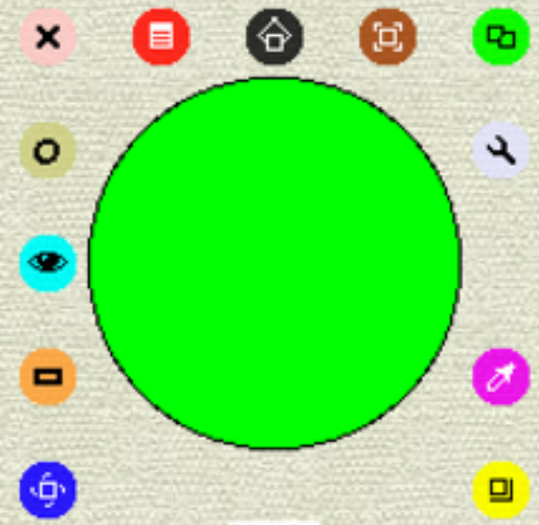
Smalltalk

- 语言设计的简单性与可扩展性
- IDE的始祖
- Squeak , Etoys , Pharo
- 开发环境与运行环境一体
- 开发过程就是在一个不停生长的过程

Squeak4.5-13680.image

Projects Tools Apps Extras Windows Help Search: 4:17:45 pm

Objects



a CircleMorph(1716)

self
all inst vars
bounds
owner
submorphs
fullBounds
color
extension
borderWidth
borderColor

(Color r: 0.03 g: 1 b: 0.03)

Workspace

```
c := CircleMorph new openInWorld
c inspect
```

Tools

Supplies Widgets

Squeak4.5-13680.image

Projects Tools Apps Extras Windows Help Search Rectangle 4:25:45 pm

Objects

System Browser: Rectangle

Graphics-Primitives	PenPointRecorder	-- all --	center:extent:
Graphics-Text	WarpBlt	instance creation	encompassing:
Graphics-Transformati	Bitmap		fromUser
GraphicsTests-Files	Color		fromUser:
GraphicsTests-Primiti	TranslucentColor		left:right:top:bottom:
GraphicsTests-Text	ColorMap		locateMagnifiedView:sc
Morphic-Balloon	LayoutFrame		merging:
Morphic-Basic	Point		origin:corner:
Morphic-Basic-NewCu	Rectangle		origin:extent:
Morphic-Borders	Quadrangle		originFromUser:
Morphic-Collections-A			originFromUser:grid:

instance ? class

browse senders implementors versions inheritance hierarchy vars source

```
left: leftNumber right: rightNumber top: topNumber bottom: bottomNumber
    "Answer an instance of me whose left, right, top, and bottom coordinates
    are determined by the arguments."

    ^ self basicNew setOrigin: leftNumber @ topNumber corner: rightNumber @
bottomNumber
```

Workspace

```
c := CircleMorph new openInWorld
c inspect
c bounds: (Rectangle left:100 right:250 top:250 bottom:450)
```

Squeak

Tools

Supplies Widgets

Squeak4.5-13680.image

Projects Tools Apps Extras Windows Help Search Integer 6:08:07 pm

Objects

System Browser: Number

Kernel-Chronology	Complex	-- all --	strictlyPositive
Kernel-Classes	Magnitude	arithmetic	stringForReadout
Kernel-Exceptions	Number	comparing	tan
Kernel-Exceptions-Kernel	Float	converting	tanh
Kernel-Methods	Fraction	filter streaming	to:
Kernel-Models	Integer	intervals	to:by:
Kernel-Numbers	LargePositiveInteger	mathematical functions	to:by:do:
Kernel-Numbers-Exceptions	LargeNegativeInteger	printing	to:do:
Kernel-Objects	SmallInteger	testing	truncateTo:
Kernel-Processes	ScaledDecimal	truncation and round off	truncated
Kernel-Processes-Variables	NumberParser	*Etoys-tiles	veryDeepCopy
Kernel-Pools	FORTRANNumberParser	*Etoys-vocabulary	vocabularyDemanded
KernelTests-Chronology		*Tools-Debugger	week
		*Graphics-converting	

instance ? class

browse senders implementors versions inheritance hierarchy vars source

```
to: stop do: aBlock
    "Normally compiled in-line, and therefore not overridable.
    Evaluate aBlock for each element of the interval (self to: stop by: 1)."
    | nextValue |
    nextValue := self.
    [nextValue <= stop]
        whileTrue:
            [aBlock value: nextValue.
            nextValue := nextValue + 1]
```

Squeak

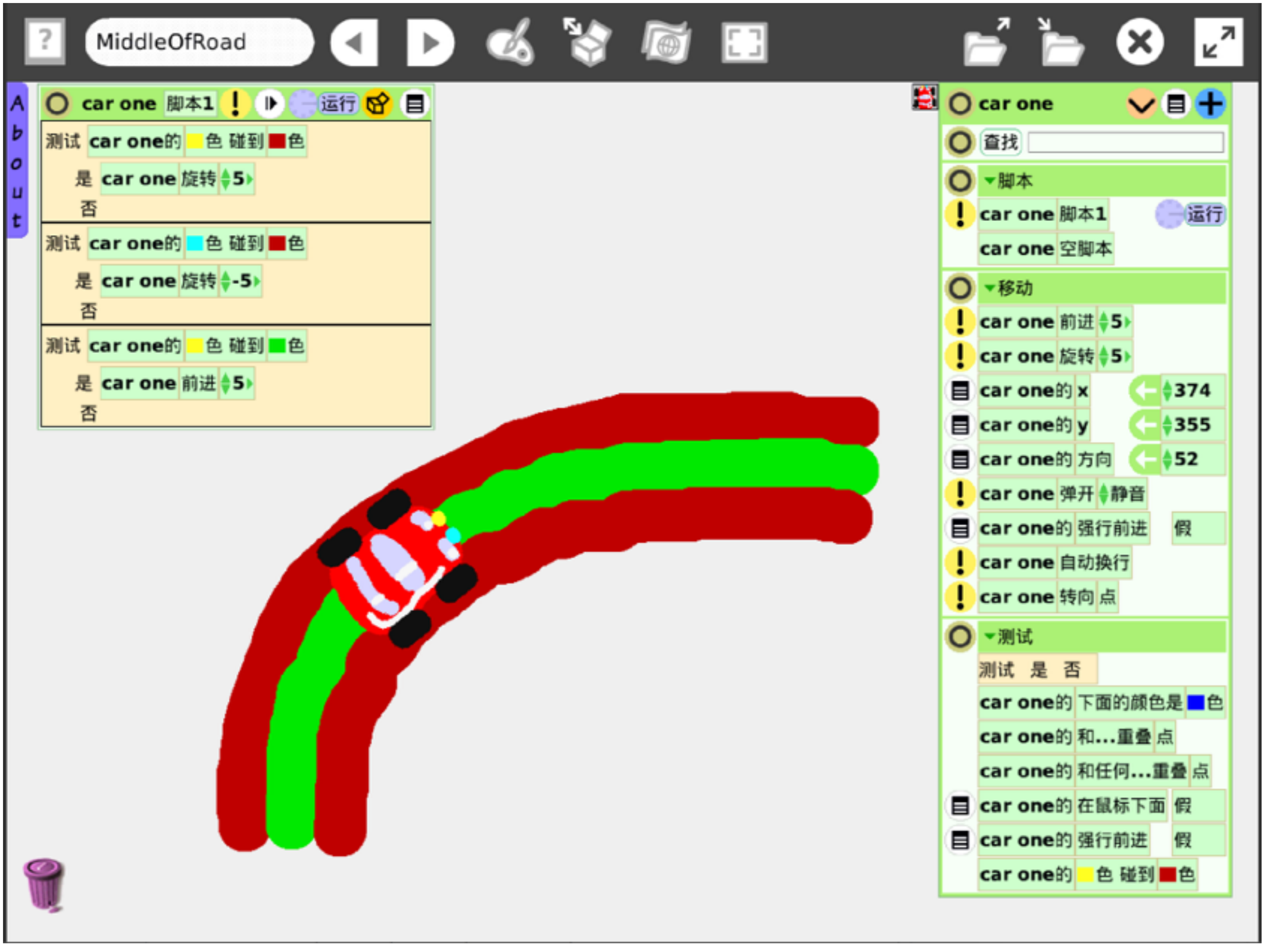
Workspace

```
1 to: 5 do: [:x | Transcript show: x]
```

Transcript

```
12345
```

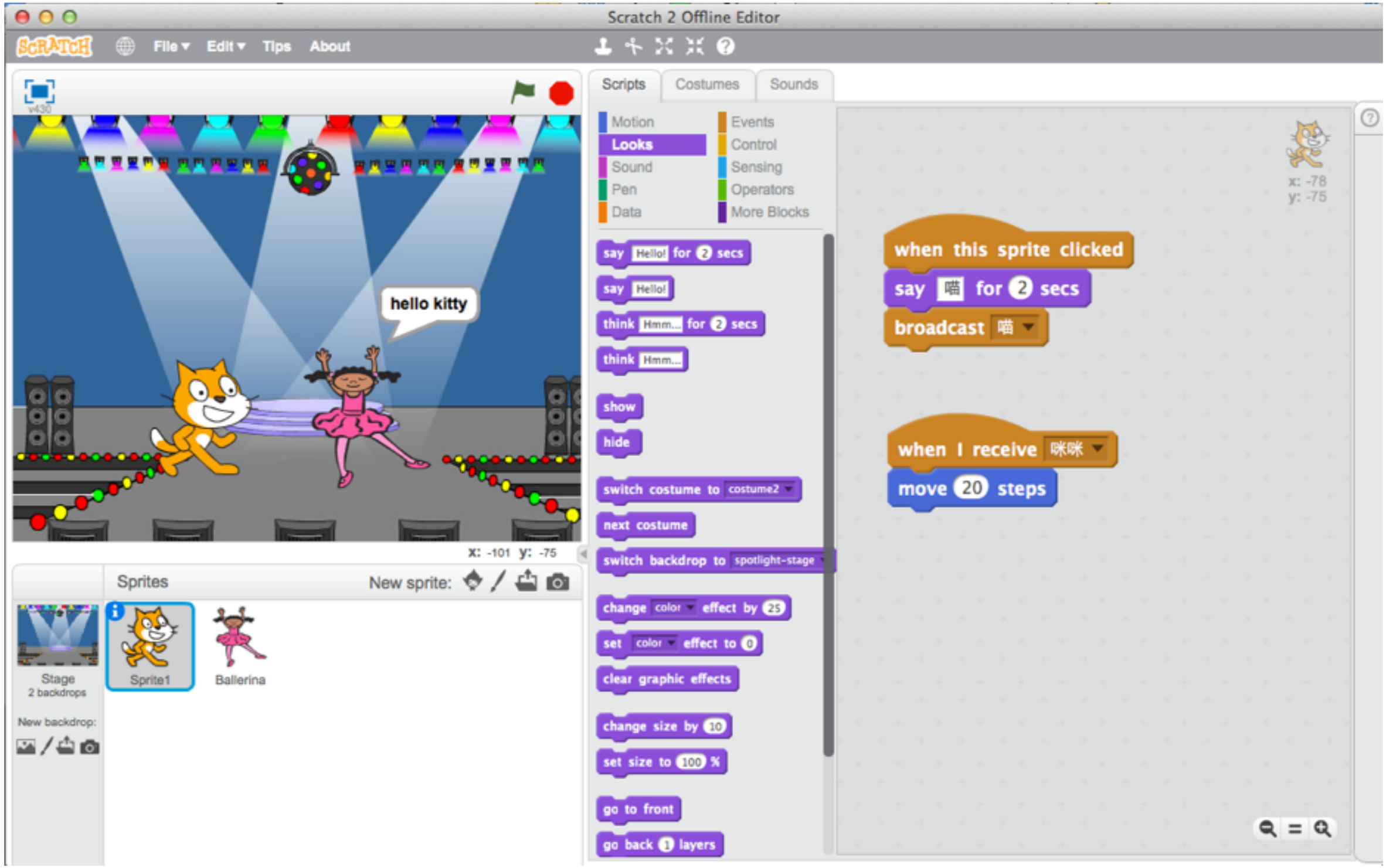
EToys


MiddleOfRoad
About
car one 脚本1 运行
测试 car one的 色 碰到 色
是 car one 旋转 5
否
测试 car one的 色 碰到 色
是 car one 旋转 -5
否
测试 car one的 色 碰到 色
是 car one 前进 5
否
car one
查找
脚本
car one 脚本1 运行
car one 空脚本
移动
car one 前进 5
car one 旋转 5
car one的 x 374
car one的 y 355
car one的 方向 52
car one 弹开 静音
car one的 强行前进 假
car one 自动换行
car one 转向 点
测试
测试 是 否
car one的 下面的颜色是 色
car one的 和...重叠 点
car one的 和任何...重叠 点
car one的 在鼠标下面 假
car one的 强行前进 假
car one的 色 碰到 色

Scratch

- 图形化的创作工具
- 互联网时代的LOGO
- 在线协作社区 http://scratch.mit.edu （被墙了）
- 把世界上最有创造力的孩子们（除了墙内的孩子们）连接在一起，给他们一个强大的工具，你想象不到他们会创造出什么！

Scratch


Scratch 2 Offline Editor
SCRATCH
File
Edit
Tips
About
hello kitty
x: -101 y: -75
Sprites
New sprite:
Stage
2 backdrops
New backdrop:
Sprite1
Ballerina
Scripts
Costumes
Sounds
Motion
Looks
Sound
Pen
Data
Events
Control
Sensing
Operators
More Blocks
say Hello! for 2 secs
say Hello!
think Hmm... for 2 secs
think Hmm...
show
hide
switch costume to costume2
next costume
switch backdrop to spotlight-stage
change color effect by 25
set color effect to 0
clear graphic effects
change size by 10
set size to 100 %
go to front
go back 1 layers
x: -78
y: -75
when this sprite clicked
say 喵 for 2 secs
broadcast 喵
when I receive 咪咪
move 20 steps

Explore

Projects | Studios

Sort by: Most followers | Past 30 Days

- Featured
- All
- Animations
- Art
- Games
- Music
- Stories

Official Beatles Fan Club

213 1851 1262

1260 Followers! Thanks!!! ---------- Welcome to the Official Beatles Fan Club! Anyone interested may join! We do not except New Scratchers!!! (Must be over 2.5 months) Thanks,...

Last modified: 6 Apr 2015

✎ Scratch Writing Group ✎

403 2329 479

FEATURED (24/03/15)! :D Thanks so much, ST (and anyone who suggested it, if they did)! <3 Welcome to Writing Studio! This studio is for writing projects only. It includes stories,...

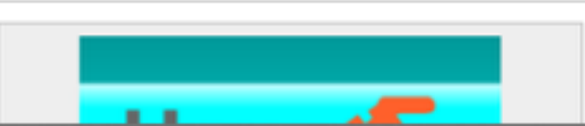
Last modified: 5 Apr 2015

The Studio Of Good Stuff You Want To See!

187 1345 931

Hello, and welcome...

Last modified: 4 Apr 2015



How To...

Music Project on Scratch

https scratch.mit.edu/projects/27853628/#editor

Reader

SCRATCH File Edit Tips About albertlee

Music Project
by lamprale000
v434a

x: 240 y: -180

Sprites New sprite:

Stage
1 backdrop

New backdrop:

John Len... George H... Paul McC... Ringo Starr Beatles_l...

Scripts Costumes Sounds

Remix See project page

Motion
Looks
Sound
Pen
Data
Events
Control
Sensing
Operators
More Blocks

move 10 steps
turn 15 degrees
turn 15 degrees
point in direction 90
point towards
go to x: -188 y: -70
go to mouse-pointer
glide 1 secs to x: -188 y: -70
change x by 10
set x to 0
change y by 10
set y to 0
if on edge, bounce
set rotation style left-right
x position
y position
direction

when space key pressed
show
go to x: -188 y: -70
point in direction 90

x: -188
y: -70

when s key pressed
wait 1.2 secs
forever
switch costume to John Lennon-0
wait 0.5 secs
switch costume to John Lennon 2
wait 0.5 secs

Backpack

这只是小孩的玩意么?

- 运行环境、实时调试、热部署
- LOGO与函数式编程
- 互联网协作创新的试验场
- 从 LOGO - Smalltalk - OLPC - Squeak - Scratch，我们旁观了一代代人不断向着同一个方向前进

小众语言的无用之用

- 思想观念的进化与结合
- 避免变成只有一把锤子的人
- 快速变化，享受技术进步的红利

创业公司?

- 客户端其实没什么选择余地：ObjC, Java, HTML+JS
- 服务端：个人熟悉Python + Go的组合
- 特定领域：R, Haskell, Prolog, Scala……
- 程序员开心才是最大的生产力

爱好者?

- LISP, Scheme, Prolog, Smalltalk, Forth, Haskell
- LOGO, Squeak, Etoys, Scratch, LEGO
- 都很好玩，每种语言都是一种风味不同的食材
- 都是无用的，好玩本身就是最大的回报

The best way to predict the future is to invent it.

—— Alan Kay