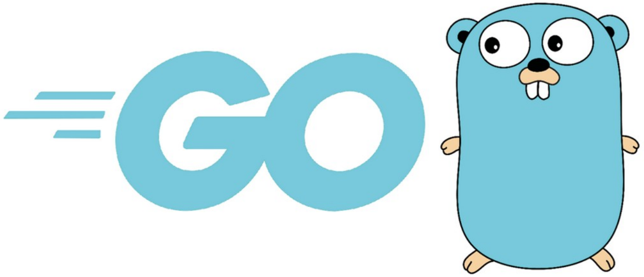

Lunch & Learn – Golang

An Intro to Go & gRPC

by Marcin Kalita

Lunch & Learn - Golang

Introduction

Go was created at Google in 2007, and since then, engineering teams across Google have adopted Go to build products and services at massive scale.

Lunch & Learn - Golang

Use case

- Cloud & Network Services

Address tradeoff between development cycle time and server performance.

Lunch & Learn - Golang

Example

Kubernetes is an open-source container-orchestration system, written in Go, for automating web app deployment.

Lunch & Learn - Golang

Use case

- Command-line Interfaces (CLIs)

Leverage fast compile times to build programs
that start quickly and run on any system

Lunch & Learn - Golang

Example

Hugo is one of the most popular open-source static site generators.

Lunch & Learn - Golang

Use case

-Web Development

Leverage Go’s out-of-the-box performance to scale with ease

Lunch & Learn - Golang

Example

Cloudflare speeds up and protects millions of websites, APIs, SaaS services, and other properties connected to the Internet.

Lunch & Learn - Golang

Use case

- Development Operations (DevOps) & Site Reliability Engineering (SRE)

Easily build small scripts with Go’s robust standard library and static typing

Scale and maintain larger applications with Go’s low memory footprint and doc generator

Lunch & Learn - Golang

Example

Docker is a platform-as-a-service that delivers software in containers. Containers bundle software, libraries, and config files, are hosted by a Docker Engine, and are run by a single operating-system kernel (utilizing less system resources than virtual machines).

Case studies

Allegro, American Express, Bitly, Dropbox,
Google, Microsoft, Netflix, Salesforce, Target,
Trivago, Twitch, Twitter, Uber and many more...

Lunch & Learn - Golang

Lunch & Learn - Golang

Language Overview

- Compiled
- Garbage Collected
- Statitcally Typed
- Strict Compiler (e.g. no unused variables)
- Code Formatter Built In (go fmt)
- Idiomatic, Opinionanted - Conventions
- Good Linters and Adoption by Text Editors

Fun fact

Go doesn't have inheritance, instead it allows composition as a way to extend the functionality of types.

The first way is embedding, which can be viewed as an automated form of composition.

```
type Foo struct {
  a string
}

type Bar struct {    // struct
  Foo                // embedded struct
  b int
}
```

The second way is interface, which provides runtime polymorphism.

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

See: https://go.dev/play/p/UweJFkL0LKb

Lunch & Learn - Golang

Fun fact

Go’s concurrency model is based on:

- Goroutines - independently running functions,
- Channels – structured data pipelines.

Example: https://go.dev/tour/concurrency/1

Lunch & Learn - Golang

gRPC is an open source, high performance Remote Procedure Call (RPC) framework, initially created by Google to connect the large number of microservices.

gRPC uses HTTP/2 for transport, by default - Protocol Buffers as the Interface Definition Language.

Main features:
- metadata,
- streaming,
- cancellation and timeouts,
- cross-platform client and server bindings generation for many languages.

RPC types

- Unary RPC
A client sends a single request and gets back a single response

- Server streaming RPC
A server returns a stream of messages in response to a client’s request

RPC life cycle

- Client streaming RPC
A client sends a stream of messages to the server

- Bidirectional streaming RPC
A client and server can read and write messages in any order

Lunch & Learn - Golang

Lunch & Learn - Golang

Workshop

https://github.com/rubyconvict/lunchnlearn