

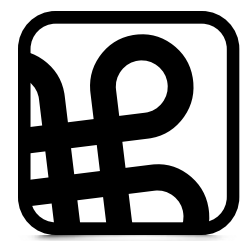
Macoun



Codable?

Decoding the Encoder

A ser.soft Talk by
Florian (@ffried) & Daniel (@dh_f)



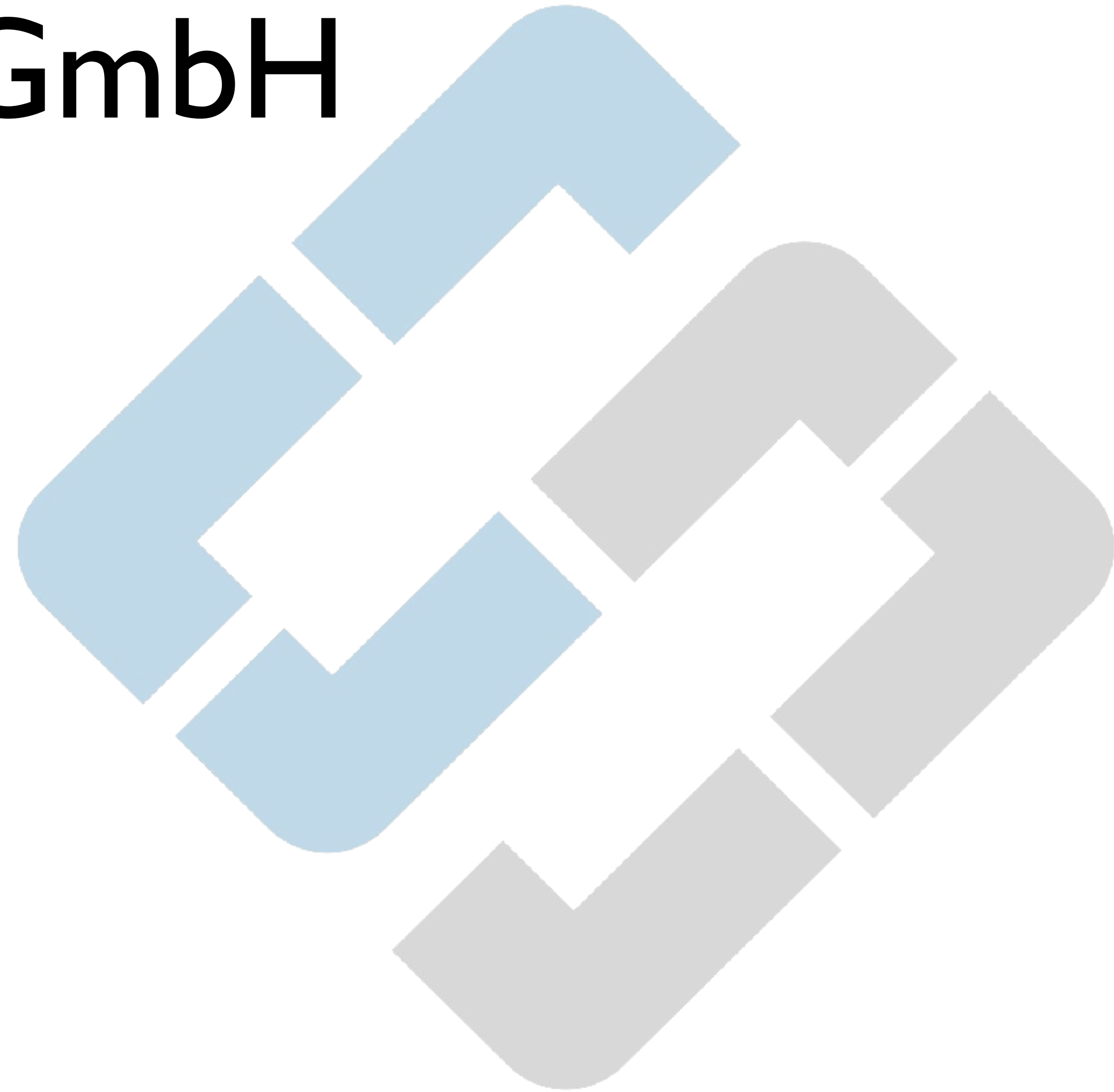
Macoun



Wer sind wir überhaupt?

ser.soft GmbH

- 2001 gegründet
- Sitz in Kürnbach
- Niederlassung in Karlsruhe-Durlach
- 5 Mitarbeiter
- Kerngebiet Softwareentwicklung
- Seit 2009 auch für mobile Endgeräte
- 3 Apps im Store (2 iOS, 1 macOS)



JavaScript Object Notation

JSON

{ JSON: Swift }

{ JSON: Swift }

- NSJSONSerialization / Dictionary<String,Any>
- Vapor (Node / JSON), PerfectLib
- SwiftyJSON
- Argo
- FFFoundation (github.com/ffried/FFFoundation)
- uvm...

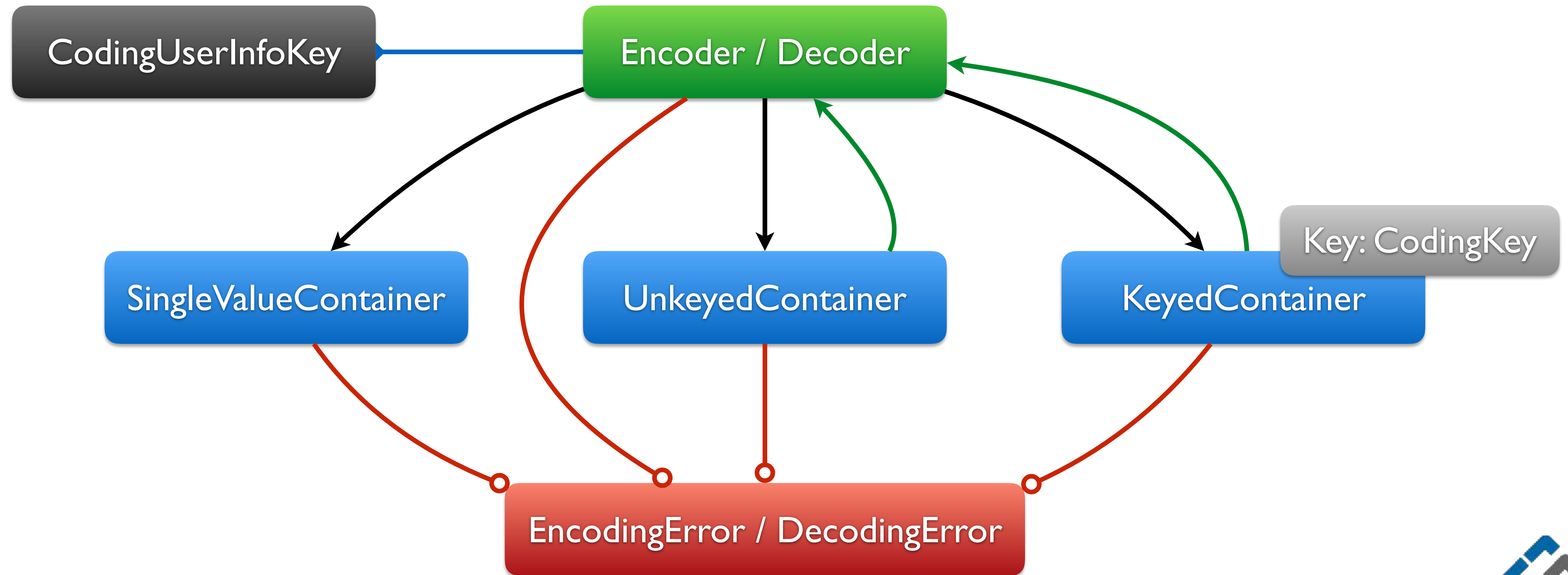
Ablauf

- Struktur von Codable & Co.
- Implementierungen in Foundation
- Flexibilität und deren Grenzen
- Eigene Datenformate?

Struktur von Codable & Co.

- 11 protocols
 - Encodable & Decodable (= Codable typealiases)
 - Encoder & Decoder
 - SingleValue-, Keyed-, Unkeyed-Container (Encoding & Decoding)
 - CodingKey
- 3 structs (CodingUserInfoKey, KeyedEncoding-, Decoding-Container)
- 2 enums (EncodingError, DecodingError)

Struktur von Codable & Co.



Implementierungen in Foundation

- JSON
 - JSONEncoder
 - JSONDecoder
- Property Lists
 - PropertyListEncoder
 - PropertyListDecoder

Flexibilität und deren Grenzen

- Code vom Compiler generiert
 - CodingKeys & Methoden
 - Methoden
 - Weder noch
- Legacy APIs können abgedeckt werden
 - Kann nicht vom Compiler generiert werden

Demo

Eigene Datenformate?

- Encoder / Decoder & Container selbst implementieren
- Tücken
 - Coding Path
 - SuperEncoder / SuperDecoder
 - Value Types mit Reference Semantics

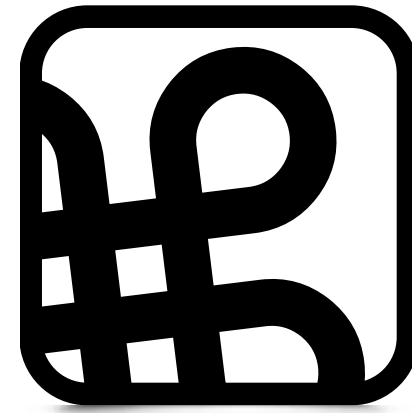
Demo

Fragen?

Code / Kontakt

- Source: https://github.com/sersoft-gmbh/Codable_Macounl7
 - Issue für Thema “IniEncoder” → Voting
- Email: info@sersoft.de
- Web: www.sersoft.de
- Twitter:
 - Florian: @ffried
 - Daniel: @dh_f

Vielen Dank



Macoun