

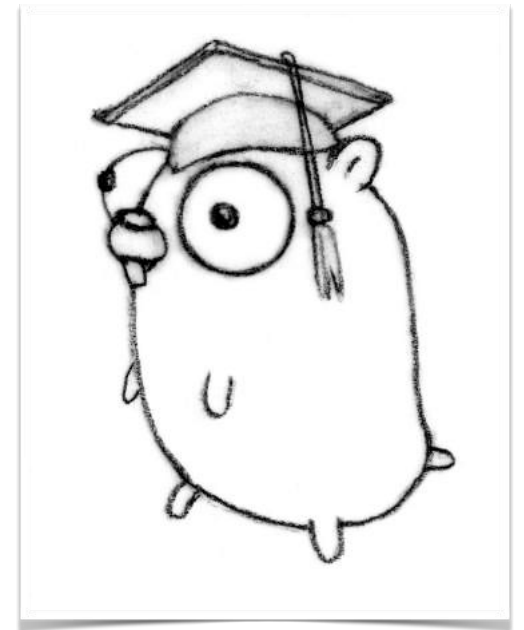
Macoun

Businesslogos plattform Schwebefähig kreiereln

Marco Feltmann

Ablauf

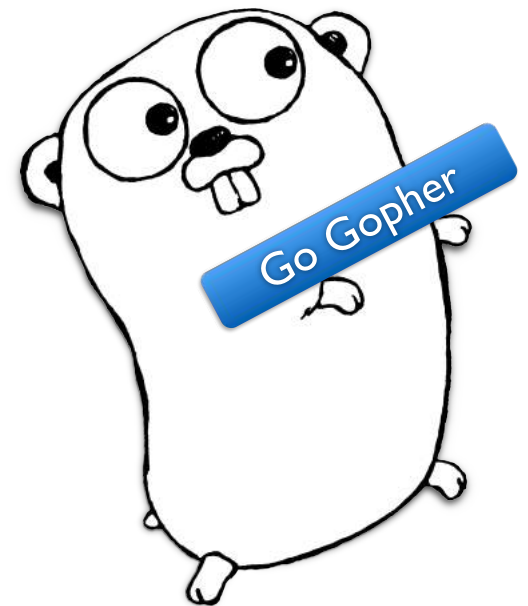
- Glossar
- Warum ausgerechnet Go?
- Entwicklung der Businesslogik
- Fremdkomponente einbinden
- Zusammenfassung



Worterkklärungen

Gopher

- Aushilfskraft [sl.]
- Taschenratte, Ziesel [zool.]
- Leidenschaftlicher Go-Programmierer [comp.]
- Golang Maskottchen
kreiert von Renee French



Businesslogik

- Gewisse Schöpfungshöhe
- Betrifft Geschäftsprozesse
- Teilschritte des Systems



Computerplattform

- Verarbeitungseinheit (CPU)
- Flüchtige Speicher (RAM)
- Persistierender Speicher (SSD)
- Eingabegeräte (Touchpad, Digital Crown, Gyroskop...)
- Ausgabegeräte (Monitor, Drucker, Lautsprecher...)

betriebssystemübergreifend

- Multi-Platform-App
 - Einheitliche Sprache für Logik, I/O und UI
 - Spezifische Anpassungen möglich
 - Systemkomponenten „unter der Haube“ genutzt
- Universal Binary
 - Logik in portabler Sprache
 - UI und I/O in Systemsprache
 - Einbinden der Logik als Binärabhängigkeit

betriebssystemübergreifend



Warum ausgerechnet Go?

Warum nicht C++?

*Note that these costs are particularly high in C++
(as opposed to other possible non-native languages like Python or C#)
because it lacks of a single, full-featured standard library.*

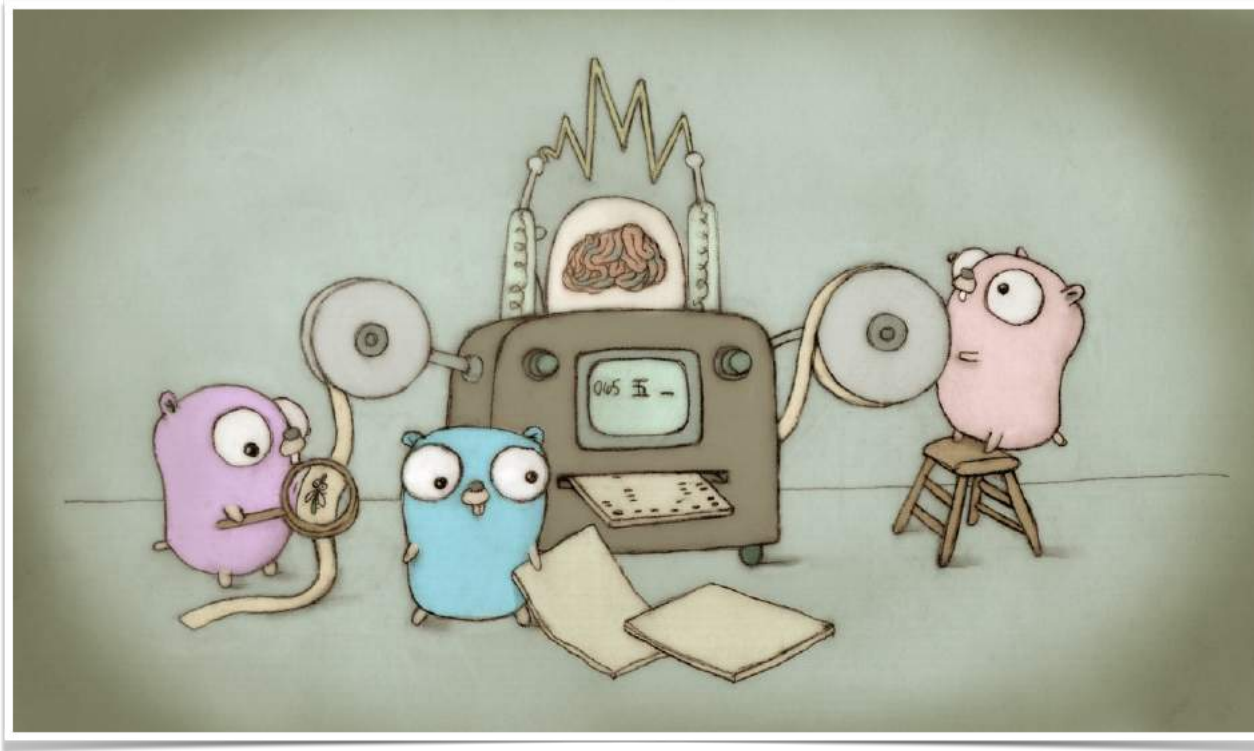
<https://blogs.dropbox.com/tech/2019/08/the-not-so-hidden-cost-of-sharing-code-between-ios-and-android/>

Warum denn nun Go?

The Go mobile repository holds packages and build tools for using Go on mobile platforms.

<https://github.com/golang/mobile/blob/master/README.md>

Kurzvorstellung gomobile



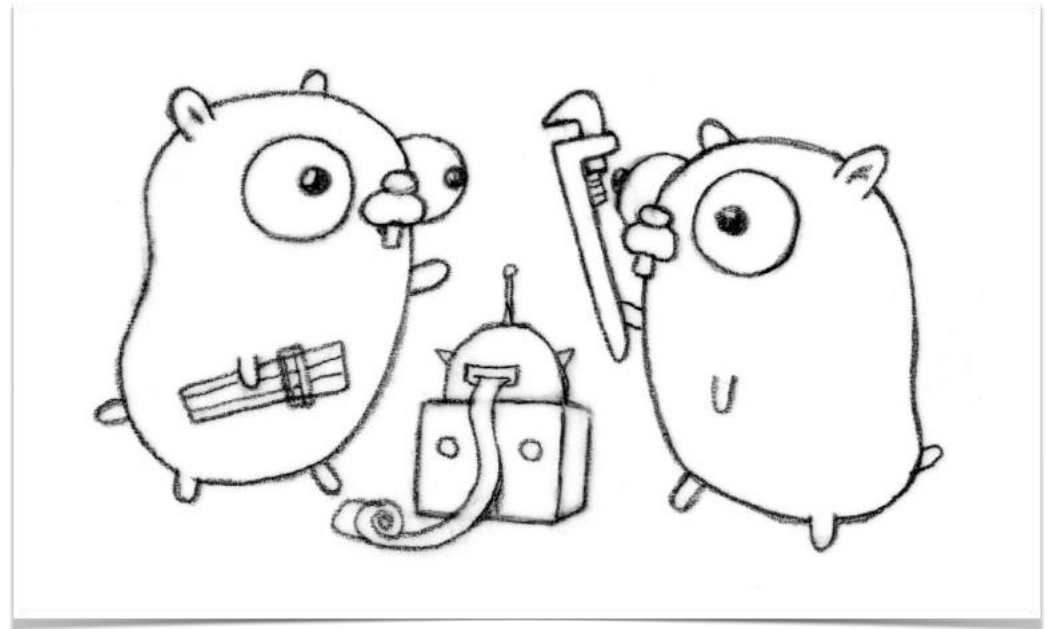
Go auf mobilen Endgeräten

- Komplette in Go
(UI: DIY OpenGL)
- Generierung eines SDK aus Go
(via cgo)

Exportierbare SDK Datentypen

	Boolean	Signed Integer	Float	String	Byteslice	Error
Member	✓	✓	✓	✓	✓	✗
Parameter	✓	✓	✓	✓	✓	✗
Rückgabe	✓	✓	✓	✓	✓	✓

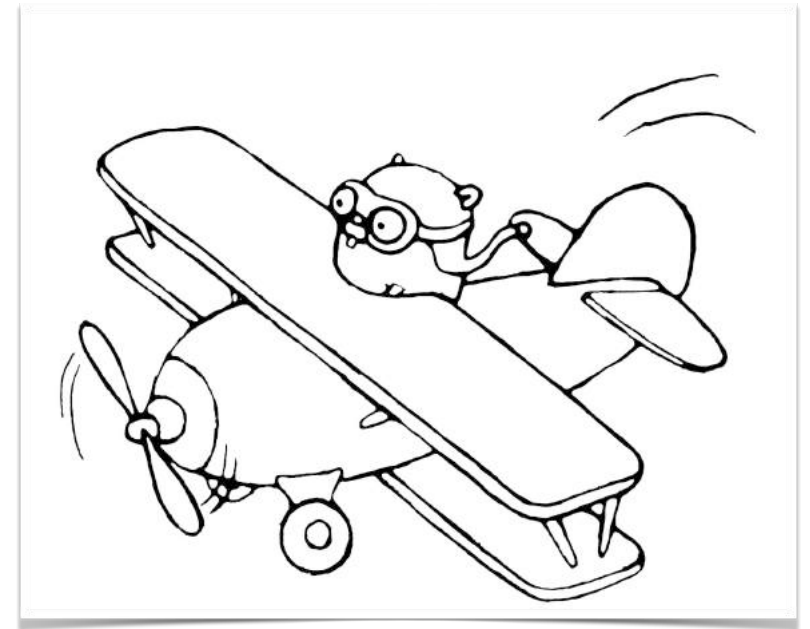
Demo gobind



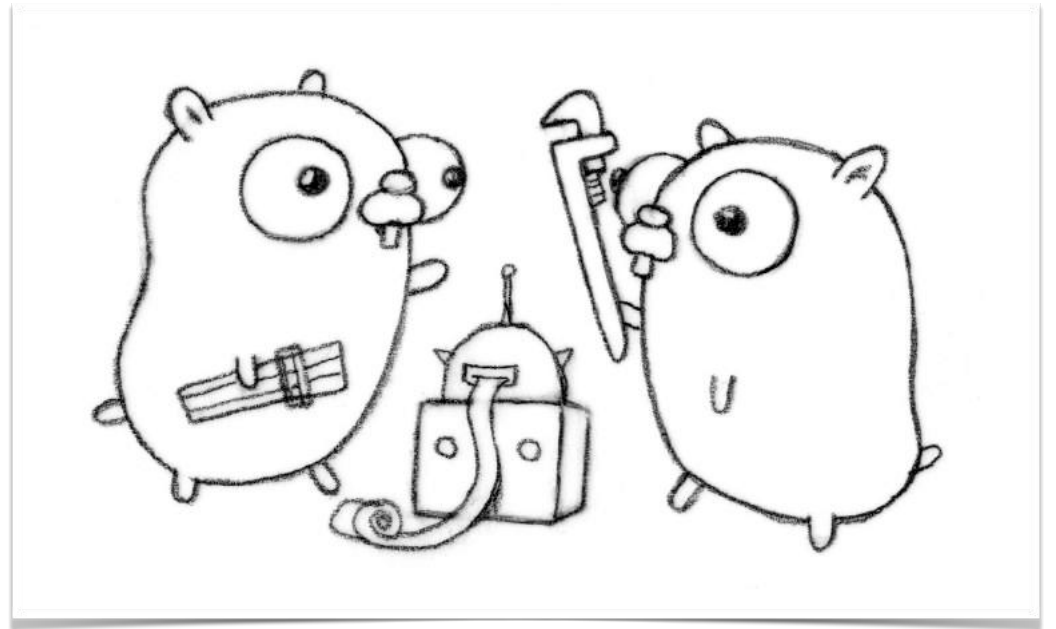
Entwicklung der Business Logik

Aus Logik wird ein Zulieferprodukt

- Komponententrennung
- Eigener Auslieferungsprozess
- Support- und Wartungsaufwände

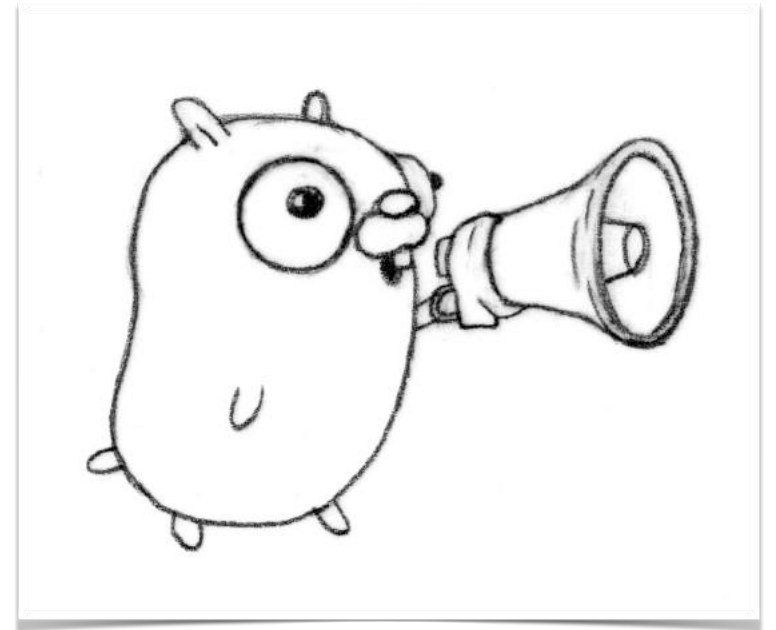


Demo spacedata



Zwischenfazit

- Keine Pointer
- Byteslice by reference und veränderbar
- Boundaries vs. encoding/json
- Reference Counting vs. Garbage Collection



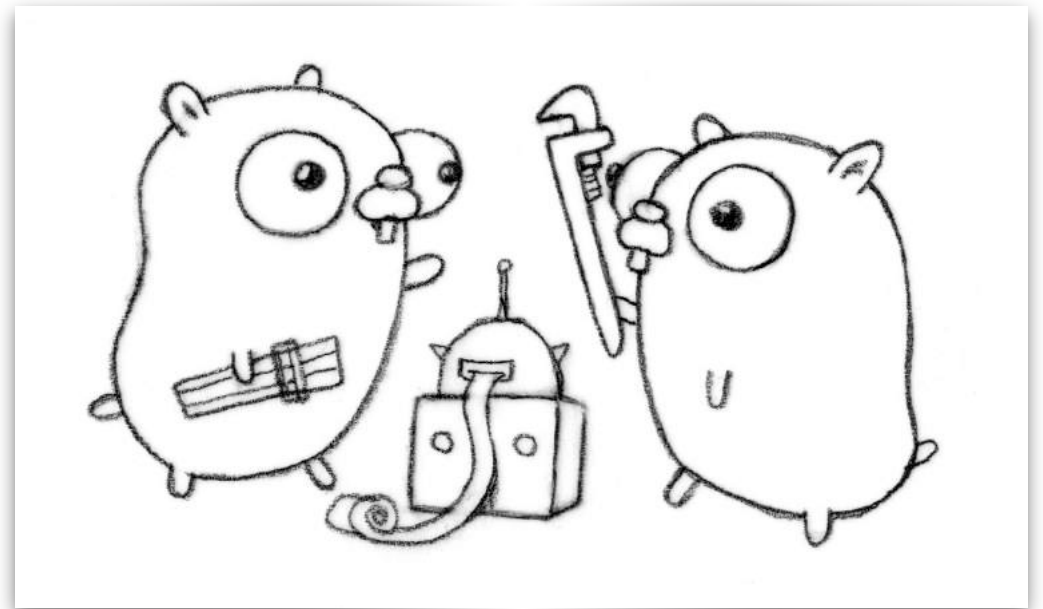
Fremdkomponente einbinden

Abhängigkeitsverwaltung

- Binärdaten
- Versioniert
- Android Studio: Gradle mit Maven Plugin
- Xcode: ~~X~~

Demo

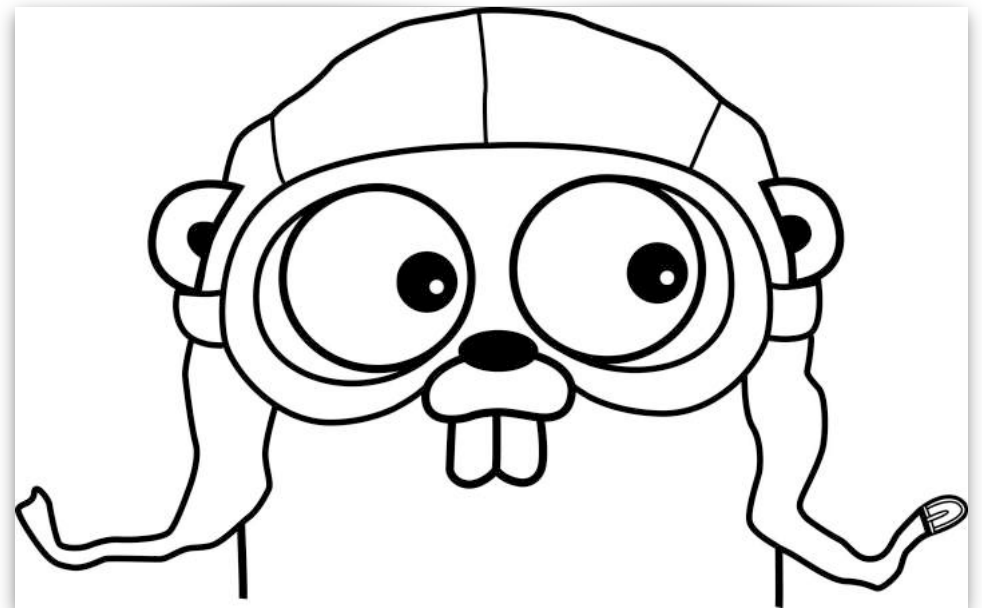
ISSLocator



Zusammenfassung

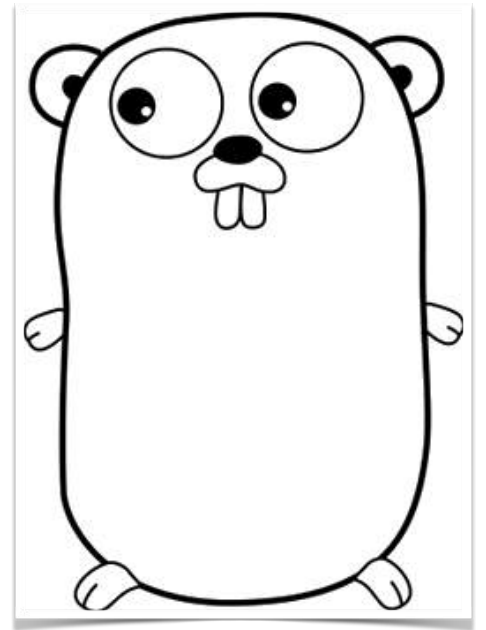
Gewinn

- Verbesserte Schichtentrennung
- Optimierte Testbarkeit
- Fokus und Fähigkeiten des Plattformteams erhalten

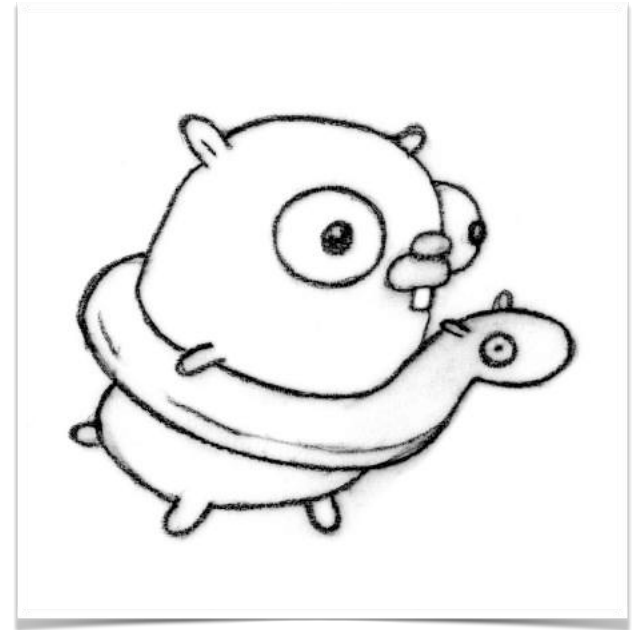


Aufwände

- Releaseprozess für Bibliotheken
- Frameworks aktuell ohne Bitcode

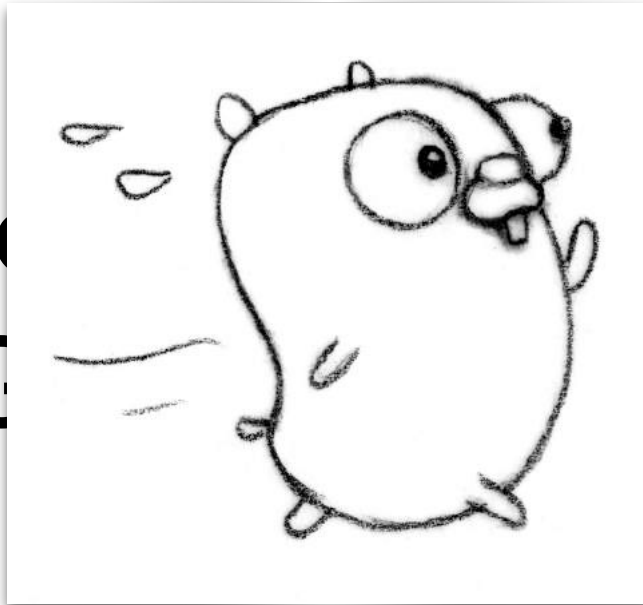


Fragen?



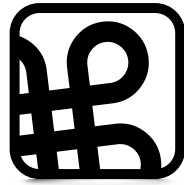
Vic

C



nd

it



Macoun