

# Why Go?

**Svätopluk Hanzel**

sledovani 

# About me

- FIT VUT
- backend developer at SledovaniTV
  - Go
  
- prepared this presentation in 3 hours

# GoWhat?

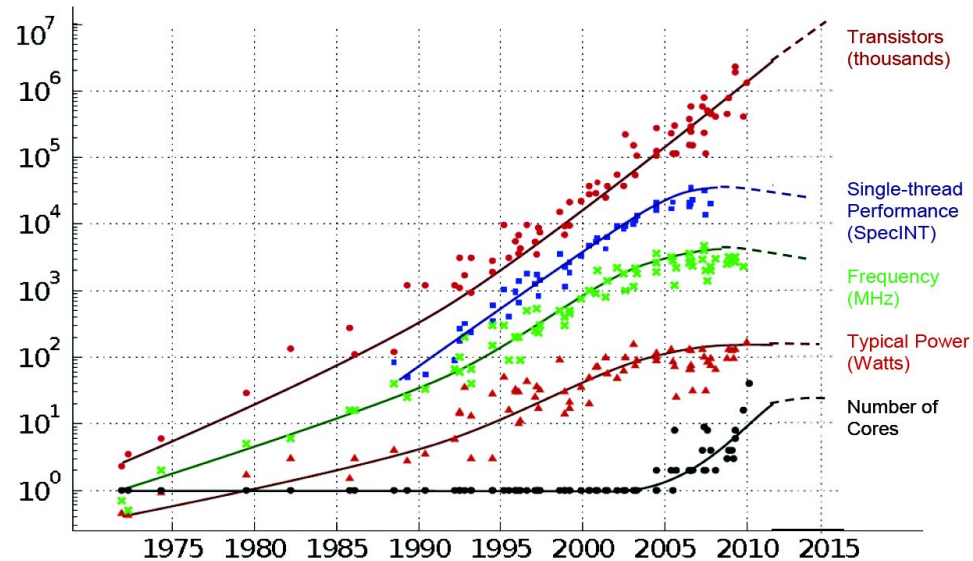
- programming language... WELL DUH!
- modern
- keeps new technologies in mind
- designed by Google
  - created by Robert Griesemer, Rob Pike and Ken Thompson

# Go history

- publicly announced in 2009
- first release in 2012
- *first thought of while waiting for a build* (not really)

# Why new language?

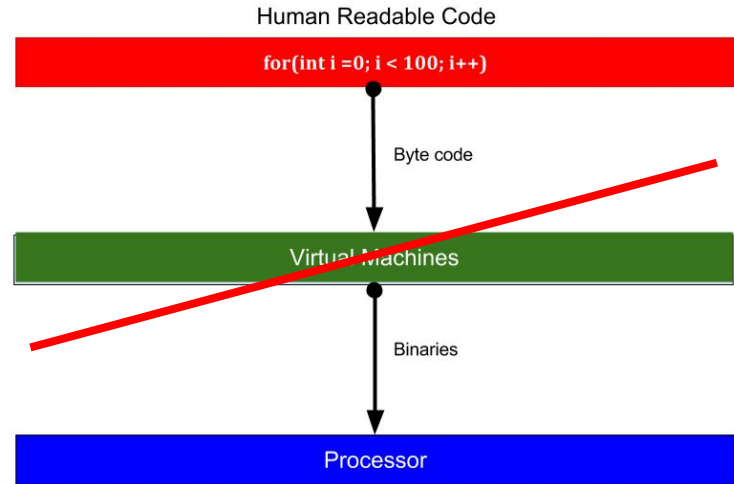
## 35 YEARS OF MICROPROCESSOR TREND DATA



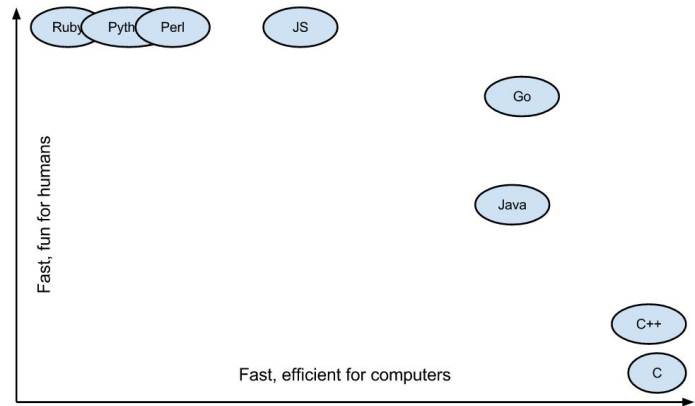
Original data collected and plotted by M. Horowitz, F. Labonte, O. Shacham, K. Olukotun, L. Hammond and C. Batten  
Dotted line extrapolations by C. Moore

PROs

# Compiles to machine code



# Performance





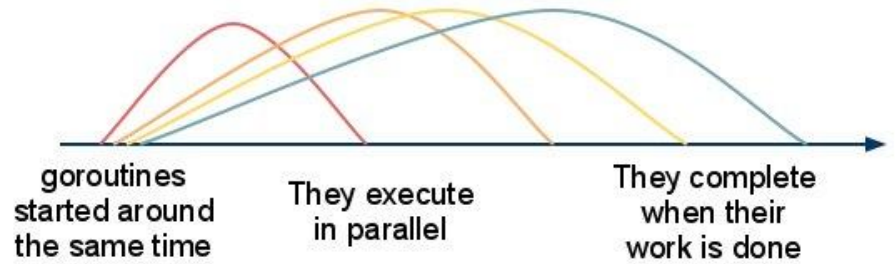
# Simplicity

- conventions
- one way to do stuff
- ease to learn new codebase
- reading code

# Ready to go async

- goroutines
- mutexes
- channels
- WaitGroups
- Race condition detection

# Concurrency



# Strong typing

# Static typing

```
var a int = 5  
var b int32 = 5
```

```
if a == b { // compile-time error: mismatched types int and int32  
    log.Println("WAAAAT?")  
}
```

# No objects

- structures
- receivers
- composition

# Interfaces

- make writing generic code easy

```
type Pet interface {
    Name() string
}

type Cat struct {
    name string
}

func (cat *Cat) Name() string { return
cat.name }

type Dog struct {
    name string
}

func (dog *Dog) Name() string { return
dog.name }

func StrokePet(pet Pet) {
    fmt.Println("Good boy, %s", pet.Name())
}
```

# Error handling

```
type error interface {
    Error() string
}

type ErrFoo struct {
    LineNo int
}

func (err *ErrFoo) Error() {
    return fmt.Sprintf("Error on %d", err.LineNo)
}

...

if err := returnErr(); err != nil {
    log.Println(err)
}
```

# Huge standard library

- http
- json, xml, csv
- sql
- templates
- ...and so much more



Build speed

# Deployment

- statically linked
- cross-platform
- cross-compilation

CONS

Bit different

No generics

# Error handling

- no exceptions

# Example

# Hello, world!

```
package main

import "fmt"

func main() {
    fmt.Println("Hello, world")
}
```

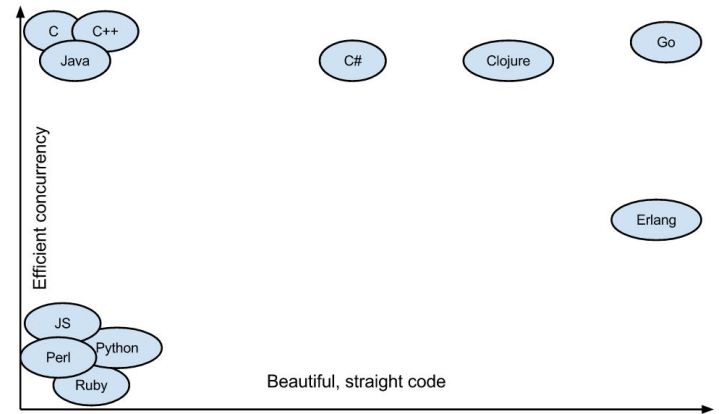
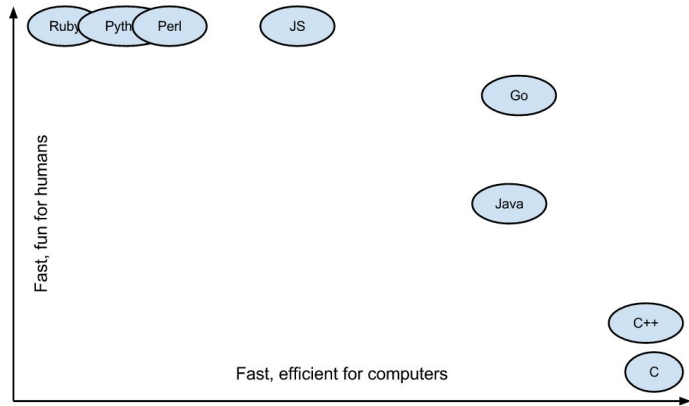


How to start?

<https://tour.golang.org>

# TL;DR

- Go is awesome



# Go v Brne

- SledovaniTV
- Meetupy - 12/2018
- Facebook (@GolangBrno)

Ďakujem za pozornosť!

Otázky?