

# Go делать игры!

Алексей Акулович (ВКонтакте)



**Golang**  
Conf 2019

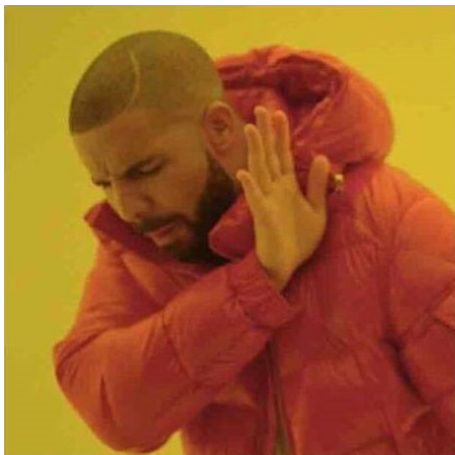
Профессиональная  
конференция  
для Go-разработчиков

# Как так вышло?





# Как так вышло?



видео                      голосовые сообщения  
                                 фотографии  
                                 CDN                      рассылка пушей  
документы                      аудио  
                                 Go RPC                      логи  
трансляции                      мониторинг

**игры**

# Подходы к GameDev'у

- Pascal (или Basic) DOS-графика

# Подходы к GameDev'у

- Pascal (или Basic) DOS-графика
- DirectDraw/DirectX, OpenGL

# Подходы к GameDev'у

- Pascal (или Basic) DOS-графика
- DirectDraw/DirectX, OpenGL
- Blitz3D, Scratch, LÖVE



# Подходы к GameDev'у

Mail.ru Game Jam 2 (2017)

Unity (C#)



# Подходы к GameDev'у

Внутренний хакатон

Go, GopherJS, WebGL



# Подходы к GameDev'у

## Внутренний хакатон

CoronaSDK (lua), WebGL



# Подходы к GameDev'у

## Ludum Dare 42

- CoronaSDK (lua), WebGL
- ASEprite, sfxr, Bosca Ceoil



# Подходы к GameDev'у

## Ludum Dare 43

CoronaSDK (lua), WebGL

ASEprite

LudumCoins: 2.99K (mining 120Mh for +1 LC) 49.7Mh (3.8Gh/s)

Farm				Power supply: 39.6KW/h from 100KW/h		Shop		x1 x10 x100	
	16.03Mh/s Outel P4	5.4KW/h	3 pcs				600Kh/s Electron 330	900W/h	BUY 1 LC
	click for Turbo mode!						3Mh/s Outel P4	1.8KW/h	BUY 10 LC
	3.38Gh/s GTX Uber	8.4KW/h	1 pcs				15Mh/s Phantom II	2.4KW/h	BUY 70 LC
	click for Turbo mode!						30Mh/s Kernel j3M	2.1KW/h	BUY 150 LC
	212Mh/s A10-5800K	11.4KW/h	1 pcs				50Mh/s Kernel 2	3.6KW/h	BUY 300 LC
	click for Turbo mode!						80Mh/s Adm Athlete	7.8KW/h	BUY 700 LC
	186Mh/s Phantom II	14.4KW/h	6 pcs				100Mh/s Kernel j7	6KW/h	BUY 1K LC
	click for Turbo mode!						150Mh/s A10-5800K	11.4KW/h	BUY 3K LC

Lua, Lua, Lua...

Где Go?

# Выбор в пользу Go

- Типизация, отладка, профилирование
- Поддержка в IDE
- Мало Go на работе 😊

А если Go...

Есть чё?





А если Go...

Какие есть игровые движки на Go?



# Какие есть игровые движки на Go

**Ebiten** 

 **Pixel**



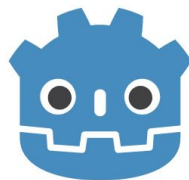
**Azul3D**

**ENGi**

**Oak**



**g3n**



**GODOT**

**raylib**

# Как выбрать ТОТ САМЫЙ?

# Как выбрать ТОТ САМЫЙ?

- ЖИВОСТЬ

# ENGi

Watch ▾

32

★ Star

384

🍴 Fork

21

Desktop + GopherJS

“SUPER ALPHA v0.6”

Не обновлялся > 3 лет 😞



Тот самый хакатон



👁 Watch ▾ 23   ★ Star 428   🍴 Fork 33

# 3D

## Desktop 🙄

### Не обновлялся > 3 лет 🙄

[github.com/azul3d/engine](https://github.com/azul3d/engine)

```
// Open the background texture.
tex, err := gfxutil.OpenTexture(abs.Path("azul3d_stencil.png"))
if err != nil {
    log.Fatal(err)
}

// Create a card object.
aspect := float32(tex.Bounds.Dx()) / float32(tex.Bounds.Dy())
var height float32 = 1.0
cardMesh := cardMesh(aspect, height)
cardMesh.TexCoords = []gfx.TexCoordSet{
    {
        Slice: cardTexCoords(0, 0, 1, 1),
    },
}
card := gfx.NewObject()
card.Textures = []*gfx.Texture{tex}
card.Meshes = []*gfx.Mesh{cardMesh}
card.State = gfx.NewState()
card.State.StencilTest = true
card.State.StencilFront = gfx.StencilState{
    ReadMask: 0xFF,
    Reference: 1,
    Fail:      gfx.SZero,
    DepthFail: gfx.SZero,
    DepthPass: gfx.SKeep,
    Cmp:       gfx.Equal,
}
```

# Какие все еще есть игровые движки

**Ebiten** 

 **Pixel**



Korok



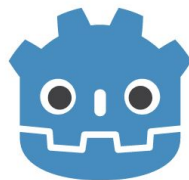
Azul3D

ENGI

**Oak**



g3n



**GODOT**

raylib

# Как выбрать ТОТ САМЫЙ?

- Живость
- Поддержка платформ



# Платформы

- Desktop

Windows, Linux, MacOS

- Mobile

Android, iOS

- Web

HTML5

# Как выбрать ТОТ САМЫЙ?

- Живость
- Поддержка платформ
  - Web HTML5
  - Mobile
  - Desktop

# Как выбрать ТОТ САМЫЙ?

- Живость
- Поддержка платформ
  - Web HTML5
  - Mobile
  - Desktop
- Предоставляемое API

Посмотрим на оставшиеся

Ebiten 

 Pixel



Korok



Azul3D

ENGi

Oak



g3n



GODOT

raylib

# Oak

👁 Watch ▾

39

★ Star

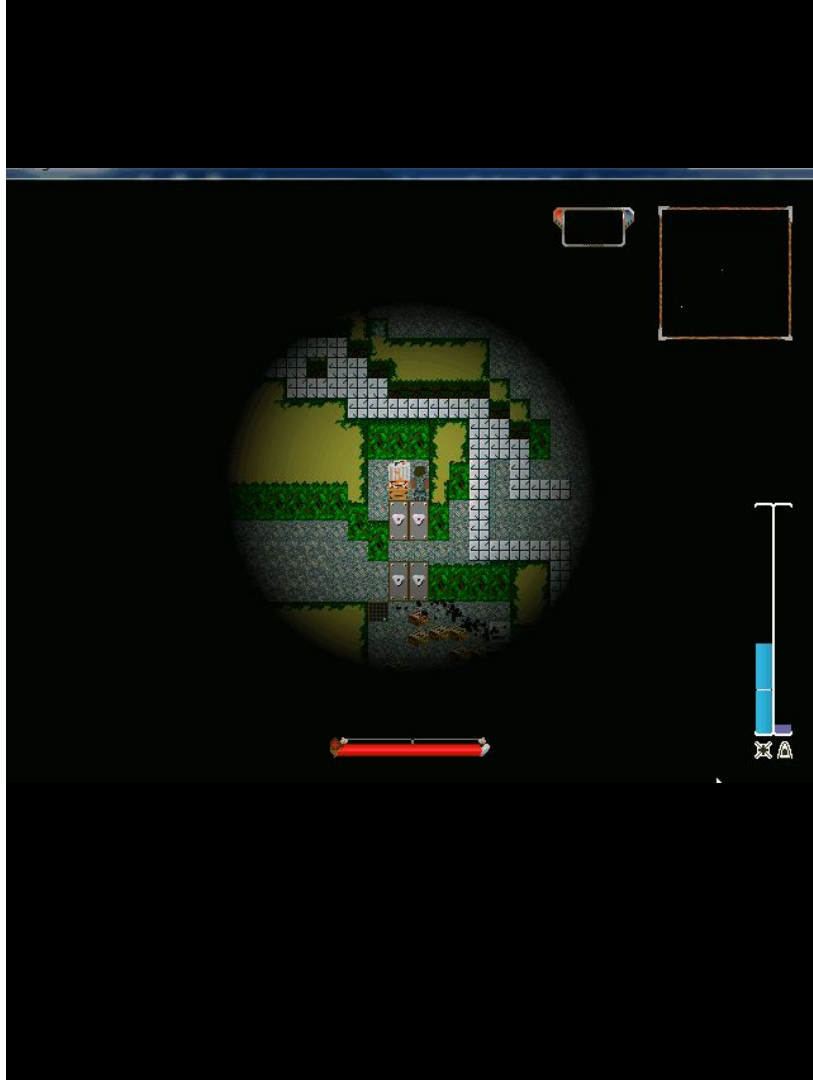
655

🔗 Fork

35

Software renderer 🙄

Desktop Windows 🙄





Watch ▾

92

★ Star

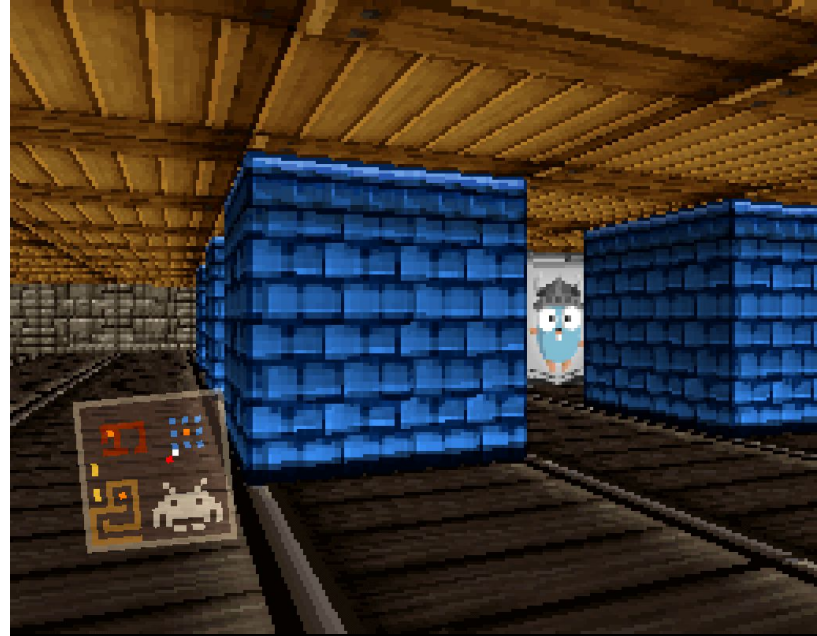
2,502

🍴 Fork

145

2D

Desktop 🙄



[pixel-examples raycaster](#)

# Korok



Watch ▾

35

★ Star

349

Fork

37

2D

Desktop, mobile, web

Делает (делал?) один китаец

Не смог запустить даже desktop 😞



# G3N



Watch ▾

59

★ Star

784

🔗 Fork

74

## 3D

Desktop sgo – сломана

wasm syscall/js – как?

Нет примеров

Почти нет документации



[github.com/danaugrs/gokoban](https://github.com/danaugrs/gokoban)





Watch ▾

1,275

★ Star

24,410

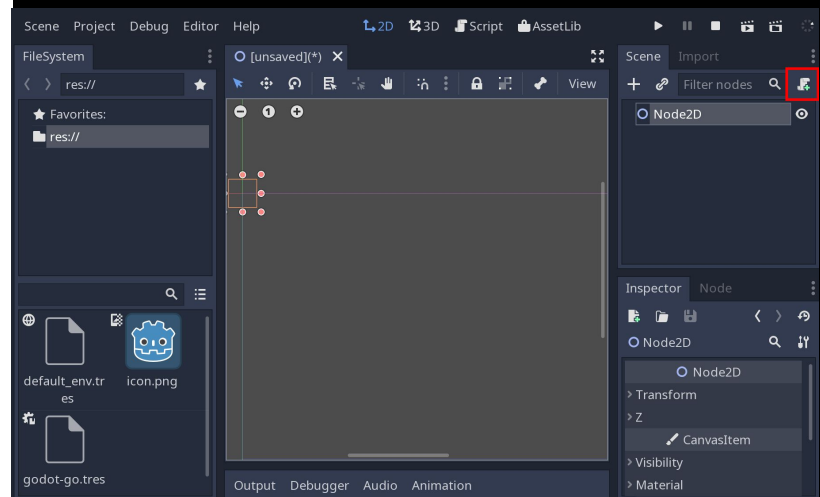
Fork

4,748

3D + 2D

Desktop, mobile, web

C++ / C# / GDScript 🙄





# GODOT-GO

Watch ▾

18

★ Star

166

🍴 Fork

14

## GDNative API

[godotengine.org/article/look-gdnative-architecture](https://godotengine.org/article/look-gdnative-architecture)

## Плагин

.so, .dylib, .dll

Нет mobile, web 😞

```
// X_Process will be called every frame.
func (p *Player) X_Process(delta gd.Real) {
    velocity := gd.NewVector2(0, 0)

    if godot.Input.IsActionPressed("ui_right") {
        velocity.SetX(velocity.GetX() + 1)
    }

    if velocity.Length() > 0 {
        normal := velocity.Normalized()
        velocity = normal.OperatorMultiplyScale(1)
    } else {
        p.animatedSprite.Stop()
    }

    // Set the position based on velocity
    position := p.GetPosition()
    newPosition := position.OperatorAdd(velocity)

    // Clamp our player's position to the size of
    newPosition.SetX(godot.Clamp(newPosition.GetX(), 0, 1))
    newPosition.SetY(godot.Clamp(newPosition.GetY(), 0, 1))
    p.SetPosition(newPosition)
}
```

[godot-go examples DodgeTheCreeps](#)



Watch ▾

151

★ Star

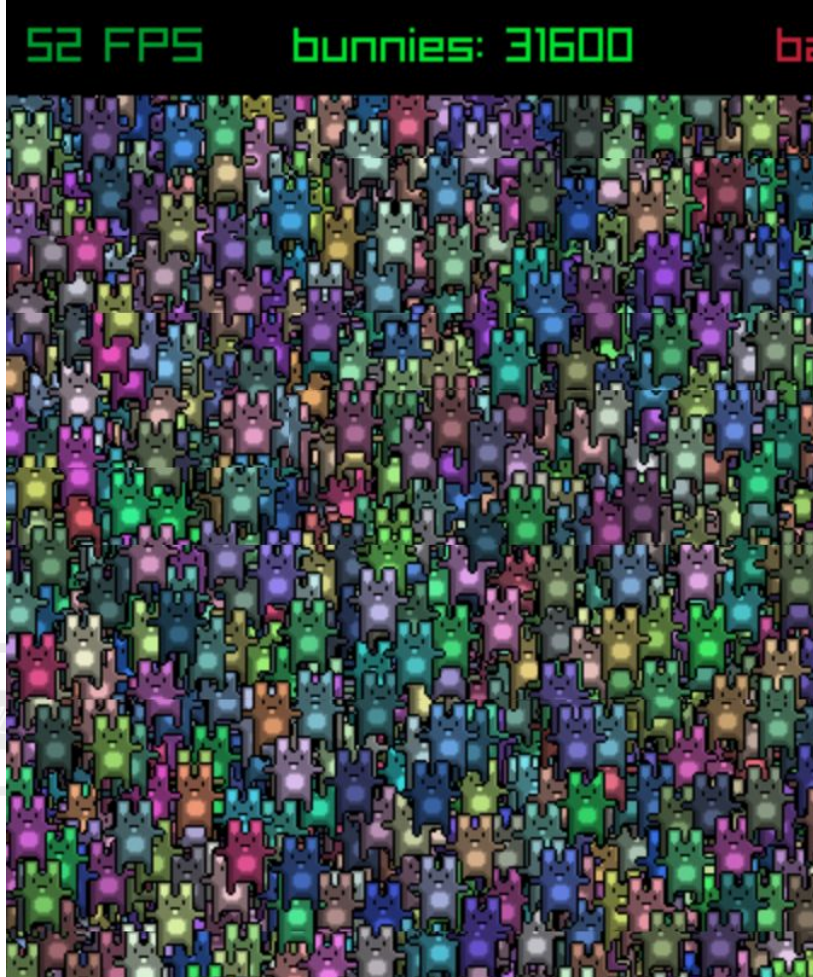
2,365

🍴 Fork

278

3D + 2D

C library





👁 Watch 21   ★ Star 391   🍴 Fork 36

cgo binding  
Desktop + Android 😞



# Что остается?

**Ebiten** 

 Pixel



Korok



Azul3D

ENGi

Oak



g3n



GODOT

raylib

# Ebiten

Watch ▾

78

★ Unstar

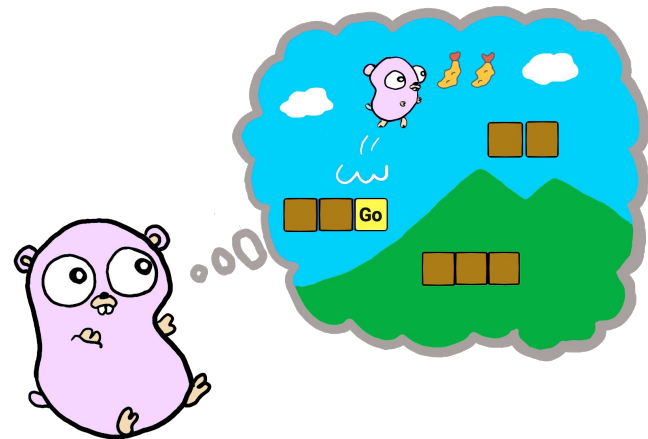
1,934

🔗 Fork

128

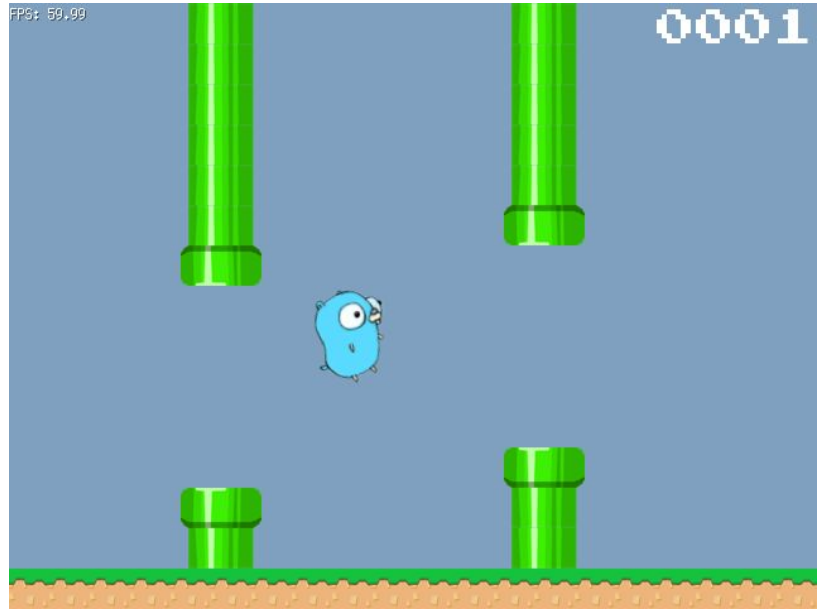
2D

Desktop, mobile, web



FPS: 59.99

0001





# Документация

## Getting Started

[Install](#)

[Hello, World!](#)

## Graphics

[Fill](#)

[Render an Image](#)

[Geometry Matrix](#)

Filter (TBD)

Offscreen (TBD)

Color Matrix (TBD)

Tiling (TBD)

## Input

Keyboard (TBD)

Mouse (TBD)

Gamepad (TBD)

## Documents

[Cheat Sheet](#)

[FAQ](#)

[Performance Tips](#)

[Package](#)



## Примеры

### Airship



Airship effect like Final Fantasy 6 with pseudo raster scrolling.

### Drag & Drop



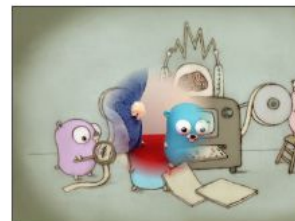
Drag & drop by handling input.

### Hi-DPI



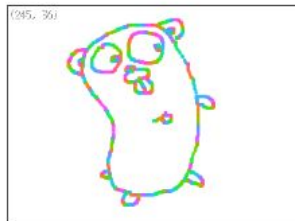
Hi-DPI rendering by specifying a special scale.

### Masking



Masking by `DrawImage` and a composition mode.

### Paint



Drawing tool using an offscreen image as a canvas.

### Perspective



Pseudo raster scrolling by rendering thin parts of an image.

### Ray Casting



Rendering ray casting regions.



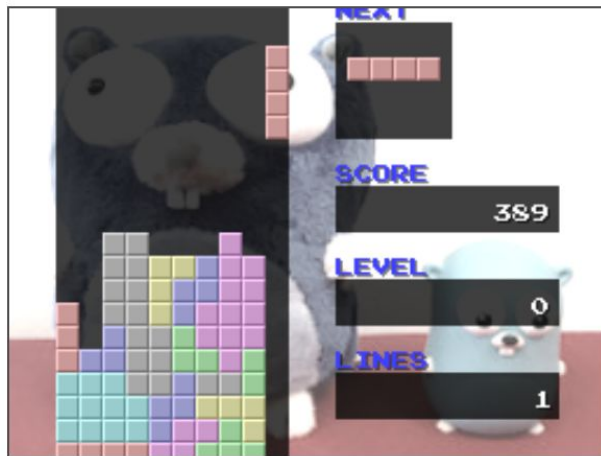


## Примеры

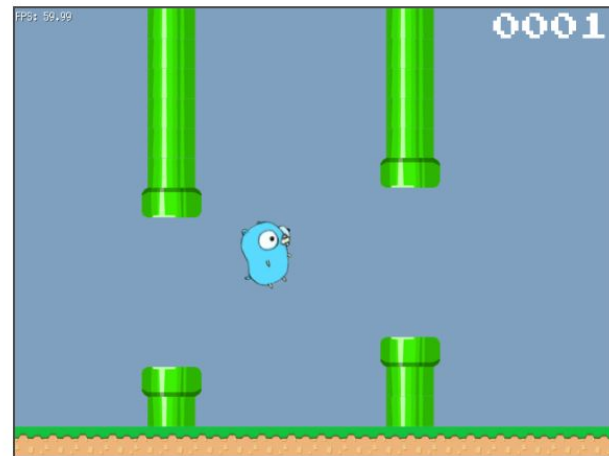
2048

		4			
4					
8	2				
32	8	2	2		

Blocks



Flappy





## Код – Go Way

```
ebiten.NewImageFromImage(source image.Image, ...)
```



## Код – Go Way

```
// +build android ios
```

# Ebiten

## Код – Go Way

```
type Graphics interface {  
    SetThread(thread *thread.Thread)  
    Begin()  
    End()  
    SetWindow(window uintptr)  
    SetVertices(vertices []float32, indices []uint16)  
    Flush()  
    NewImage(width, height int) (Image, error)  
    NewScreenFramebufferImage(width, height int) (Image, error)
```



Медленная загрузка картинок

Немного лучше с go-bindata, statik, ...

Ограниченный API

Но есть всякие `DrawImageOptions.ColorM` и т.п.

Очень жирный и медленный для веба GJS/WASM  
9+МБ на пустышке  
GJS еще и глючный





“Но ENGO-то мне точно подойдет!”



# Engo



👁 Watch ▾

46

★ Star

1,102

🔗 Fork

92

2D

Desktop, mobile, web

+ *VulkanAPI*



# Engo

## Документация

# Tutorials

This is our full blown tutorial series.

[Foreword](#)

[Hello World](#)

[Our First System](#)

[Camera Movement](#)

[HUD](#)

[Tilemaps](#)

[Spritesheets and Automatic City Building](#)

[HUD Text](#)

[Road Building](#)

# Conceptual Explanations

These are links to all of our explanations on how certain concepts in engo work.

[Animation](#)

[Building for Mobile](#)

# Road Building

In this tutorial

## Recap

Remember what we did in [the last tutorial](#)?

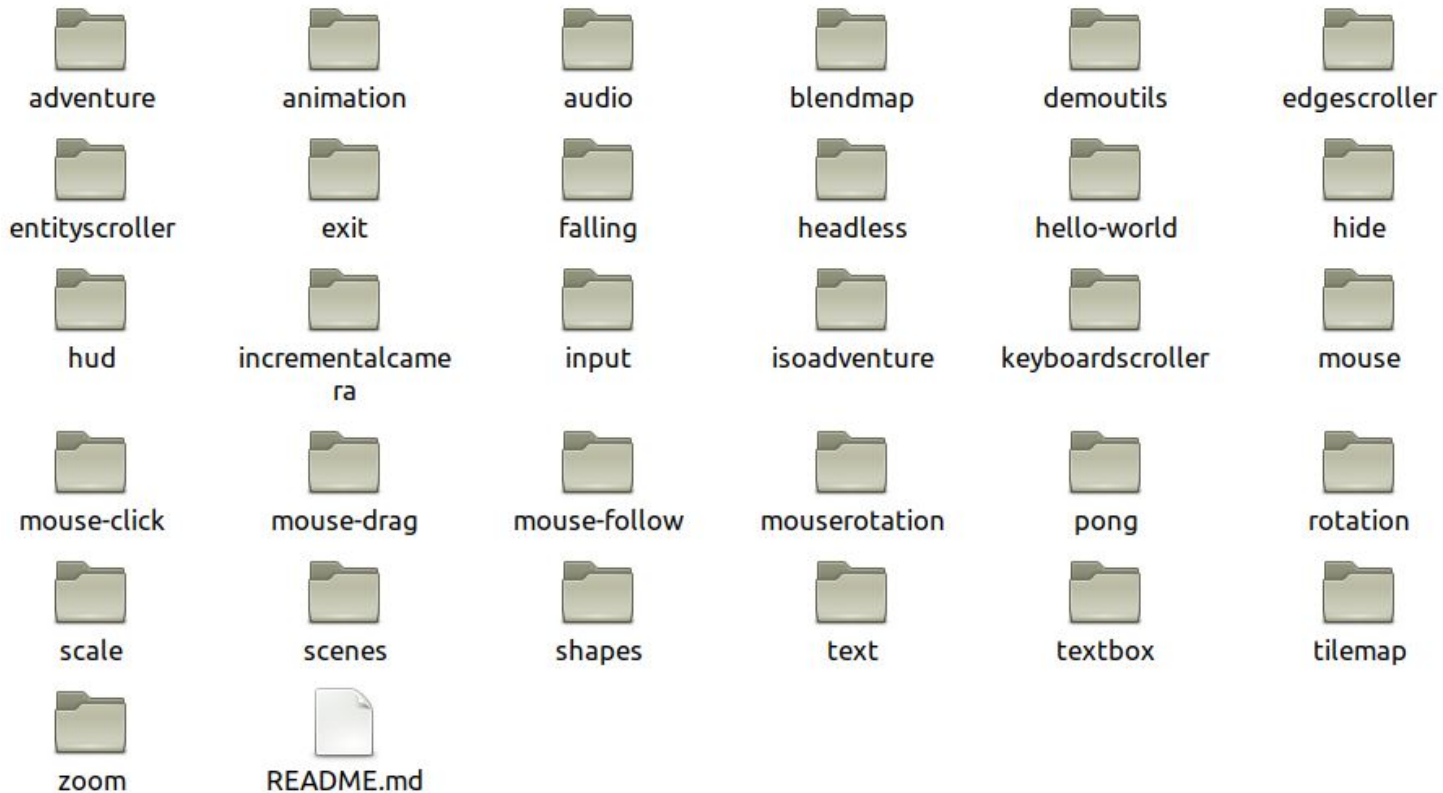
We used `engo.Mailbox` to send messages between systems so we could update our HUD text to give the player information during gameplay.

## Final Code

The final code for **tutorial 8** is available [here at GitHub](#).

# Engo

## Примеры



# Engo

A cross-platform game engine written in Go following an interpretation of the **Entity Component System paradigm**.

...

**v1.0** is now available!

...

# Engo



Engo

Как включить рендер?



# Engo

// Setup is called before the main loop starts. It allows you to add entities and systems to your Scene.

```
func (*myScene) Setup(u engo.Updater) {  
    world, _ := u.(*ecs.World)  
    world.AddSystem(&common.RenderSystem{})  
}
```

Engo

# Как добавить спрайт на сцену?

# Engo

```
type MySuperPuperGopher struct {  
    ecs.BasicEntity  
    common.RenderComponent  
    common.SpaceComponent  
}
```

# Engo

```
type MySuperPuperGopher struct {  
    ecs.BasicEntity  
    common.RenderComponent  
    common.SpaceComponent  
    common.MouseComponent  
}
```

# Engo

```
type MySuperPuperGopher struct {  
    ecs.BasicEntity  
    common.RenderComponent  
    common.SpaceComponent  
    common.MouseComponent  
    common.AnimationComponent  
}
```

# Engo

```
type MySuperPuperGopher struct {  
    ecs.BasicEntity  
    common.RenderComponent  
    common.SpaceComponent  
    common.MouseComponent  
    common.AnimationComponent  
    common.AudioComponent  
}
```

# Engo

```
type MySuperPuperGopher struct {  
    ecs.BasicEntity  
    common.RenderComponent  
    common.SpaceComponent  
    common.MouseComponent  
    common.AnimationComponent  
    common.AudioComponent  
    common.CollisionComponent  
}
```

LOADING...



# Engo

Компоненты сами себя не обновят



# Engo

## Компоненты сами себя не обновят

```
func (*myScene) Setup(u engo.Updater) {  
    world, _ := u.(*ecs.World)  
    world.AddSystem(&common.RenderSystem{})  
    world.AddSystem(&common.MouseSystem{})  
    world.AddSystem(&common.AnimationSystem{})  
    // ...  
}
```

# Engo

```
game.AddSystem(&engo.RenderSystem{})
game.AddSystem(&engo.AnimationSystem{})

spriteSheet := engo.NewSpritesheetFromFile("hero.png", 150, 150)

animationAction := &engo.AnimationAction{Name: "run", Frames: []int{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10}}

hero := scene.CreateEntity(&engo.Point{0, 0}, spriteSheet, animationAction)

// Add our hero to the appropriate systems
for _, system := range w.Systems() {
    switch sys := system.(type) {
    case *engo.RenderSystem:
        sys.Add(&hero.BasicEntity, &hero.RenderComponent, &hero.SpaceComponent)
    case *engo.AnimationSystem:
        sys.Add(&hero.BasicEntity, &hero.AnimationComponent, &hero.RenderComponent)
    case *ControlSystem:
        sys.Add(&hero.BasicEntity, &hero.AnimationComponent)
    }
}
```

# Engo

```
for _, system := range w.Systems() {  
    switch sys := system.(type) {  
    case *engo.RenderSystem:  
        sys.Add(&hero.BasicEntity, &hero.RenderComponent, &hero.SpaceComponent)  
    case *engo.AnimationSystem:  
        sys.Add(&hero.BasicEntity, &hero.AnimationComponent, &hero.RenderComponent)  
    case *ControlSystem:  
        sys.Add(&hero.BasicEntity, &hero.AnimationComponent)  
    }  
}
```

```
        sys.Add(&hero.BasicEntity, &hero.RenderComponent, &hero.SpaceComponent)  
    case *engo.AnimationSystem:  
        sys.Add(&hero.BasicEntity, &hero.AnimationComponent, &hero.RenderComponent)  
    case *ControlSystem:  
        sys.Add(&hero.BasicEntity, &hero.AnimationComponent)  
    }  
}
```



Engo

Как обработать click/tap?

# Engo

```
btn.AddToWorld(u)
mouseSystem.Add(&btn.BasicEntity, &btn.MouseComponent,
               &btn.SpaceComponent, &btn.RenderComponent,
               )
scene.World().AddSystem(&systems.MousableSystem{
    Callback: func(dt float32) {
        if btn.MouseComponent.Clicked {
            scene.join()
        }
    }})
```

# Engo



# Engo

“All problems in computer science  
can be solved by another level of  
indirection”

David Wheeler

# Engo

- Странная реализация ECS



# Engo

## - Странная реализация ECS

```
// RunOptions are the options used to Run engo  
type RunOptions struct {  
    // Update is the function called each frame during the runLoop to  
update all of the systems. If left blank, it defaults to &ecs.World{}. Use  
this if you plan on utilizing engo's window / GL management but  
    don't want to use the ECS paradigm.  
    Update Updater  
}
```

# Engo

- Странная реализация ECS
- Баги, нехватка документации

# Engo

- Странная реализация ECS
- Баги, нехватка документации
- Работает



# Engo

## Нативный на компе



# Engo

Браузер на компе



# Engo

Браузер на компе

Сеть?

```
gorilla/websocket
```

```
+
```

```
js.Global().Get(`WebSocket`)
```



# Engo

Браузер на компе

Производительность?

Пару секунд мир подождет



# Engo

Браузер на телефоне  
Samsung / iPad





# Engo

Браузер на телефоне

Samsung / iPad

GJS / WASM файл: **7+МБ**

Парсинг на ноуте: **2 сек**

Парсинг на телефонах:  **$+\infty$**



А если Go...

“А разве еще что-нибудь есть?”



# Handmade Web HTML5



# Handmade Web HTML5

- WebGL

[github.com/nuberu/webgl](https://github.com/nuberu/webgl)

- js WebSockets

- GopherJS / WASM



# Handmade Web HTML5

Ebiten - 9 МБ

Engo - 7 МБ

Фигак-фигак - 1.5 МБ

# Handmade Web HTML5

Ebiten - 9 МБ

Engo - 7 МБ

Фигак-фигак - 1.5 МБ

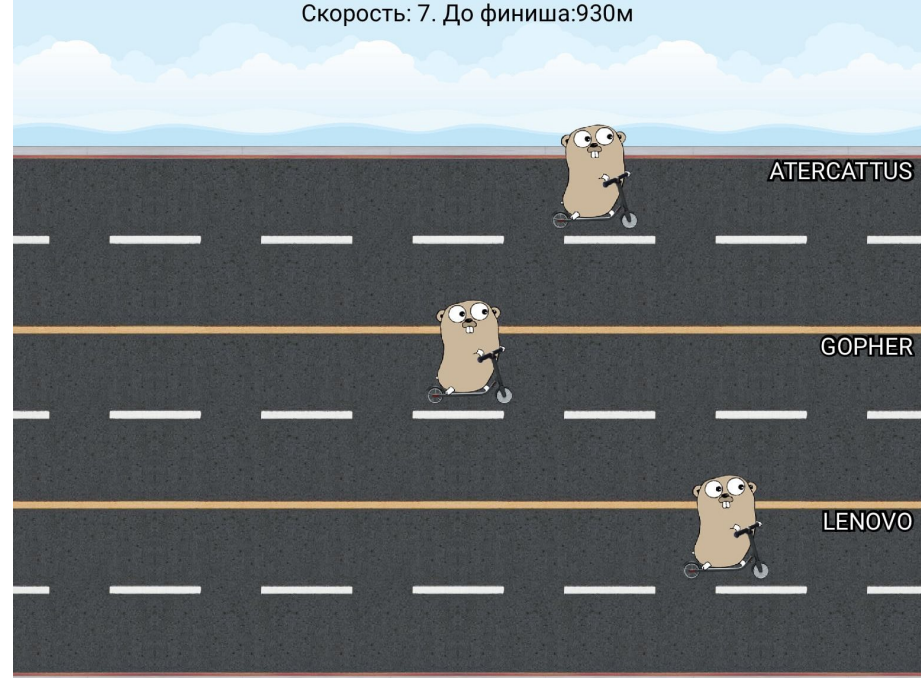
```
$packages ["reflect"] 168859
$packages ["fmt"] 93209
$packages ["encoding/json"] 140254
$packages ["math/rand"] 68914
$packages ["syscall/js"] 15510
$packages ["github.com/nuberu/webgl"] 272875
$packages [".../handmade/net"] 17650
$packages [".../handmade/render"] 76125
$packages [".../handmade/events"] 24128
$packages [".../handmade/resources"] 4661
$packages [".../handmade/objects"] 23629
$packages [".../handmade/scenes"] 64204
$packages ["main"] 8316
```



**DEMO TIME**

# Demo

## GO RACES





# Demo

Попробовать:

**[ater.me/go\\_races](https://ater.me/go_races)**

Исходники:

[github.com/atercattus/golangconf19\\_examples](https://github.com/atercattus/golangconf19_examples)



# Что выбрать?

- Desktop

Godot-go выглядит круто / Raylib для души

- Mobile

Android – raylib; iOS – ?

- Web

Handmade или никак

[ater.me/conf/goc19.pptx](https://ater.me/conf/goc19.pptx) Презентация

---

[ater.me/go\\_races](https://ater.me/go_races) Демка

---

[vk.com/ac](https://vk.com/ac)



[@atercattus](https://t.me/atercattus)



Вопросы

