



**Golang #1**

# **Golang in real work**

**Vu Nguyen**  
vu.nguyen@will.vn



# Golang in real work

- Go is not new!
- Myths and trusts about Go!
- What make Go unique?
- Go's ecosystem.
- Stories about Go.

A vertical bar on the left side of the slide, composed of several colored rectangular segments: purple, blue, teal, green, yellow, orange, and red.

# **Go is not new!**

Go 1.1 was released in Mar 2012



# Search



Repositories

22,463



Code



Issues

187,602



Users

11,041

## Languages

JavaScript

371,865

Ruby

245,139

Java

213,398

Python

204,142

PHP

167,570

C

97,485

C++

88,020

Objective-C

62,205

Shell

58,670

C#

57,874

We've found 22,463 repository results

[ethereum/go-ethereum](#)

Jeffrey Wilcke's Go implementation of the Ethereum y/w paper

Updated 4 minutes ago



[wcong/ants-go](#)

ants in go

Updated 5 minutes ago

[hkwi/gopenflow](#)

golang openflow implementation

Updated 10 minutes ago





# golang Meetups

Find out what's happening in golang Meetup groups around the world and start meeting up with the ones near you.



Groups  
134

Members  
26,104

Interested  
2,216

Cities  
112

Countries  
40



golang  
Search term

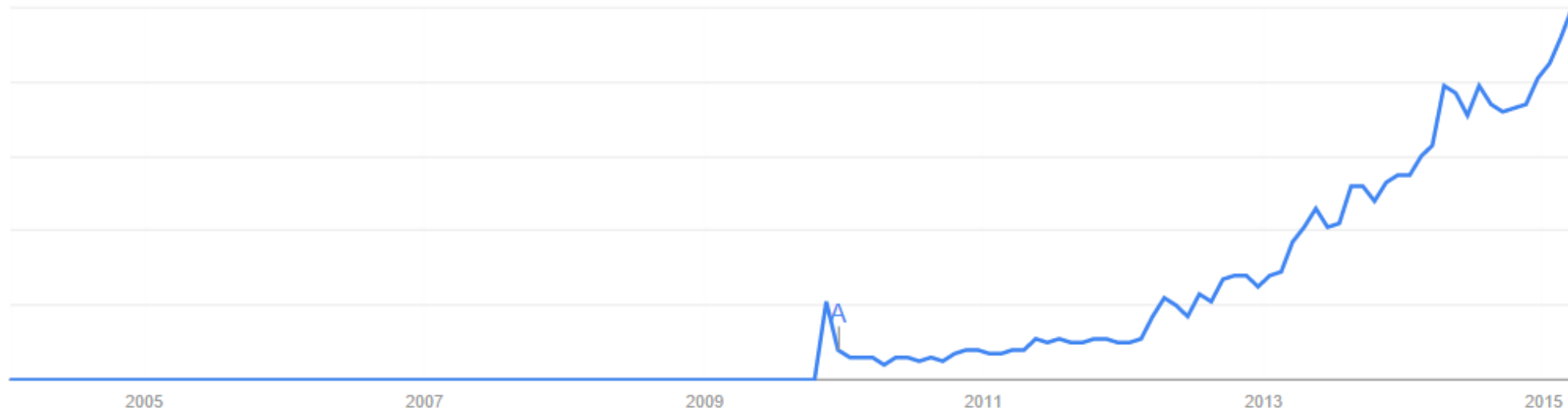
+ Add term

Interest over time



☒ News headlines

☐ Forecast





php  
Search term

+ Add term

Interest over time ?

☒ News headlines ☐ Forecast





java

Search term

+ Add term

Interest over time ?

☒ News headlines ☐ Forecast







# Go versions

- 1.0 – 3/2012
  - 1.1 – 5/2013
  - 1.2 – 12/2013
  - 1.3 – 6/2014
  - 1.4 – 12/2014
  - 1.5 – 6/2015
- First release
  - Performance improve (30-40%)  
Race detector
  - Better scheduler  
Test coverage
  - Speed up GC  
New platform: NaCl
  - Translate compiler from C to Go  
Continuous stacks  
New platform: Android ARM
  - Concurrency GC  
Completely support Android



**Myths and trusts about Go!**



# 1. Is Go “system language” ?

- Initialize, Go is described as “system language”.
- Today, they call it “general purpose language”.
- Not suitable for writing drivers, OS or low-level performance-critical app.



## 2. Is Go “compile once, run everywhere”?

- Instead, Go is “write once, run everywhere”.
- You must compile for each platform.
- It’s basically “C in 2015”.



### 3. Go has no OOP?

- No. Go is not an OOP language.
- Go follows “composition over inheritance” principle.



## 4. Writing code in Go is verbose and ugly?

- Yes. It is more verbose than Python or JavaScript.
- But it is better readable and easier collaborating.
- Your code looks same as my code!



**What make Go unique?**



**Go has nothing special !**



# Go has nothing special !

- Go is not elegant as Python.

# Go has nothing special !

- Go is not elegant as Python.
- Go is not excel at meta programming like JavaScript or Ruby.

# Go has nothing special !

- Go is not elegant as Python.
- Go is not excel at meta programming like JavaScript or Ruby.
- Go is not type safe as Rust or Haskell.

# Go has nothing special !

- Go is not elegant as Python.
- Go is not excel at meta programming like JavaScript or Ruby.
- Go is not type safe as Rust or Haskell.
- Go does not have macro or pre-processing.

# Go has nothing special !

- Go is not elegant as Python.
- Go is not excel at meta programming like JavaScript or Ruby.
- Go is not type safe as Rust or Haskell.
- Go does not have macro or pre-processing.
- Go does not have VM to execute code cross-platform.



Go only focus on one thing:

A vertical bar on the left side of the slide, composed of several colored rectangular segments: purple, blue, teal, green, yellow, orange, and red.

Go only focus on one thing:

**Get the work done !**



# Go only focus on one thing:

## **Get the work done !**

- Write code.
- Pass all test cases.
- Sleep well.
- Watch your business grow up.  
And earn money!





**Really?**

# Really?

- Single binary deployment.
- Optimize readability.
- Powerful standard libraries.
- go fmt, go doc, go build, go get, go test, oracle, ...
- Easy concurrency.



GOLANG

**bình luận**

liên quan

muốn

↑  
31  
↓

Honest question: what do you consider Go's biggest strength?

(self.golang)

↑ [-] **Queue29** 37 điểm 1 tháng trước đây

↓ Static binary deploys.

↑ [-] **thesnowmancometh** 35 điểm 1 tháng trước đây

↓ I never realized the benefits of having a small language.

↑ [-] **CorgiMann** 20 điểm 1 tháng

↓ Go is very minimalistic.

## Java:

Code -> WAR -> Classloader -> Micro Kernel (JBoss) -> JVM -> Docker -> Operating System -> Hardware

## Ruby:

Code -> Rails -> RVM -> Docker -> Operating System -> Hardware

## Python:


Code -> Framework -> PVM -> Docker -> Operating System -> Hardware

## Javascript:

Code -> Node.js -> V8 -> Docker -> Operating System -> Hardware

## Go:

Code -> Operating System -> Hardware



# Copy and run the compiled binary to deploy.

- Writing code like how we were in '90s.
- No dependency nightmare.
- No VM. No Docker.  
Just single binary.



bin

build

node\_modules

public

res

src

vendor

.bowerrc

.gitattributes

.gitignore

.gitmodules

.jshintrc

bower.json

config-local.json

gulpfile.js

LICENCE

Size: 158 MB (165,737,550 bytes)

Size on disk: 174 MB (183,480,320 bytes)

Contains: 16,240 Files, 4,360 Folders

Created: Wednesday, December 3, 20

Attributes: ☒ Read-only (Only applies to

☐ Hidden

OK

Cancel



# Gogs

## A painless self-hosted Git service written in Go

[Sign In](#)[Register](#)

### Easy to install

Simply **run the binary** for your platform. Or ship Gogs with **Docker** or **Vagrant**, or get it **packaged**.



### Cross-platform

Gogs runs anywhere **Go** can compile for: Windows, Mac OS X, Linux, ARM, etc. Choose the one you love!

# Really?

- Single binary deployment.
- Optimize readability.
- Powerful standard libraries.
- go fmt, go doc, go build, go get, go test, oracle, ...
- Easy concurrency.



# Who are using Go?

- Web server
- Command line tool



# Stories about Go!

People are switching to Go.  
And they're happy.



# Build, Ship and Run Any App, Anywhere

Docker - An open platform for distributed applications for developers and sysadmins.

## Why Go?

- Static compilation
- Neutral
- Asynchronous, low-level interfaces
- Full development environment
- Multi-arch build





# Blog

SEARCH

TUESDAY, MARCH 12, 2013

## How We Went from 30 Servers to 2: Go

When we built the first version of [IronWorker](#), about 3 years ago, it was written in Ruby and the API was built on Rails. It didn't take long for us to start getting some pretty heavy load and we quickly reached the limits of our [Ruby](#) setup. Long story short, we switched to [Go](#). For the long story, keep reading, here's how things went down.

### The Original Setup





- Memory
- Concurrency
- Reliability
- Deployment
- Talent



## Rewriting a large production system in Go

My team at Google is wrapping up an effort to rewrite a large production system (almost) entirely in **Go**. I say "almost" because one component of the system -- a library for transcoding between image formats -- works perfectly well in C++, so we decided to leave it as-is. But the rest of the system is 100% Go, not just wrappers to existing modules in C++ or another language. It's been a fun experience and I thought I'd share some lessons learned.



Plus, the Go language has a cute mascot ... awwww!





GOBOT

[Docs](#)

[Platforms](#)

[Resources](#)

[Blog](#)

[Github](#)

# Go, Robot, Go!

## Golang Powered Robotics

Next generation robotics framework with support for 15 different platforms

[Start Now](#)

[Star](#) 1,224 [Fork](#) 105 [Tweet](#) [Follow @gobotio](#)

Gobot is a framework for robotics, physical computing, and the Internet of Things, written in **the Go programming language**



# Gogs

## A painless self-hosted Git service written in Go

[Sign In](#)[Register](#)

### Easy to install

Simply **run the binary** for your platform. Or ship Gogs with **Docker** or **Vagrant**, or get it **packaged**.



### Cross-platform

Gogs runs anywhere **Go** can compile for: Windows, Mac OS X, Linux, ARM, etc. Choose the one you love!





Go

Moving from Node.js to Go at  
Bowery



**Easy**

Simply **run the binary** for your platform. Or ship Gogs with **Docker** or **Vagrant**, or get it **packaged**.



**Cross-platform**

Gogs runs anywhere **Go** can compile for: Windows, Mac OS X, Linux, ARM, etc. Choose the one you love!



# Cockroach

✓ PASSED

godoc reference

status alpha

## A Scalable, Geo-Replicated, Transactional Datastore

### Table of Contents

- [Status](#)
- [Running Cockroach](#)
- [Get in touch](#)
- [Contributing](#)
- [Design and Datastore Goal Articulation](#)
- [Architecture and Client Architecture](#)



 WIRED KLINT FINLEY

ARM,



# Cockroach

PASSED

godoc reference

status alpha

## A Scalable, Geo-Replicated, Transactional Database

### Table of Contents

 **reddit**

PROGRAMMING

comments

related

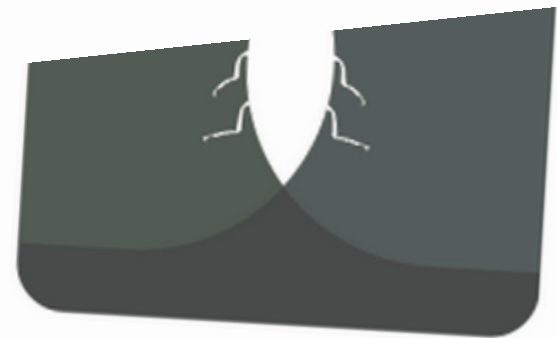
other discussions (2)

Samsung develops sorting algorithm which sorts 3.7 TB data in a minute, breaking the record of 1.5 TB (sortbenchmark.org)

submitted 3 months ago by cbruegg

379 comments share pocket

Client Architecture



 WIRED KLINT FINLEY

ARM,

# When not use Go?

- Prototyping.
- Low-level code.

A vertical bar on the left side of the slide, composed of several colored rectangular segments: purple, blue, teal, green, orange, and red.

# Meta programming in Go

# JavaScript

```
Object.assign({}, a, b)
```

```
for (let i in a) {  
    a[i] = b[i];  
}
```

# Go

```
typ := reflect.TypeOf(a)
vA := reflect.ValueOf(a)
vB := reflect.ValueOf(a)
for i:= 1; i < typ.NumFields(); i++ {
    vAi := vA.GetField(i)
    vBi := vB.GetField(i)
    vAi.Set(vBi)
}
```



# Golang #1: Golang in real work

## **Thanks for listening!**

**Vu Nguyen**  
vu.nguyen@will.vn