

SAJÁT PLATFORMOM SZÜLETÉSE

PLATFORM? 

RÓLAM

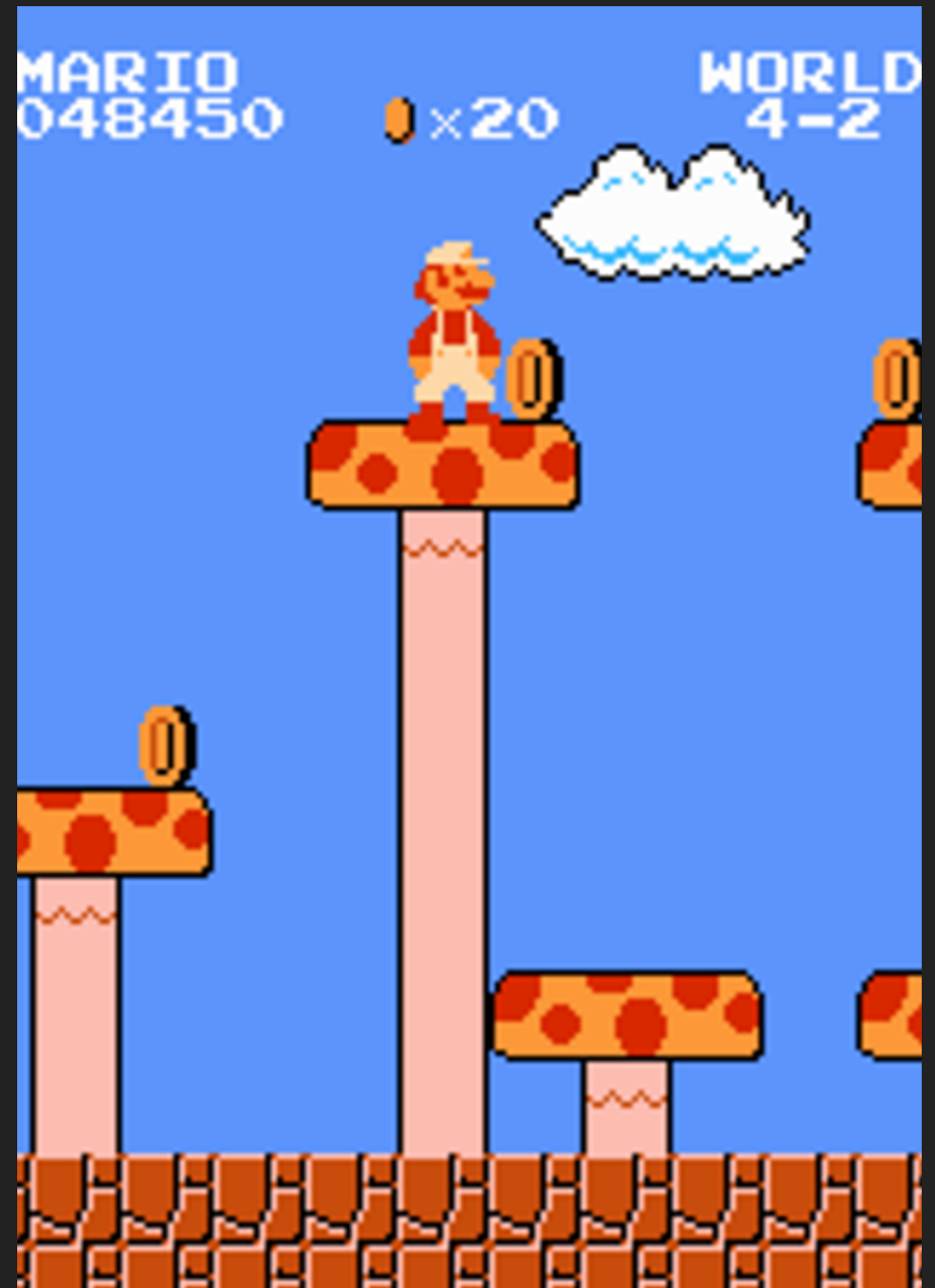
- ▶ Lead systems engineer – DevOps
- ▶ PhD hallgató / Oktató / Pataki Norbert padawanja
- ▶ Mortal Kombat 1, Kombat League – Demi God 🦂🔥
- ▶ Szeretek edzeni, alkalmanként deszkás sportokat űzni, evezni
- ▶ Van egy nagyon okos mudink

ELŐADÁSOM

- ▶ Mit nevezünk platformnak?
- ▶ Szoftver platformok
- ▶ Konténerizáció, Function As A Service
- ▶ LambdaKube
- ▶ A mi platformunk!

MIT NEVEZÜNK PLATFORMNAK?

- ▶ Platform – főnév
 - ▶ Egy emelt sík, amelyen személyek vagy tárgyak állhatnak.



PLATFORMOK KÜLÖNBÖZŐ KONTEXTUSOKBAN

- ▶ Influencer-ek / Média szakértők
 - ▶ Instagram, Facebook, YouTube, Tv, TikTok...
- ▶ Tartalomgyártók
 - ▶ Twitch, YouTube, TikTok...
- ▶ Operációs rendszerek
 - ▶ Macintosh, Windows, Linux*, BSD, Solaris...
 - ▶ iOS, Android, Tizen...
- ▶ Gaming platformok
 - ▶ PlayStation, Nintendo, XBox, PC, Macintosh(🤔)...
 - ▶ Unity3D, Unreal Engine, GoDot, Löve
- ▶ Alacsonyabb szinteken
 - ▶ x86, x64, stb.

SZOFTVER PLATFORMOK

- ▶ Absztrakciós szint és domain szerint kategorizálhatók
- ▶ Az adott domain és szint általános műveleteit támogató szolgáltatásokat/interfészeket nyújtanak
- ▶ Fejlesztő eszközöket biztosít



NODE JS

- ▶ Szolgáltatások:
 - ▶ Interpreter – futtatási környezet
 - ▶ Fájlrendszer elérés
 - ▶ Hálózati kommunikáció
 - ▶ Memória kezelés
- ▶ Fejlesztői eszközök:
 - ▶ Debugger
 - ▶ Csomag kezelő

KONTÉNERIZÁCIÓ, FAAS

- ▶ A konténerek könnyű súlyú virtuális gépek
- ▶ Egy konténerben egy alkalmazást szállítanak annak minden függőségével és jól definiált interfészekkel
- ▶ Egy konténer orkesztrációs rendszer konténereket menedzsel több host között, kezeli a kommunikációt és az erőforrásokat



FUNCTION AS A SERVICE

- ▶ Több szolgáltatás által használt függvényeket szervez külön szolgáltatásba
- ▶ A függvényeket hálózaton keresztül hívjuk
- ▶ Létező megoldások:
 - ▶ Azure functions, Google Cloud Functions, AWS Lambda, Kubeless

LAMBDAKUBE

- ▶ Szeretnénk biztosítani a FAAS függvényhívások típushelyességét
- ▶ Vizsgáljuk a FAAS függvények kompozíciós lehetőségeit
- ▶ Készítünk egy tisztán funkcionális programnyelvet a számítások kifejezésére
- ▶ Készítünk egy fordítót
- ▶ Fejlesztői eszközök – célközönség
- ▶ Világbéke



LAMBDAKUB-EH?

▶ PROGRAMNYELVET AKAROK!

- ▶ Oké, használtam már jó párat, és elég jól kritizálom őket, de azért...
- ▶ Rendszerben, írtam már fordítót, de Norbi kedvéért Haskell-ben teszem most... 😊
- ▶ Tényleg jól akarom csinálni, így felelevenítettem a tudásom
 - ▶ Típusrendszerek
 - ▶ Fordprog
 - ▶ Csörnyei Zoltán – Lambda Kalkulus

LAMBDAKUB-EH?

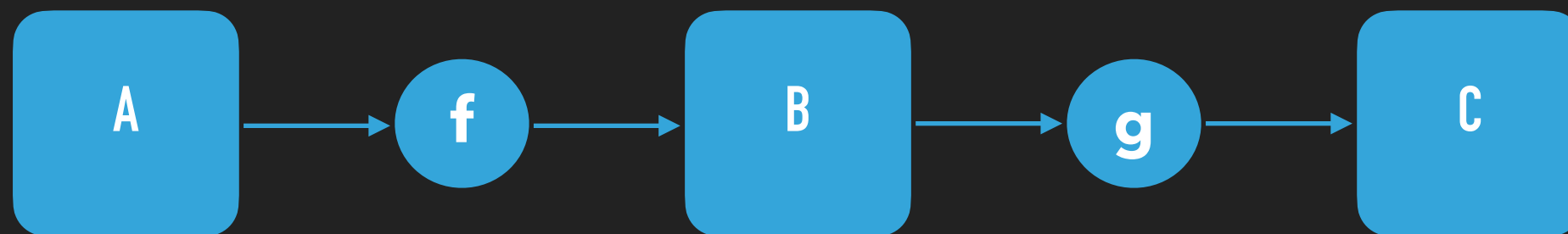
- ▶ Környezet?
 - ▶ Kubernetes + OpenFAAS
- ▶ MiRE fordítók?
 - ▶ Python kód
 - ▶ OpenFAAS projekt leíró
 - ▶ Később Helm – Kubernetes resource csomagkezelő – telepíthetőség

KOMPOZÍCIÓ

$$f \in A \rightarrow B, g \in B \rightarrow C, \circ \in (B \rightarrow C) \rightarrow (A \rightarrow B) \rightarrow (A \rightarrow C)$$



$$g \circ f \in A \rightarrow C$$



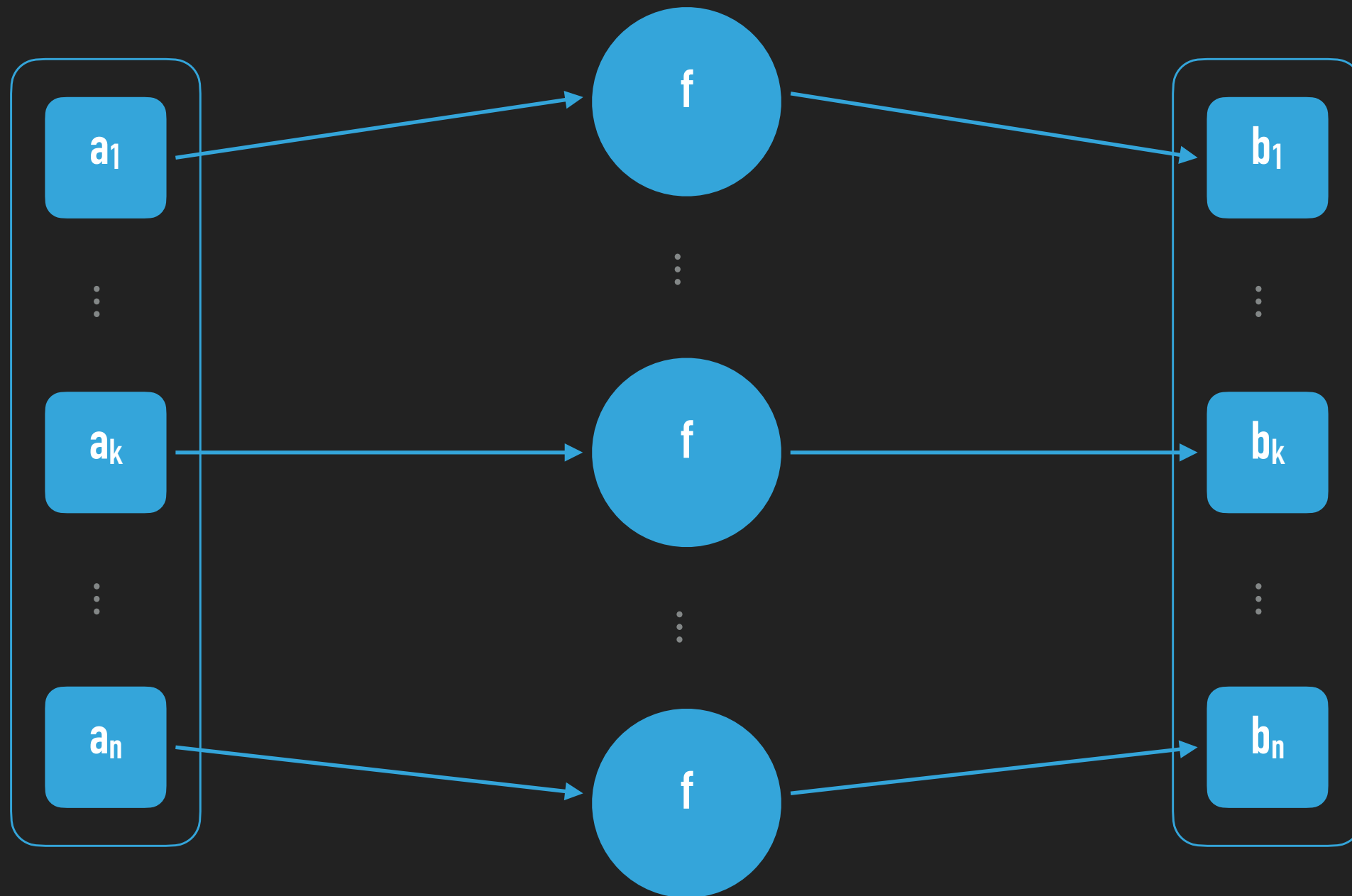
MAP

$f \in \mathbb{A} \rightarrow \mathbb{B}, l \in [\mathbb{A}], \text{map} \in (\mathbb{A} \rightarrow \mathbb{B}) \rightarrow [\mathbb{A}] \rightarrow [\mathbb{B}]$



$\text{map}(f, l) \in [\mathbb{B}]$

MAP



REDUCE

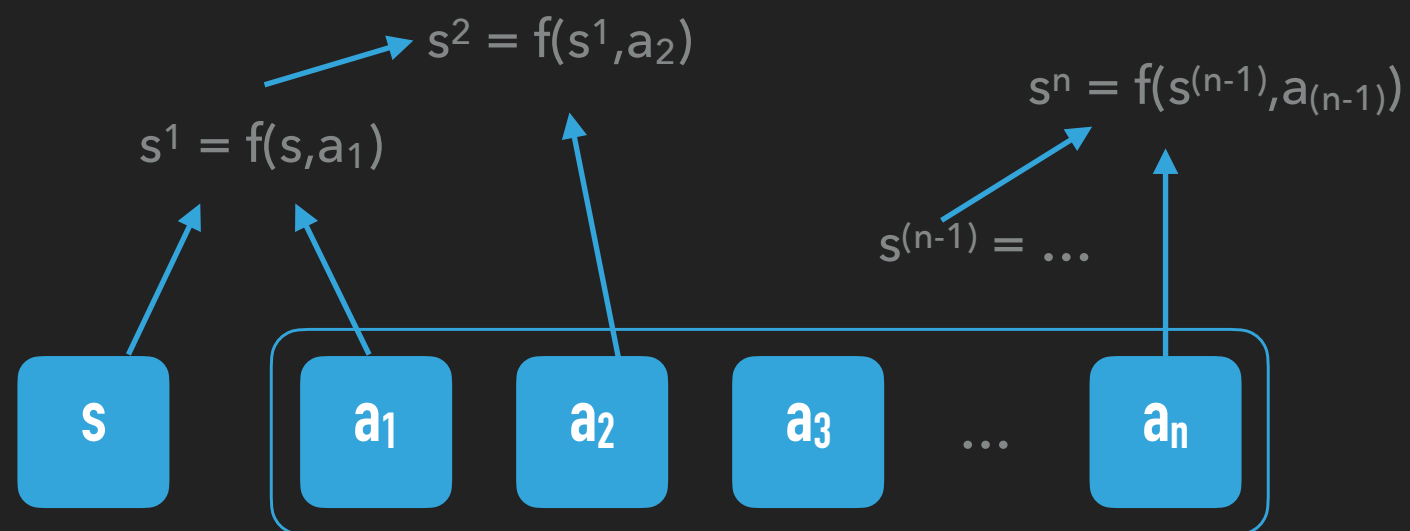
$$f \in A \rightarrow B \rightarrow B, s \in B, l \in [A],$$

$$reduce \in (A \rightarrow B \rightarrow B) \rightarrow B \rightarrow [A] \rightarrow B$$

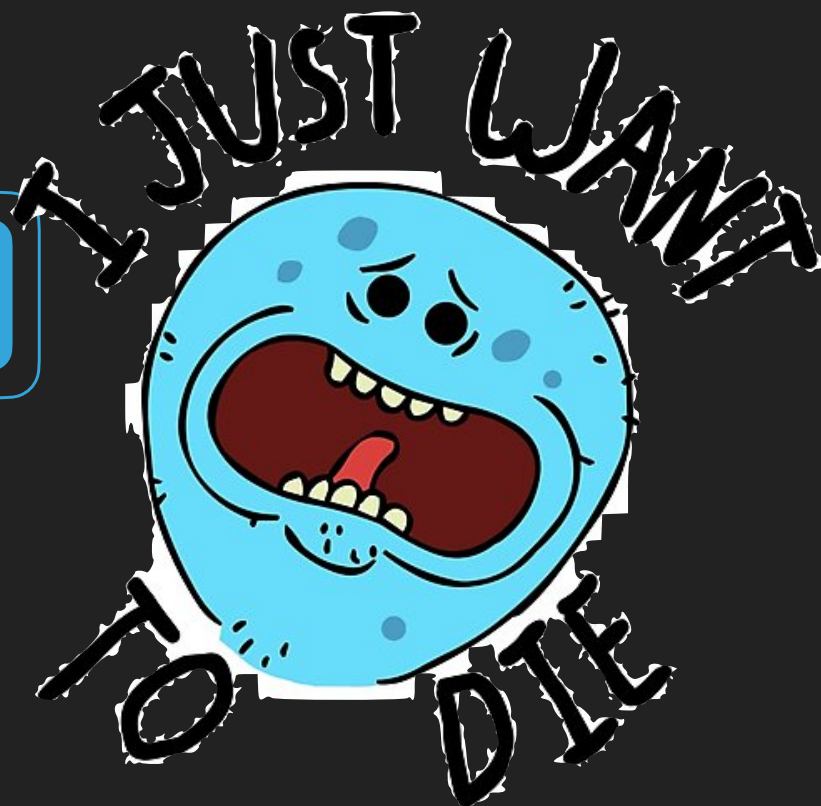
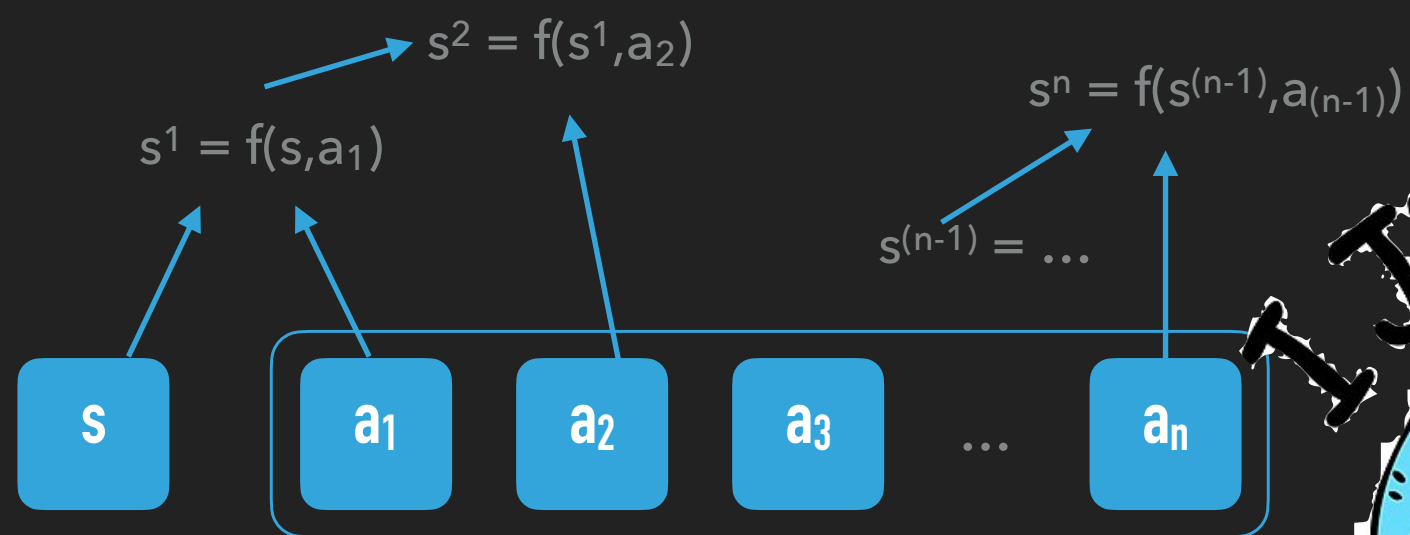


$$reduce(f, s, l) \in B$$

REDUCE



REDUCE

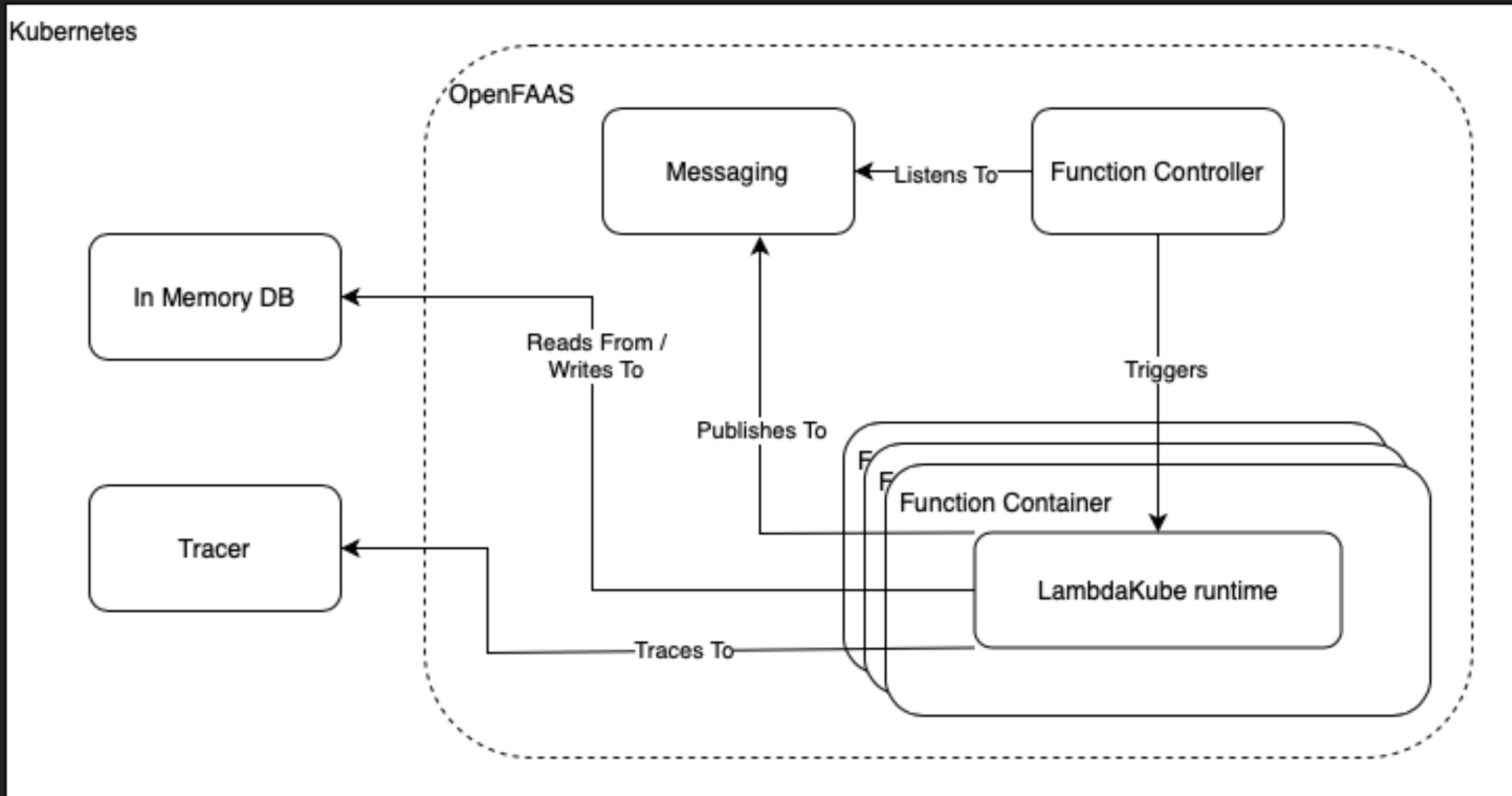


LÉPÉSEINK

- ▶ Környezet elkészítése
- ▶ Funkcionális programok alap konstrukcióinak támogatása
 - ▶ Template kódok kialakítása
- ▶ Mielőtt úrrá lenne a káosz, jó volna egy tracing eszköz
 - ▶ Jaeger? Jöhet! – Ma már OpenTracer
 - ▶ Hova rakjuk?
 - ▶ Hogyan integráljuk?
 - ▶ Egyáltalán milyen rétegeink vannak?
 - ▶ Ki - mihez fér hozzá? Kit-mi érint?
 - ▶ Hol van a mi termékünk?
 - ▶ Mi egyáltalán a termékünk?
 - ▶?

PLATFORM?

A MI PLATFORMUNK!!!



RÉTEGEK

- ▶ Kubernetes platform
 - ▶ Konténerek kezelése – szolgáltatások futtatásra
 - ▶ Service discovery
- ▶ LambdaKube platform környezet
 - ▶ OpenFaas
 - ▶ Függvény konténerek kezelése
 - ▶ Messaging
 - ▶ In-memory DB
 - ▶ Tracer
- ▶ LambdaKube runtime
- ▶ Fordító
- ▶ Programnyelv

FORDÍTÓ

- ▶ Fordító architektúra
 - ▶ CLANG-like modern felépítés
 - ▶ Frontend
 - ▶ Jelenleg parsec, váltunk mparsec-re
 - ▶ Backend
 - ▶ Közte: Internal representation (IR)

PROGNYELV

- ▶ Haskell-like syntax, ponttal a definíciók végén...

```
increment :: Int -> Int
```

```
increment a = (a + 1).
```

```
double :: Int -> Int
```

```
double a = a * 2.
```

**KÖSZÖNÖM A FIGYELMET!
KÉRDÉSEK?**

Révész Ádám