Cats Ecosystem

Agenda

- What is the Cats?
- Functional Programming & Cats
- What are the main features / elements of Cats Core Library
- Intro to Cats Effect

What is the Cats?

Cats Core

Cats is a library which provides abstractions for functional programming in the Scala language.

Cats Effect

Cats Effect is a high-performance, asynchronous, composable framework for building real-world applications in a purely functional style within the Cats ecosystem

IO Monad

Why functional programming?

- Fighting with **complexity**, especially in concurrent and multithreading environments
- More efficient resource management
- It removes one important **dimension** of **complexity**

  - *To understand a program part (a function) you need no longer account for the possible **histories** of executions that can lead to that program part*

FP Concept

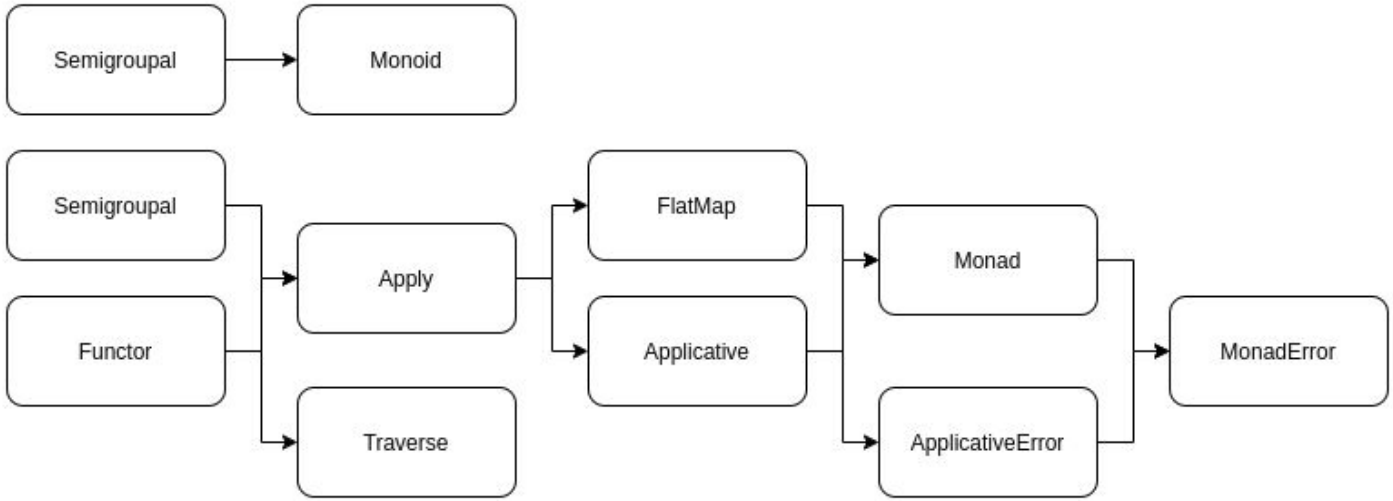
- Process of building software by
  - composing **Pure Functions** (**Referential Transparency**)
  - avoiding
    - **Mutable Data**
    - **Side Effects** (basically encapsulated into **lazy** data abstractions)
    - **Shared State**

- Application state flows through **Pure Functions**

- In reality modern scala apps use **hybrid** between **FP & OOP**

Cats & FP

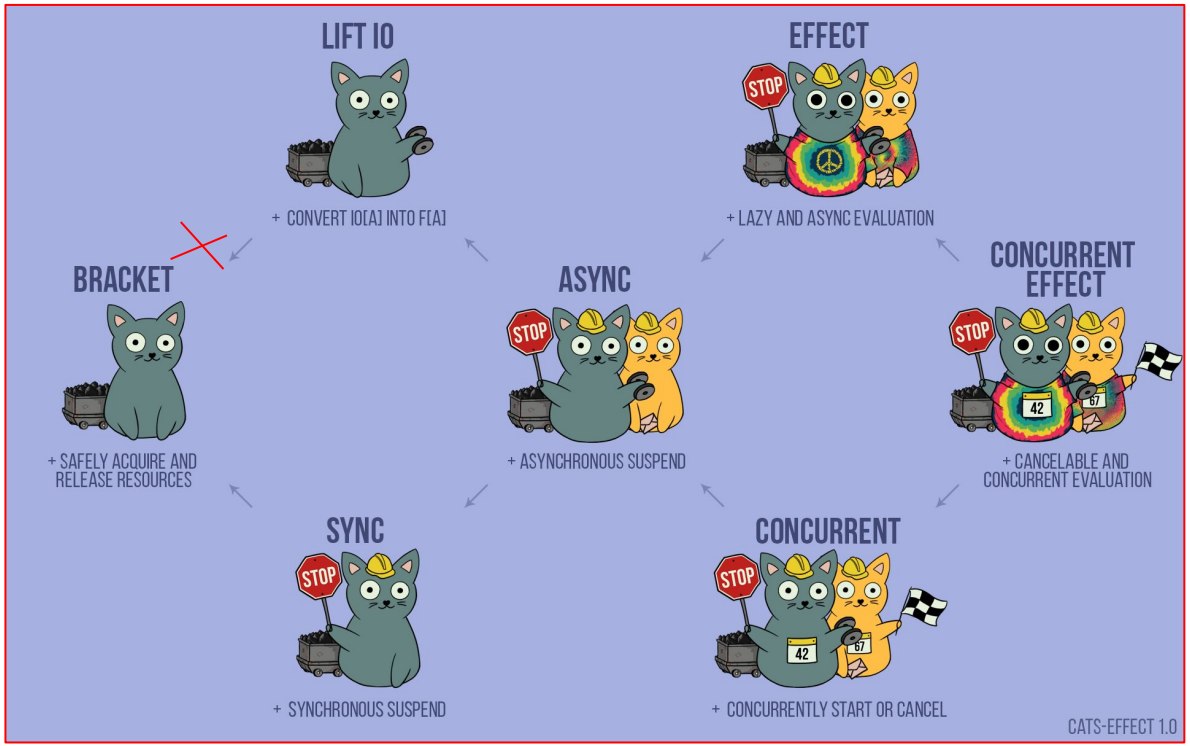
- It’s difficult to follow FP principles in real apps
- Constructive math: category theory -> comes on the scene
  - It helped a lot in order to create data abstractions for practical FP (type class hierarchy)
  - “Cats” is a playful shortening of the word “*category*”.
  - Every library that wants to be compatible with Cats need to extend type classes below:

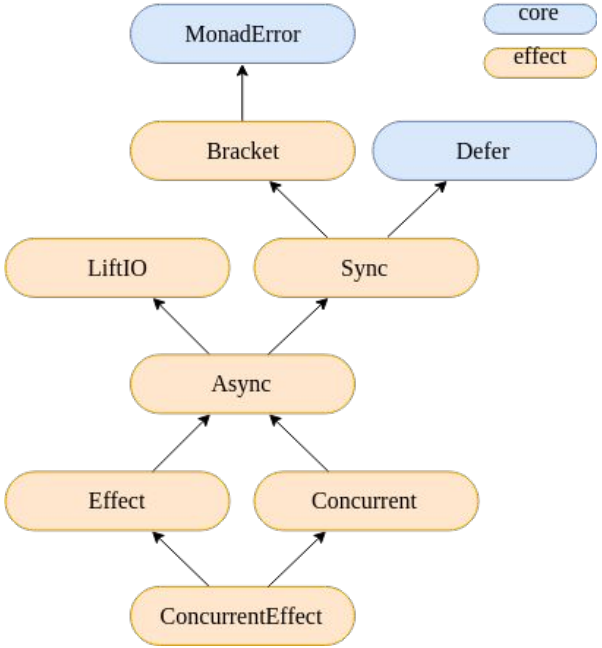


Deep dive into code

https://github.com/gleb-streltsov-4by/scala-workshops/tree/master/src/main/scala/com/workshops/cats

Cats-effect typeclasses





Cats Effect

https://docs.google.com/presentation/d/1K4ZOdxLXTJP0bmp7q0dD2alQvicPgVrfJ5sdTJmYxXc/edit?usp=sharing

Alternatives to cats

ZIO

- Appeared around 2019
- Younger ecosystem
- Risky for production
- Built-in DI capability -> layers
- Nice built-in pattern for error handling
- Benchmarks are similar to Cats

Monix

- Appeared around 2014
- Lack of compatibility with latest stable FP libs and frameworks
- Rare in new projects
- Tend to be replaced with Cats or ZIO