

# Golang

## по обе стороны баррикад

Алексей Акулович

Backend разработчик

ВКонтакте

ПРИВЕТ



Devconf

<http://www.devconf.ru>

2017.06.16#3



VK SUMMER CODE

*Hackathon*  
VK.COM

# Хакатон

Сутки на реализацию

Команда до 3х человек

Любая тема

Любая технология

# GopherJS

Транспи́лер Go -> JS

(примерно как Babel из мира JS)

(\*go -> .js + .js.map)

Написан на Go

Компилирует сам себя в JS для работы playground

<https://github.com/gopherjs/gopherjs>

# Делаем игру

Делаем игру

HTML5 canvas/webgl

2D со спрайтами

Мультиплеер

Синхронизация с VK API

## Выбор рендера - ENGi

Поддерживает desktop (GLFW) и web (GopherJS)

Использует WebGL рендер

Очень простое API

Очень малофункционален

<https://github.com/ajhager/engi>



Делаем игру

HTML5 canvas/webgl - GopherJS + ENGi

2D со спрайтами

Мультиплеер

Синхронизация с VK API

Почти вся графика - спиратчена :)



## Random Heroes

Noodlecake Studios Inc

<https://play.google.com/store/apps/details?id=com.noodlecake.randomheroes>

Делаем игру

HTML5 canvas/webgl - GopherJS + ENGi

2D со спрайтами - Random Heroes + GIMP

Мультиплеер

Синхронизация с VK API

# Синхронизация мультплеера

Клиент-серверная архитектура

~20 qps при общении

Возможность работы в браузере

Выбрали Websocket

Делаем игру

HTML5 canvas/webgl - GopherJS + ENGi

2D со спрайтами - Random Heroes + GIMP

Мультиплеер - WebSocket

Синхронизация с VK API

## Синхронизация с VK

“Игровая платформа” <https://vk.com/dev/games>

IFrame приложение

Придется адаптировать JS SDK к GopherJS

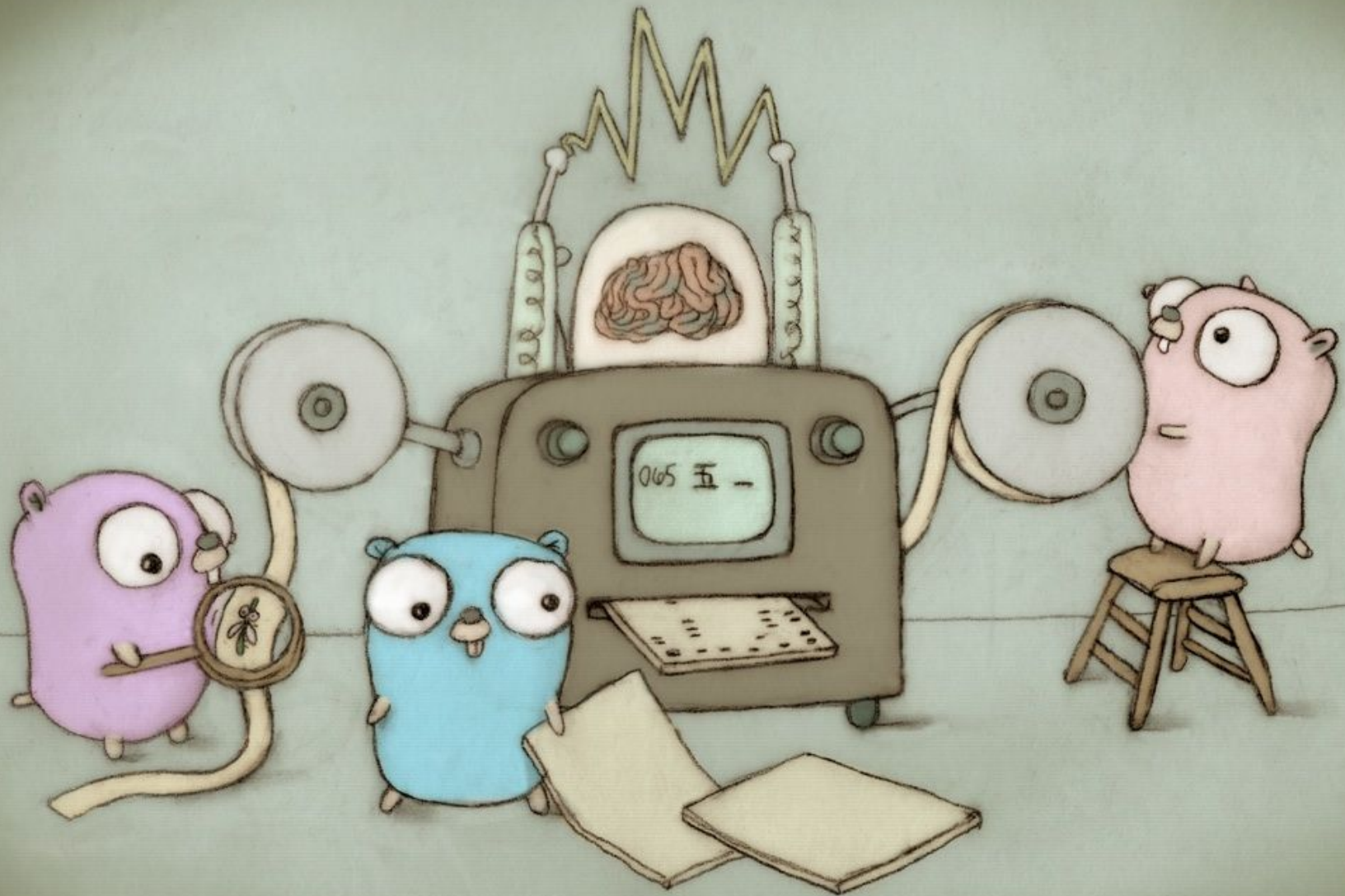
Делаем игру

HTML5 canvas/webgl - GopherJS + ENGi

2D со спрайтами - Random Heroes + GIMP

Мультиплеер - WebSocket

Авторизация через VK API - IFrame + JS SDK





# GopherJS

## GopherJS - типы данных

Go type	JavaScript type	Back to interface{}
bool	Boolean	bool
integers and floats	Number	float64
string	String	string
[]int8	Int8Array	[]int8
[]int32, []int	Int32Array	[]int
[]float64	Float64Array	[]float64
all other slices	Array	[]interface{}
-	instanceof Node	*js.Object
maps, structs	instanceof Object	map[string]interface{}
...	...	...

## GopherJS - stdlib

Всё, кроме: unsafe, syscall, runtime/\*, net/pprof, os, ..., cgo

```
Event struct {  
    Type           EventType           `json:"type"`  
    Payload        json.RawMessage    `json:"pl"`  
    ParsedPayload interface{}         `json:"-"`  
}
```

```
var ev Event  
json.Marshal(ev)
```

## GopherJS - обработка ошибок

**js:**

```
try {  
    // ...  
} catch (err) {  
}
```

**gopherjs:**

```
go func() { recover() }()  
// ...
```

## GopherJS - доступ к DOM

**js:**

```
document.write(`Hello`)  
console.log(`DevConf`)  
document.getElementById('hello').innerHTML = 'world'
```

**gopherjs:**

```
js.Global.Get(`document`).Call(`write`, `Hello`)  
fmt.Println(`DevConf`)  
js.Global.Get(`hello`).Set(`innerHTML`, `world`)
```

## GopherJS - горутины

```
btn := js.Global.Get(`myButton`)  
  
btn.Call(`addEventListener`, `click`, func(){  
    go func() { // обязательно для gopherjs  
        someBlockingFunction()  
    }()  
})
```

## GopherJS - каналы

```
func loadImage(imgUrl string) (*js.Object, error) {
    ch := make(chan error, 1)

    img := js.Global.Get(`Image`).New()
    img.Call(`addEventListener`, `load`, func(*js.Object) {
        go func() { ch <- nil }()
    }, false)
    // img.Call(`addEventListener`, `error`, func...
    img.Set(`src`, imgUrl)

    if err := <-ch; err != nil {
        return nil, err
    }

    return img, nil
}
```

## GopherJS - объявление callback для JS

**gopherjs:**

```
func initJsFields() {
    js.Global.Set(`player`, map[string]interface{}{
        `connectToServer`: PlayerConnectToServer,
    })
}

func PlayerConnectToServer(skinId int, nickname string) {
    menu.Get(`style`).Call(`setProperty`, `display`, `none`)
}
```

**js:**

```
function joinGame() {
    player.connectToServer(charSkin, nickname)
}
```



## GopherJS - websocket

```
var ws *js.Object = js.Global.Get(`WebSocket`).New(serverUrl)

ws.Call(`addEventListener`, `message`, func(ev *js.Object) {
    // ev.Get(`data`).String()
})

ws.Call(`addEventListener`, `error`, func(ev *js.Object) {
    // ev.Get(`message`).String()
})
```

**ENGi**

## ENGi - что есть

Загрузчик ресурсов

Рендер текстурированных спрайтов (WebGL)

Примитивные спрайтовые шрифты

Базовая поддержка звука (HTML5 audio)

Ввод с мышки и клавиатуры

## ENGi - чего нет

Камера

Матричные операции

Анимация

# ENGi

```
import "github.com/ajhager/engi"

type Game struct {
    *engi.Game
    img    engi.Drawable
    batch *engi.Batch
}

func (g *Game) Preload() {
    engi.Files.Add(`img`, `path/to/image.png`)
}

func main() {
    engi.Open(`Hello DevConf`, 800, 600, false, &Game{})
}
```

## ENGi

```
func (g *Game) Setup() {  
    g.img = engi.Files.Image(`img`)  
    g.batch = engi.NewBatch(engi.Width(), engi.Height())  
}  
  
func (g *Game) Render() {  
    g.batch.Begin()  
    g.batch.Draw(g.img, 0, 0, 0.5, 0.5, 1, 1, 45, 0xffffffff, 1)  
    g.batch.End()  
}
```

## Backend

[github.com/gorilla/websocket](https://github.com/gorilla/websocket)



# Синхронизация клиента и сервера



## Синхронизация клиента и сервера

Физику и коллизии рассчитывают и клиент и сервер

Клиент отсылает действия игрока серверу

Сервер рассылает обновления всем

Сервер авторитарен

# Синхронизация с VK

## Синхронизация с VK

```
vk := js.Global.Get(`VK`)
vk.Call(`init`, func(){
    vk.Call(`api`, `users.get`, params, usersGetCallback)
}, func() {
    // error
}, `5.64`)
```

```
func usersGetCallback(data map[string]interface{}) {
    if response, ok := data[`response`]; ok {
        // ...
    }
}
```

## Синхронизация с VK

Исходный

```
data map[string]interface{}
```

Превращается в

```
data map[string] []map[string]interface{}
```

```
response, ok := data[`response`]
```

```
responseList, ok := response.([]interface{})
```

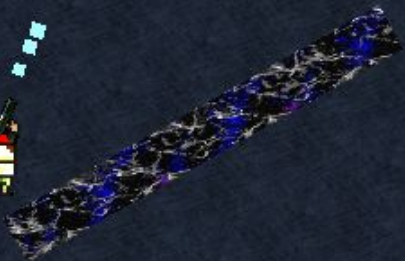
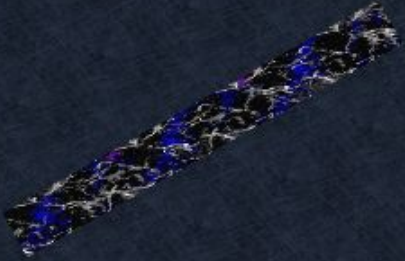
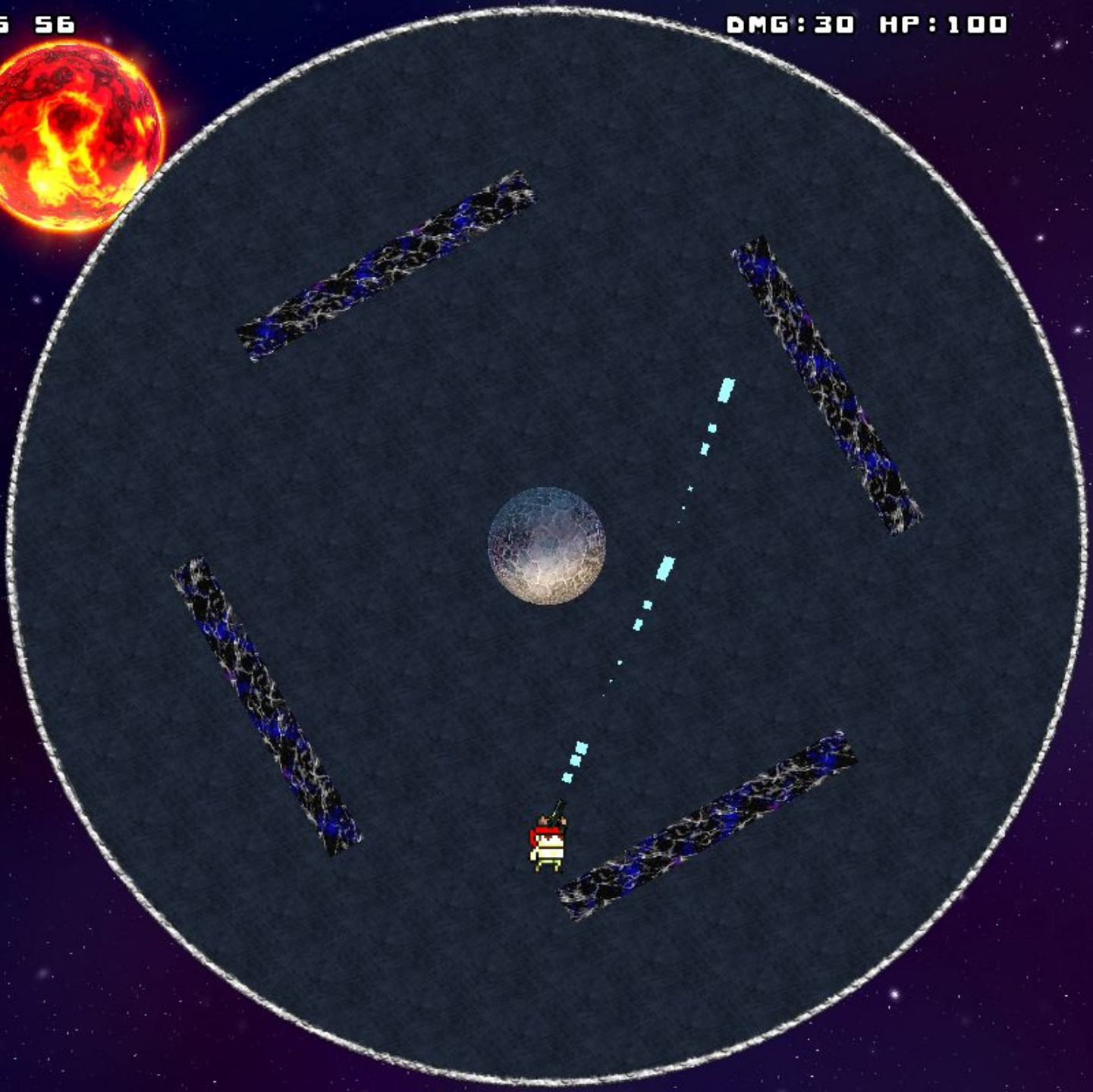
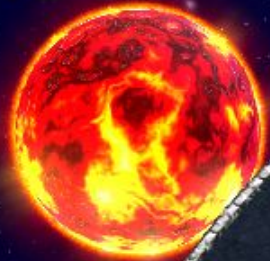
```
item, ok := responseList[0].(map[string]interface{})
```

```
...
```

## Что в итоге вышло

FPS 56

DMG: 30 HP: 100



# Sphere

Valera

Generate new



Join game

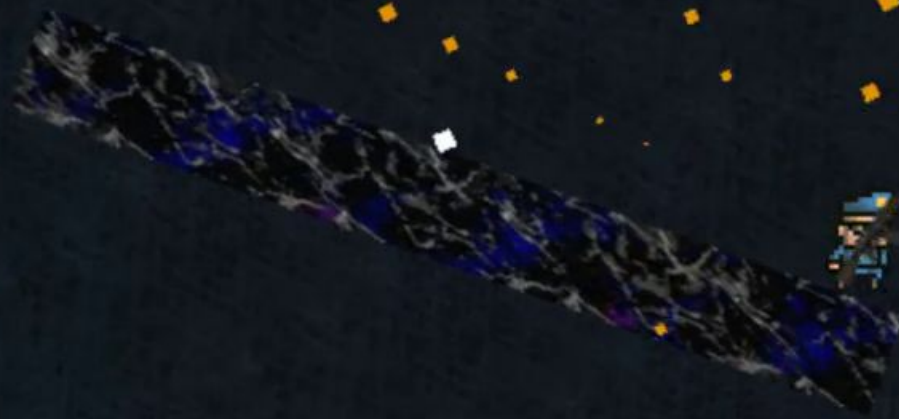
FPS 60





FPS 25

DMG:2 HP:100



Top players  
1. --Xom9k-- (YOU): 0

## Видео

[https://ater.me/conf/devconf2017\\_go.mp4](https://ater.me/conf/devconf2017_go.mp4)

## Выводы

GopherJS работает

Разделение кода на клиенте и сервере

Дебажить очень не удобно

JS-style callback hell изо всех дыр

Очень много interface{}

## Выводы

Код на GopherJS - это не Go

## Вопросы?



<https://vk.com/ac>

[conf@ater.me](mailto:conf@ater.me)

