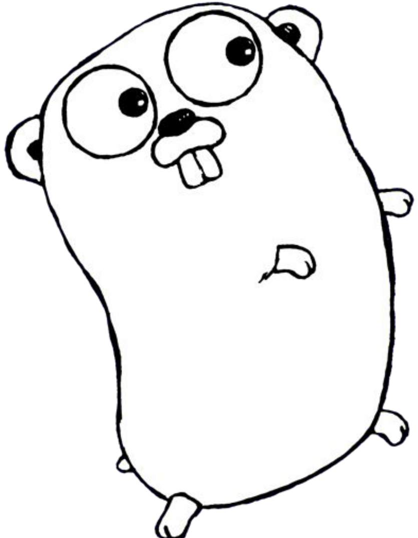
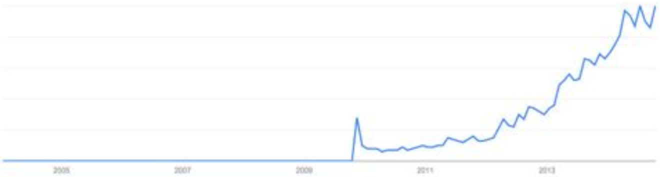
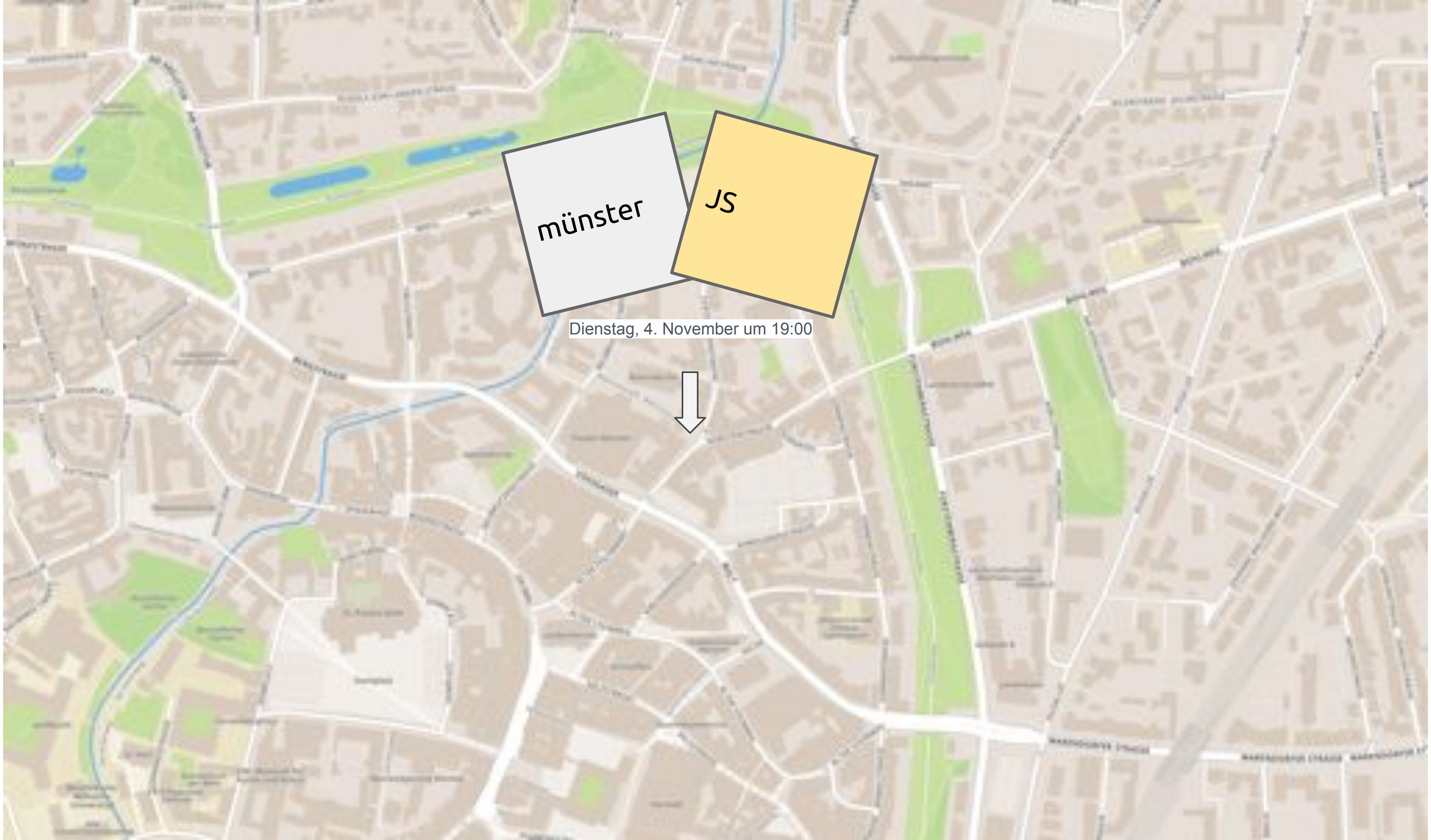
Go

@MoR







münster JS

Dienstag, 4. November um 19:00

Why did Google invent Go?

- Efficiency
- Safety
- Concurrency
- Scalability
- Fast dev. cycle

The language

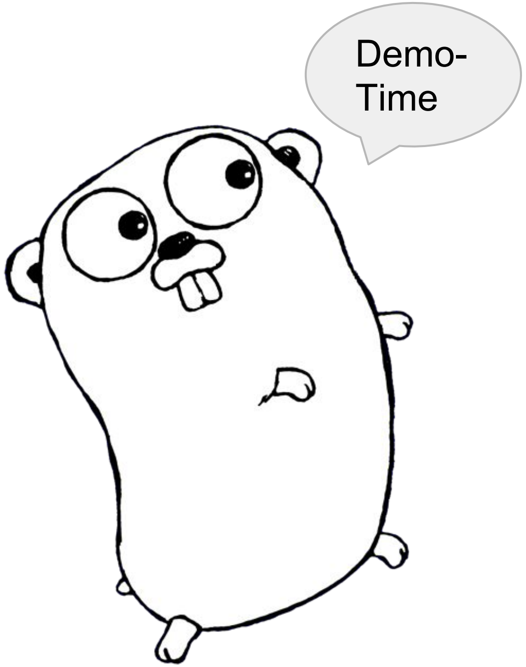
- Somehow like C
- { Braces are mandatory }
- compiled, statically-typed

```
package main

import "fmt"

func main() {
    fmt.Println("Hello World!")
}
```

“The first step to making Go scale, dependency-wise, is that ***the language defines that unused dependencies are a compile-time error*** *(not a warning,* ***an error****).”* - Rob Pike

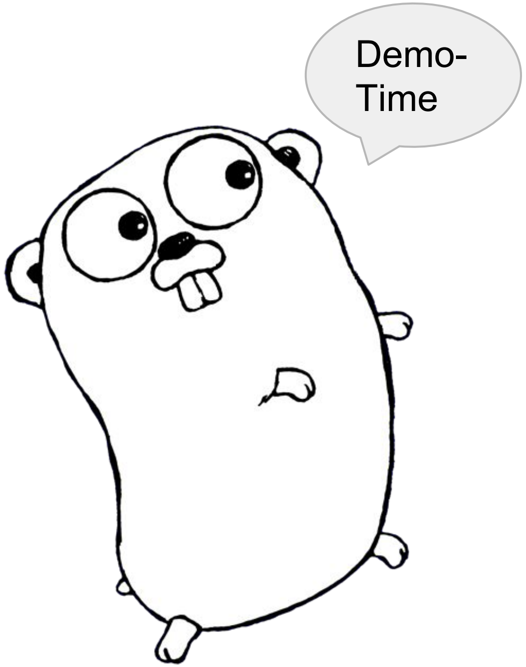


Demo-
Time

Web Frameworks

- Revel
- Martini
- Negroni
- Gocraft/web
- Gorilla
- net/http

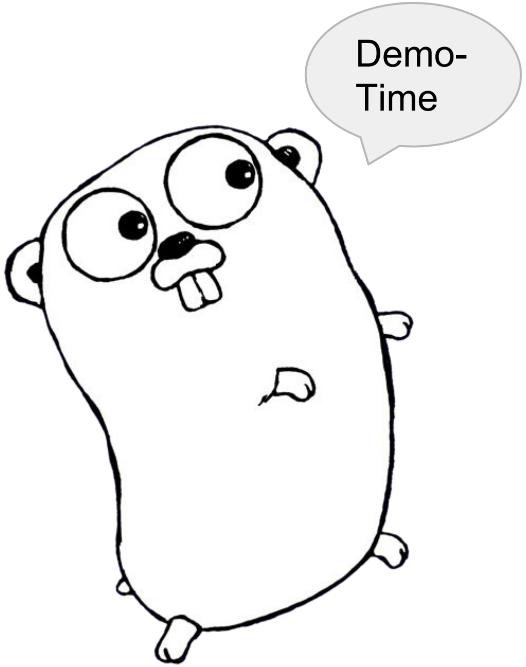
Web-Framework	Routing	Data binding	Controllers / context	Middleware	Notes
Revel (revel)	parameter, wildcards	arguments	Must have Controllers	“interceptors”	“ **Magic**
Martini (Code Gangsta)	parameter, wildcards, regex	injected	DIY	middleware all over it	inspi d l **Magic**
Negroni (Code Gangsta)	BYOR	DIY	DIY	middleware all over it	**library / not a framework**
Gocraft/web (Gocraft)	parameter, regex	wrappend, web.Request	not directly	general / contextual	**library / not a framework**
Gorilla (Gorilla Web Toolkit)	paramter (&regex)	mux.Vars (request)	DIY	DIY	**independent libraries**
net/http	BYOR	DIY	DIY	DIY	**purist**


Demo-
Time

Packages

- Everything lives in a package
- Packages can be very small, or very large
  - package `errors` → one declaration
  - package `net/http` → > 100 declarations
- Case determines visibility
  - `Foo` → exported
  - `foo` → not exported

Structs, Interfaces and Packages



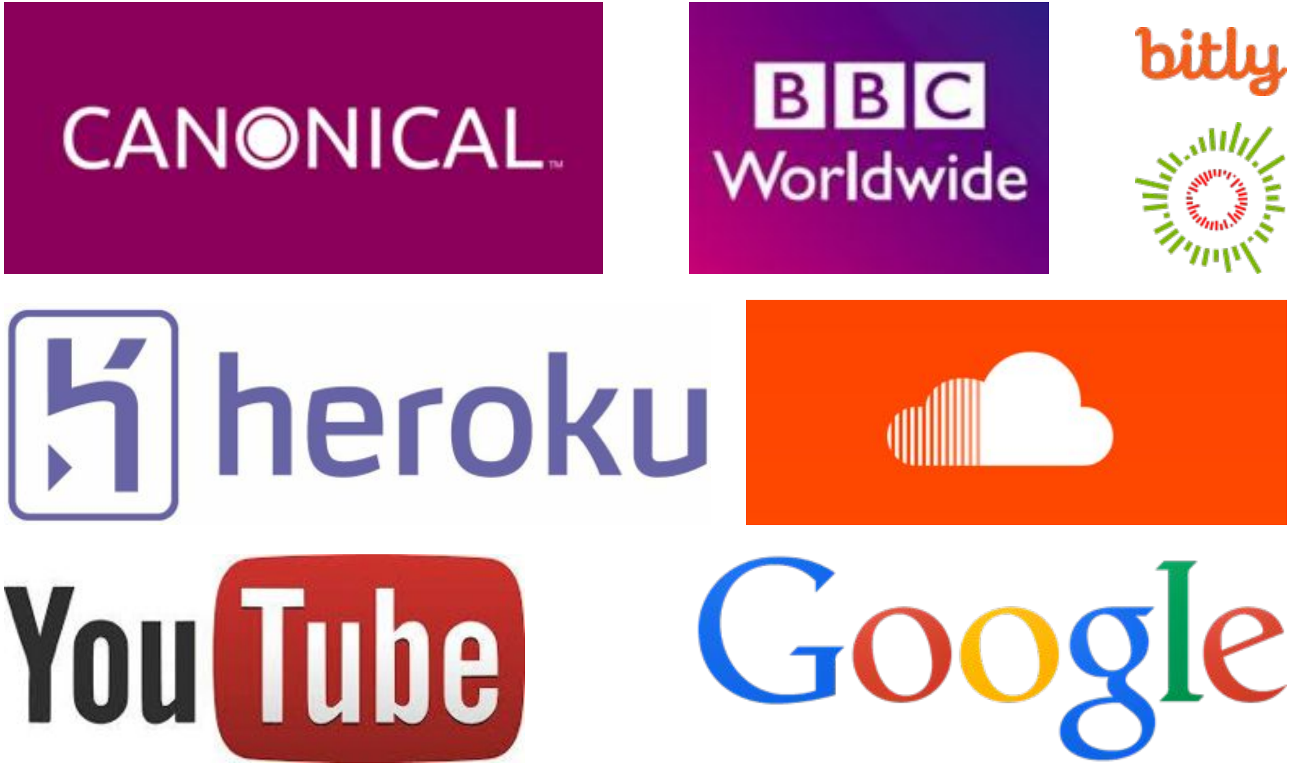
- Integrating With External Code
- Managing Dependencies

- Deploying Go
- Go's Ecosystem

Go is used for…

- Web frameworks
- Image processing
- Crypto
- Replacing shell scripts
- Controlling flying drones ;-)
- Emulators
- Disassembler, linker, compiler.
- Mobile
- Audio
- Cloud Infrastructure
- Load balancers & servers
- System administration
- ...

Who is using Go?



Links and resources

http://golang.org/

http://blog.golang.org/

http://www.golang-book.com/

http://openmymind.net/assets/go/go.pdf

http://talks.golang.org/2012/splash.article

http://talks.golang.org/2012/simple.slide#1

https://talks.golang.org/2013/bestpractices.slide#1

http://blog.iron.io/2013/03/how-we-went-from-30-servers-to-2-go.html

http://blog.iron.io/2013/08/go-after-2-years-in-production.html

Q&A

Merci