## Extending kotlinx.serialization functionality with Arrow Meta

Karin-Aleksandra Monoid

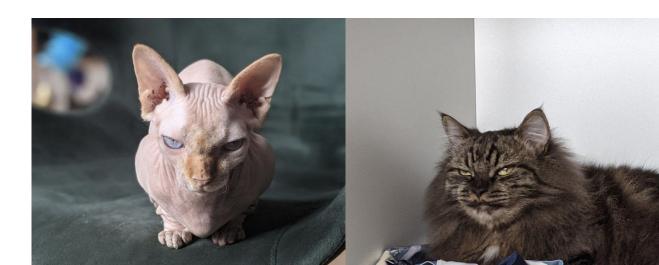
Joker, 2021



- Senior Software Engineer at CircleCl
- Background in Kotlin & Scala (both front and backend)
- Living in Berlin with two cats
- GDE in Kotlin



- Senior Software Engineer at CircleCl
- Background in Kotlin & Scala (both front and backend)
- Living in Berlin with two cats
- GDE in Kotlin



Agenda

#### Why?

#### What is Arrow Meta?

#### Sample project

Showtime!

#### **Project Stack**

- Just backend with Ktor + Arrow
- Jackson was kind of "default"
- Smooth but not Kotlin-first

#### kotlinx.serialization – pros 🚀

- Kotlin-first
- No boilerplate for sealed classes

#### kotlinx.serialization - cons

- Not a lot of serializers out of the box
- No global naming strategy

#### **Global Naming Strategy**

UPPER\_SNAKE\_CASE -> camelCase

USER\_NAME -> userName

#### Jackson: Object Mapper

val objectMapperUpperCamelCase =
 jacksonObjectMapper().apply {
 propertyNamingStrategy =
 PropertyNamingStrategies.UPPER\_SNAKE\_CASE
}



### Fields name policy converters #33



gildor opened this issue on 17 Nov 2017  $\cdot$  21 comments

https://github.com/Kotlin/kotlinx.serialization/issues/33

**Global Naming Strategy** 

# @SerialName("USER\_NAME") val userName: String

### What is Arrow Meta?



What is Arrow Meta?

#### **"Functional companion**

to Kotlin's Compiler"



Image source: https://arrow-kt.io/

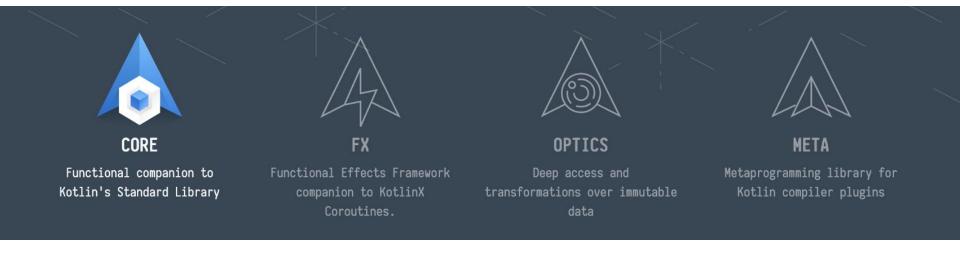
3 circleci

GIF source: https://github.com/arrow-kt/arrow-meta

**O** circle**ci** 

#### Why Arrow Meta?

#### • Already using other part of Arrow for FP



#### Why Arrow Meta?

- More functionality than KSP/KAPT
- Easier than kotlinc

# Sample Project



#### Link to repo

#### https://bit.ly/3C2lHGp



#### Setting: a Zoo

- various animals (actually, only cats and dogs)
- various sources (= 3rd party services)
- various standards of documentation (in JSON format with different naming policies)

# Live coding time



#### Thoughts

- It works
- Not production ready
- Requires more knowledge than expected
- KSP is preferable if it fits



- https://meta.arrow-kt.io/
- <u>https://github.com/arrow-kt/arrow-meta-examples</u>
- <u>https://paranoidmonoid.github.io/articles/Kotlin%20and%20frien</u> <u>ds/Naming%20strategy%20plugin</u>