

Как избежать постройки «велосипеда» на примере реальной задачи



Андрей
Богданов

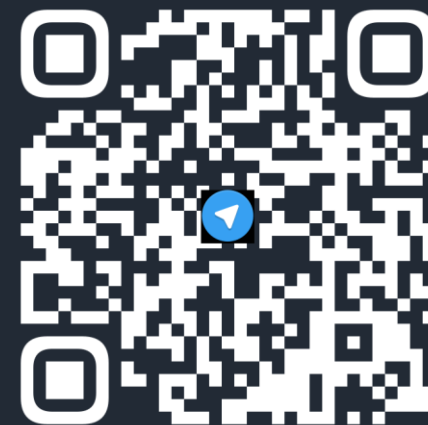
VK Cloud

Joker<?>

Кто я такой?



- Архитектор. Занимаюсь проектированием высоконагруженных систем
- Разработчик. В основном на Java и JavaScript
- Велосипедостроитель. За плечами реализация RAD среды для разработки на JavaScript



Идеальный план

- ✓ Задача / 4
- ✓ Муки выбора / 13
- ✓ Проверка на проф. пригодность / 20
- ✓ Итоговое решение / 22
- ✓ Проблемы и решения / 12
- ✓ Выводы / 14

О чем будем говорить?



Контекст



Архитектура имеющейся системы



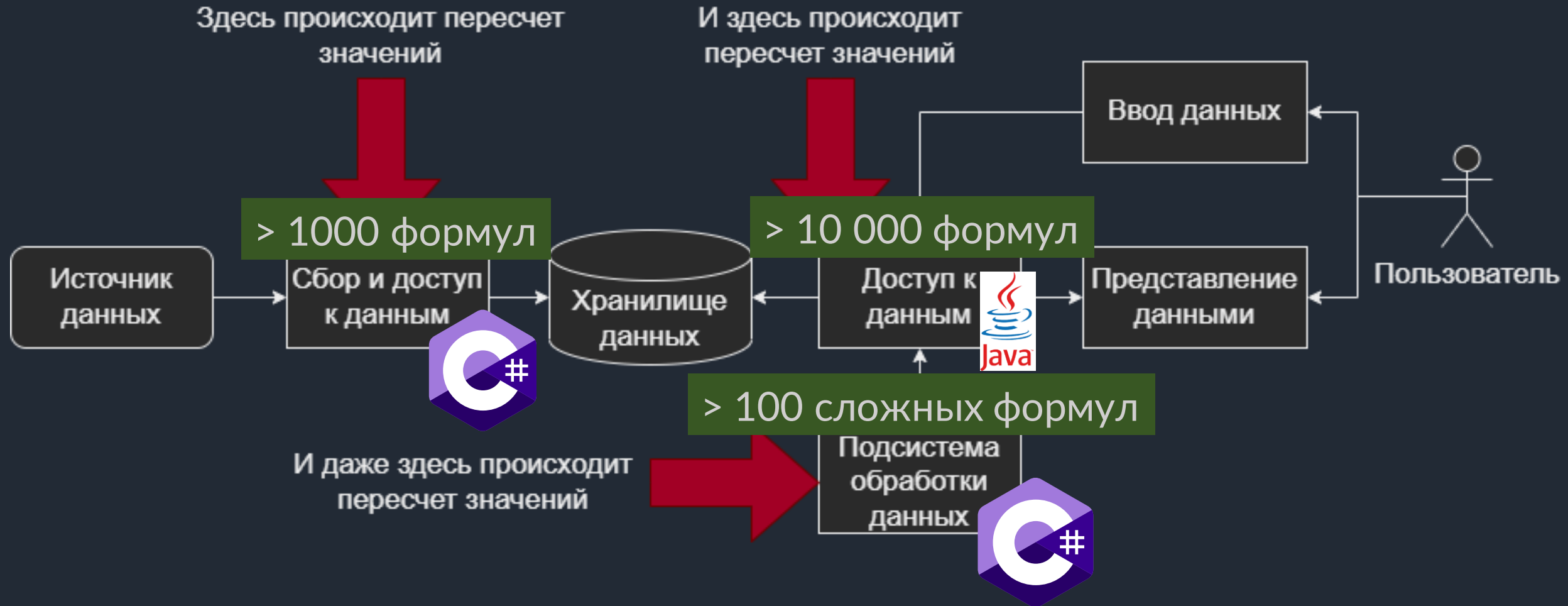
А что не так?



А что еще не так?



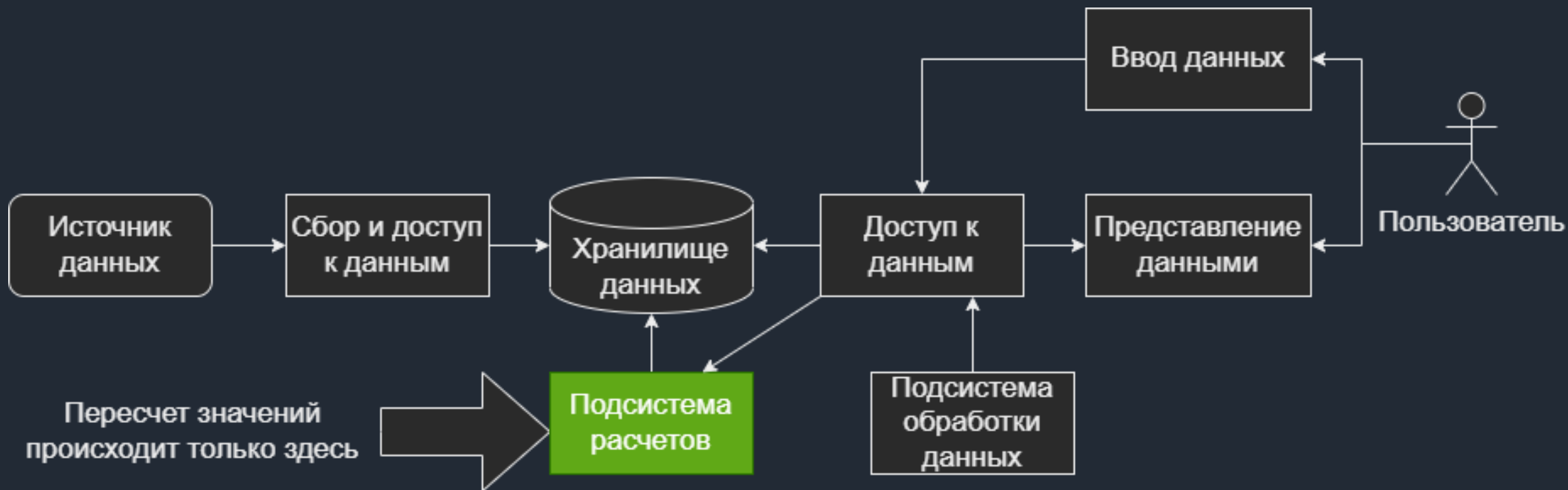
А что еще не так?



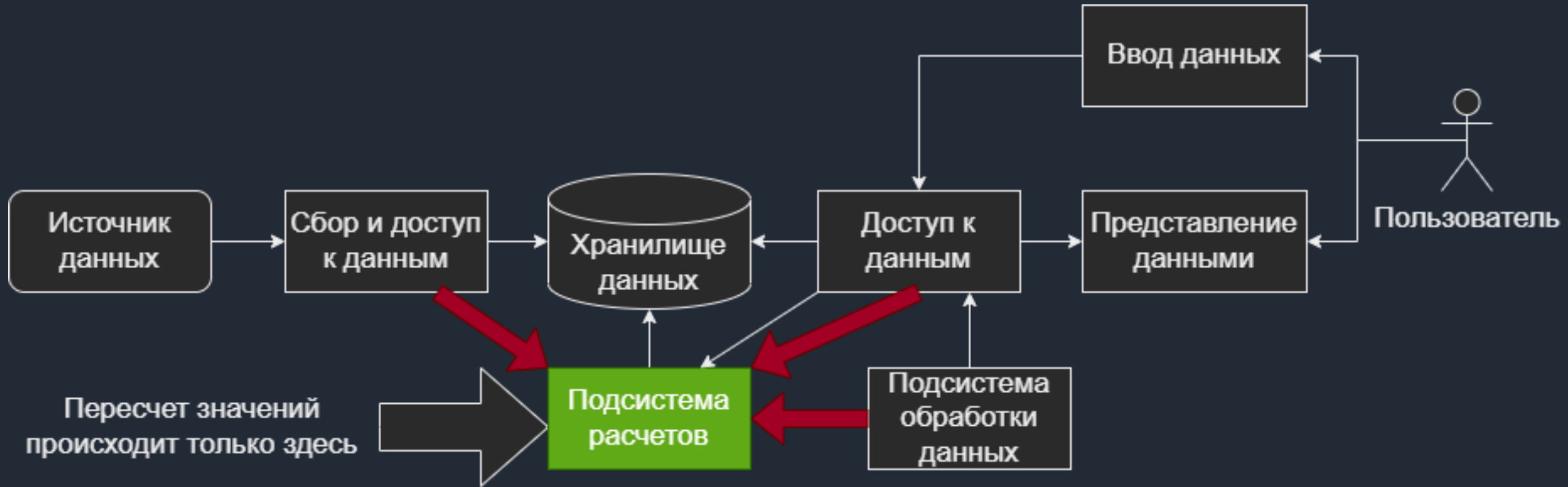
Задача



Что и зачем?



Что и зачем?



Требования нужно
обязательно
«истребовать» с
аналитиков.



Требования

- Cloud native (запуск в K8S)

Требования

- Cloud native (запуск в K8S)
- Исполнять расчеты на MVEL (JVM) и C#(.NET Framework)

Требования

- Cloud native (запуск в K8S)
- Исполнять расчеты на MVEL (JVM) и C#(.NET Framework)
- Количество одновременно работающих расчетов от 100 до 10000

Требования

- Cloud native (запуск в K8S)
- Исполнять расчеты на MVEL (JVM) и C#(.NET Framework)
- Количество одновременно работающих расчетов от 100 до 10000
- Предоставить возможность отладки написанных выражений

Требования

- Cloud native (запуск в K8S)
- Исполнять расчеты на MVEL (JVM) и C#(.NET Framework)
- Количество одновременно работающих расчетов от 100 до 10000
- Предоставить возможность отладки написанных выражений
- Время «холодного» запуска расчета не более 30 мс



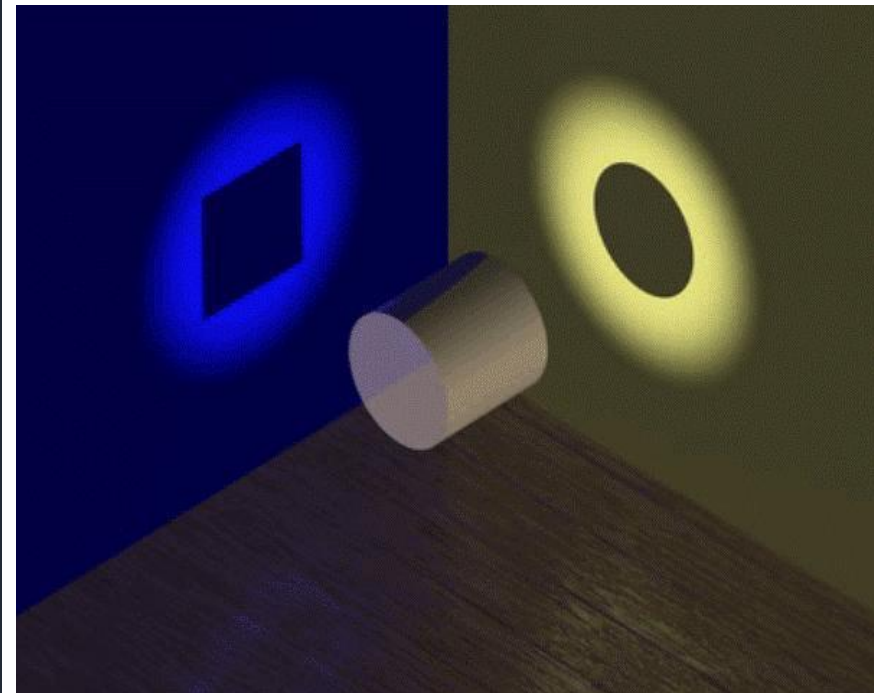
«Вангую» нужен FaaS



Сравниваем популярные FaaS

Требование/Решение	Knative	OpenFaas	fission
Cloud native (запуск в K8S)	✓	✓	✓
Исполнять расчеты на MVEL (JVM) и C# (.NET Framework)	✓	✓	✓
Количество одновременно работающих расчетов от 100 до 10000	✓	✓	✓
Предоставить возможность отладки написанных выражений	✓	✓	✓
Время «холодного» запуска расчета не более 30 мс	✗	✗	✗

А что если применить акторы ...



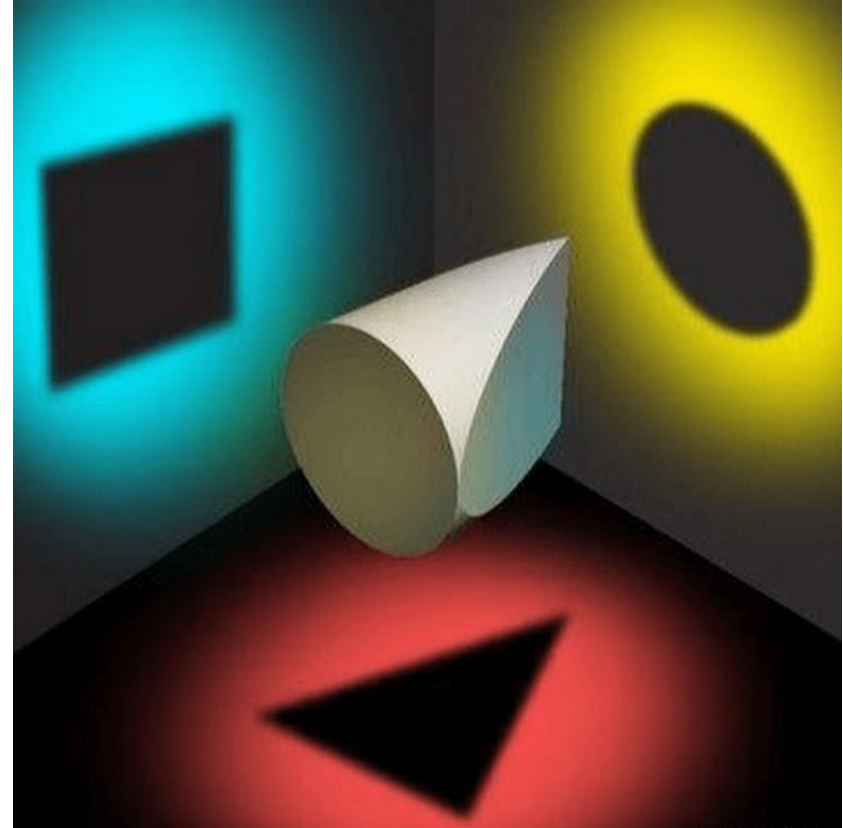
Сравниваем популярные акторные движки

Требование/Решение	Akka.NET	Vert.x	Orleans
Cloud native (запуск в K8S)			
Исполнять расчеты на MVEL (JVM) и C# (.NET Framework)			
Количество одновременно работающих расчетов от 100 до 10000			
Предоставить возможность отладки написанных выражений			
Время «холодного» запуска расчета не более 30 мс			

A word cloud featuring various cloud-native technologies and frameworks. The words are arranged in a circular pattern, with 'apache' being the largest and most central. Other prominent words include 'openfaas', 'proto.actor', 'knix', 'pipelineai', 'goblins', 'openfunction', 'xcraft', 'fission', 'rotor', 'kubeless', 'openwhisk', 'riff', 'vert.x', 'appscale', 'camel', 'nuclio', 'keda', 'kubernetes', 'krustlet', 'orleans', 'faas', and 'virtual'. The words are color-coded in shades of red, green, yellow, and blue, and are written in a casual, handwritten-style font.

openfaas
proto.actor knix
pipelineai goblins
openfunction xcraft
fission apache rotor
kubeless
openwhisk riff
appscale camel nuclio
orleans keda kubernetes
krustlet
faas
virtual

А если виртуальные акторы ...



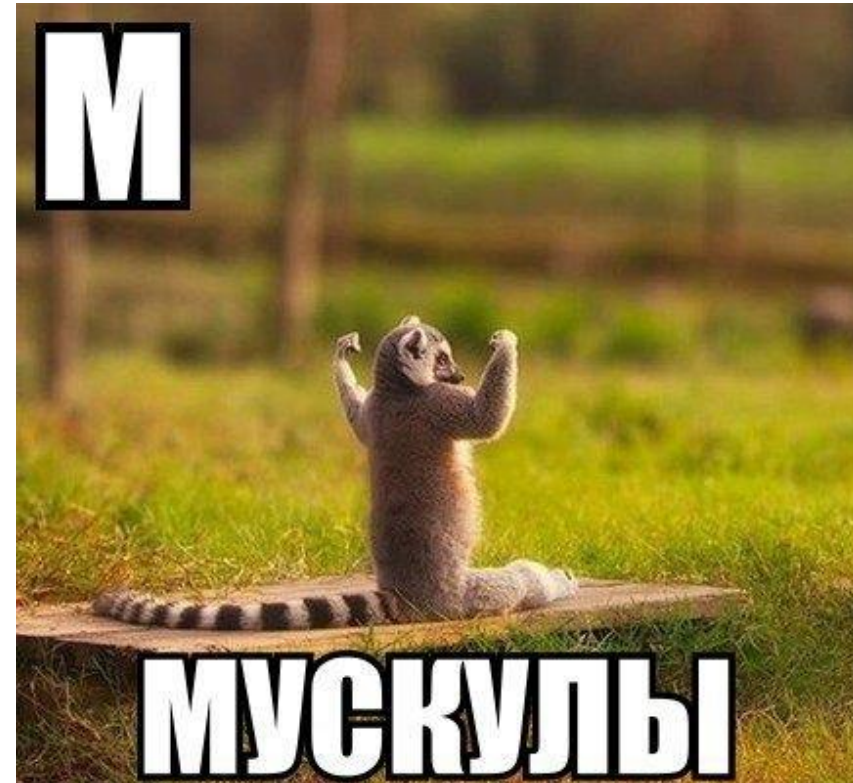
Кажется нашли ...

Требование/Решение	DAPR
Cloud native (запуск в K8S)	✓
Исполнять расчеты на MVEL (JVM) и C# (.NET Framework)	✓
Количество одновременно работающих расчетов от 100 до 10000	✓
Предоставить возможность отладки написанных выражений	✓
Время «холодного» запуска расчета не более 30 мс	✓

Кто слышал про DAPR до
сегодняшнего дня?



Проверка на проф.
пригодность



DAPR - Distributed Application Runtime

✉ Publish & Subscribe

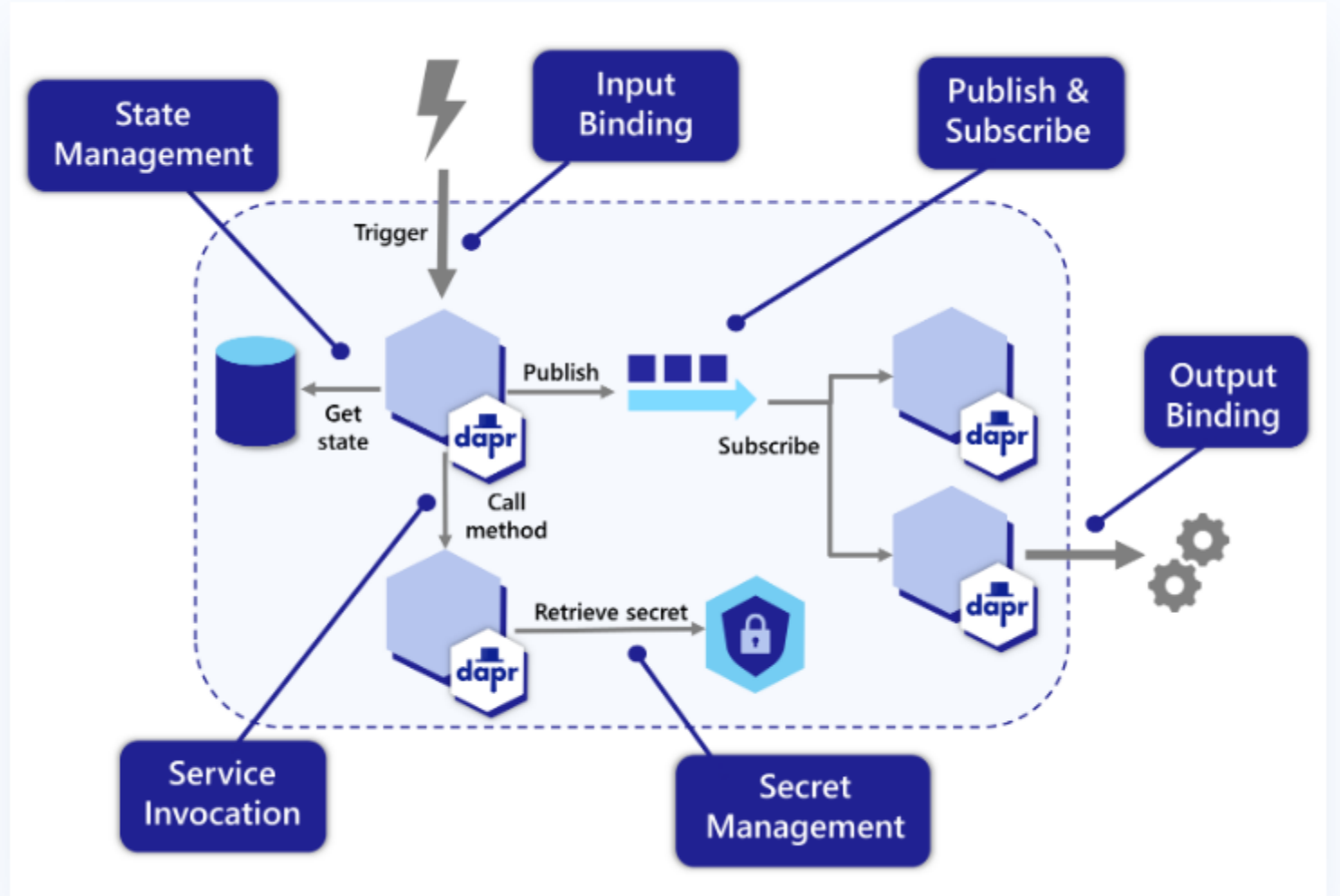
↕ Service Invocation

🛡 Secret Management

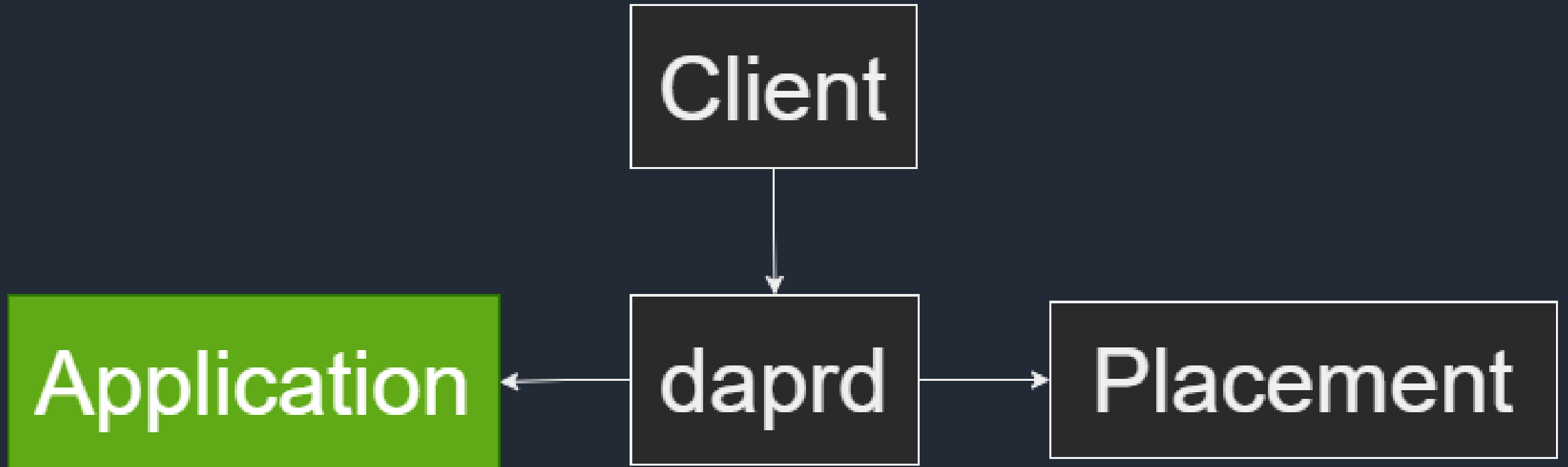
🔌 Input/Output Bindings

🗄 State Management

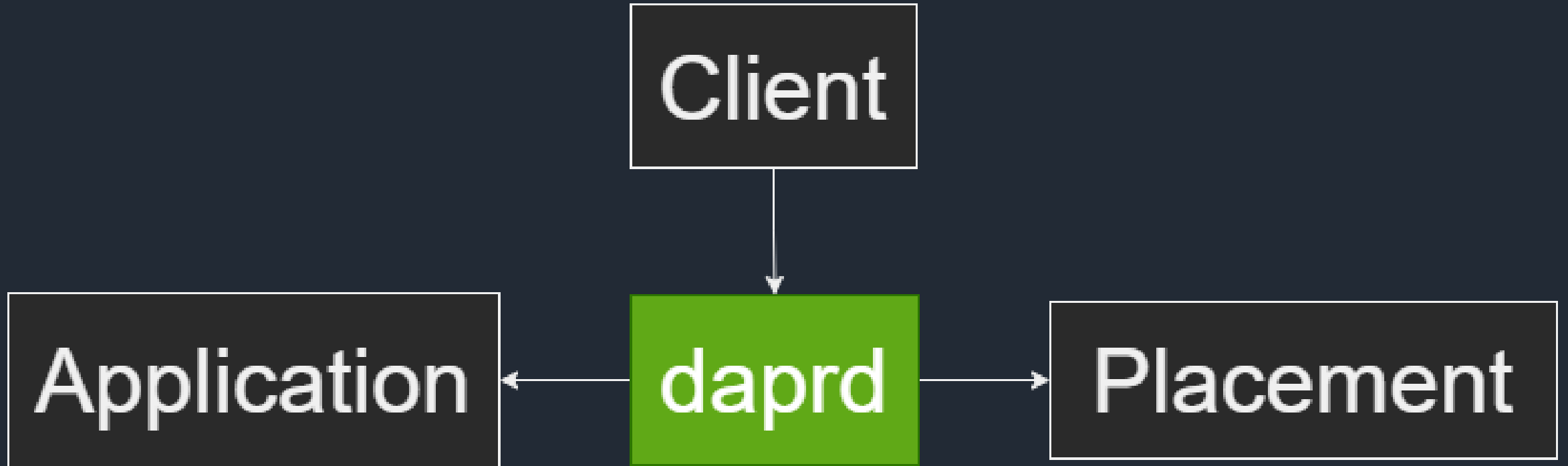
🏠 Virtual Actors



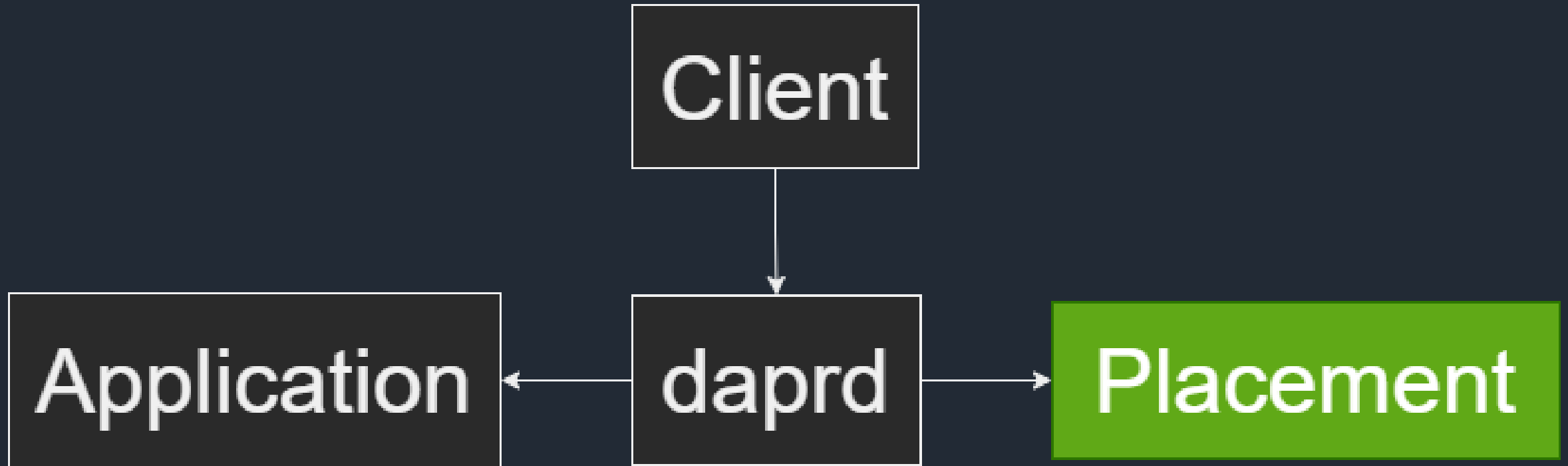
DAPR. Как все устроено в части акторов?



DAPR. Как все устроено в части акторов?



DAPR. Как все устроено в части акторов?



Проверяем требования







- Cloud native (запуск в K8S)

Deploy Dapr on a Kubernetes cluster

Follow these steps to deploy Dapr on Kubernetes.

When setting up Kubernetes you can use either the Dapr CLI or Helm.

For more information on what is deployed to your Kubernetes cluster read the [Kubernetes overview](#)

 Dockerfile	Use the ca-certs bundle that's part of distroless (#4380)
 Dockerfile-debug	Changed debug Dockerfile (#5327)
 Dockerfile-dev	Update linter version a few other places
 Dockerfile-mariner	Mariner fixes (#4761)
 Dockerfile-windows	Allow setting custom certificates in Windows containers (#4741)
 Dockerfile-windows-base	Allow setting custom certificates in Windows containers (#4741)

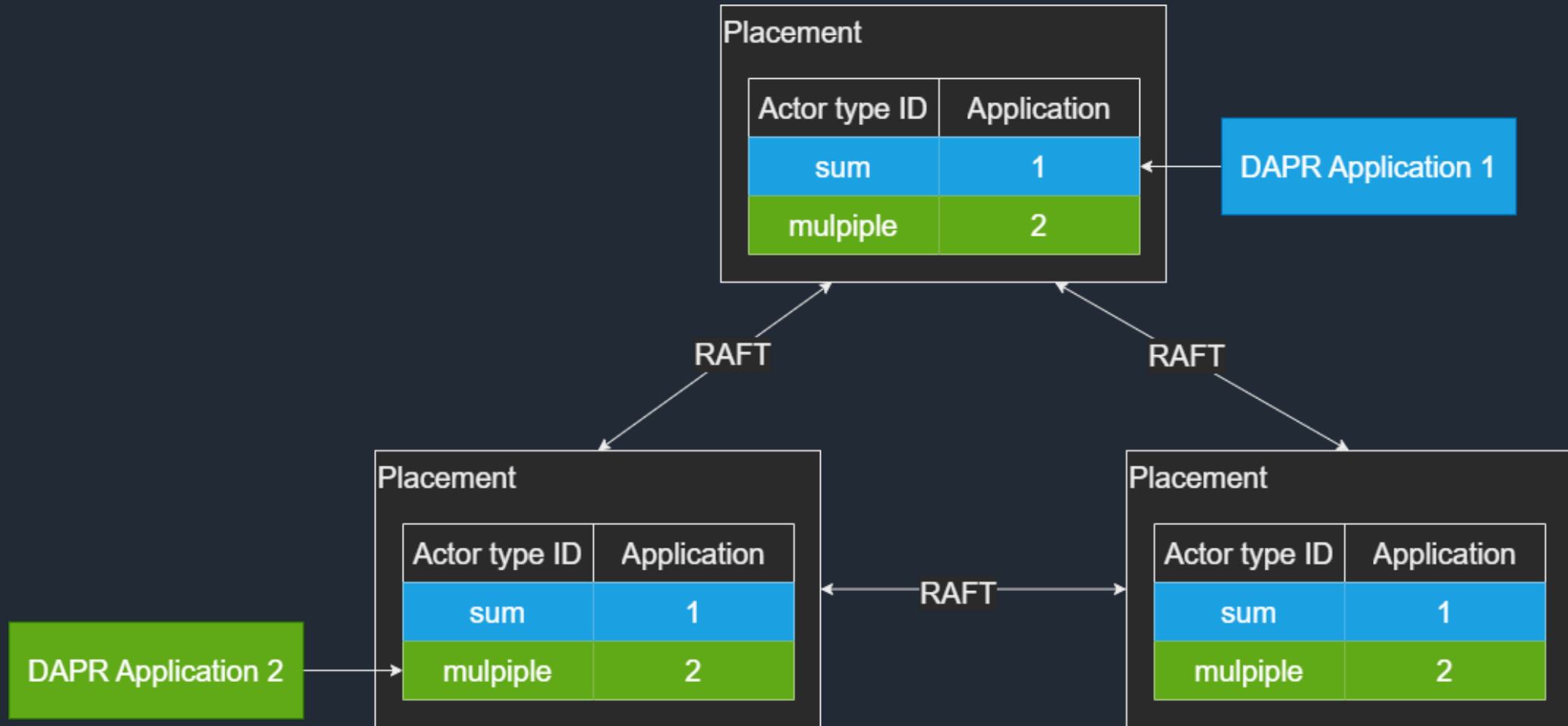
Проверяем требования

- Исполнять расчеты на MVEL (JVM) и C#(.NET Framework)

Language	Status	Client SDK	Server extensions	Actor SDK
.NET	Stable	✓	ASP.NET Core	✓
Python	Stable	✓	gRPC FastAPI Flask	✓
Java	Stable	✓	Spring Boot	✓

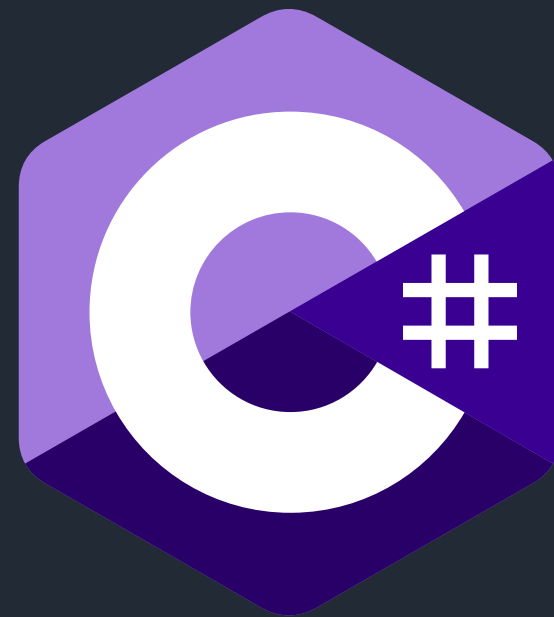
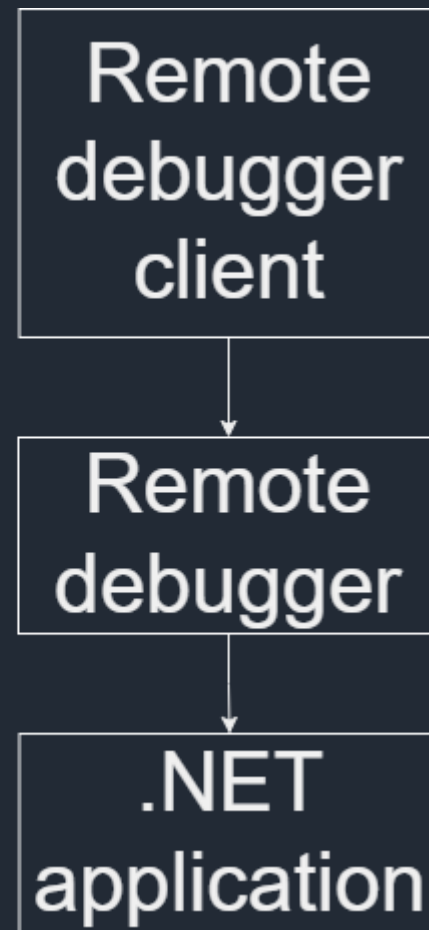
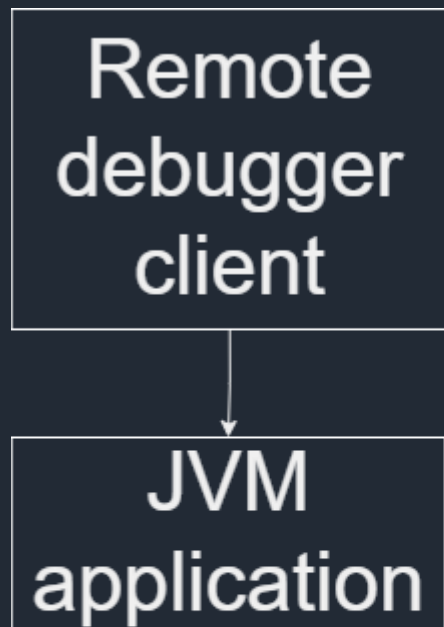
Проверяем требования

- Количество одновременно работающих расчетов от 100 до 10000



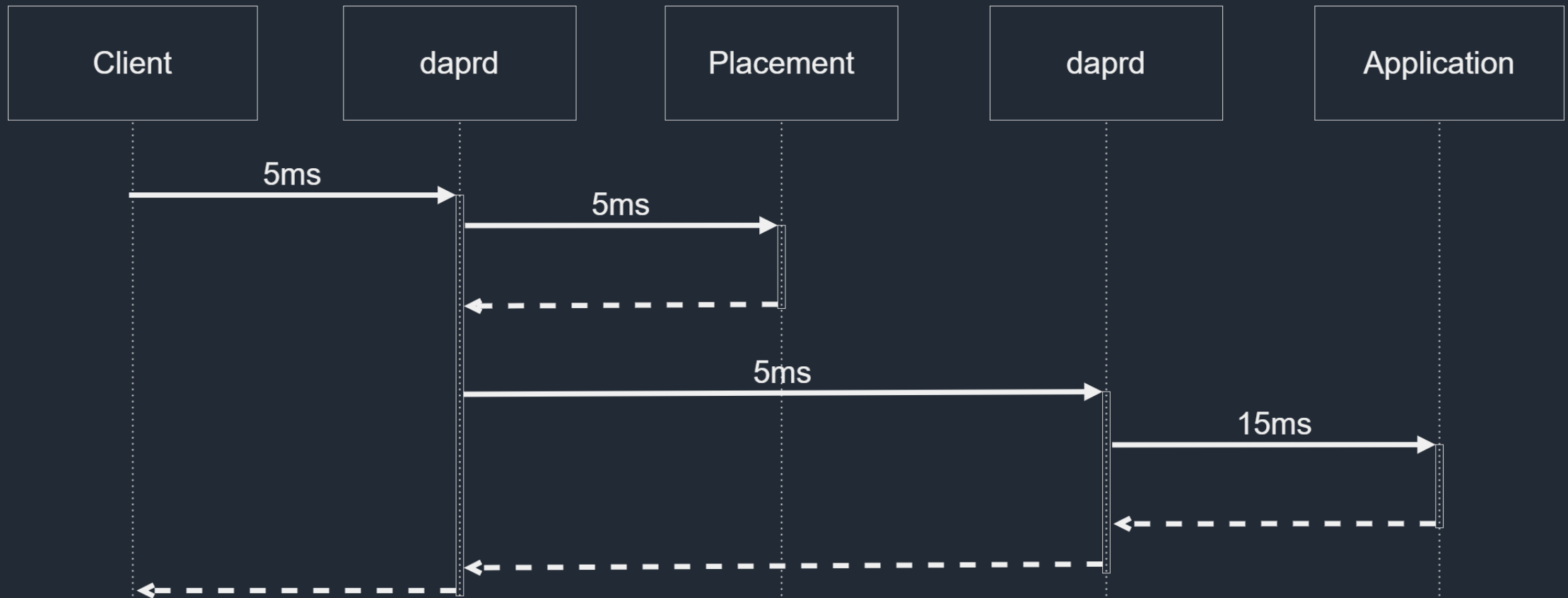
Проверяем требования

- Предоставить возможность отладки написанных выражений

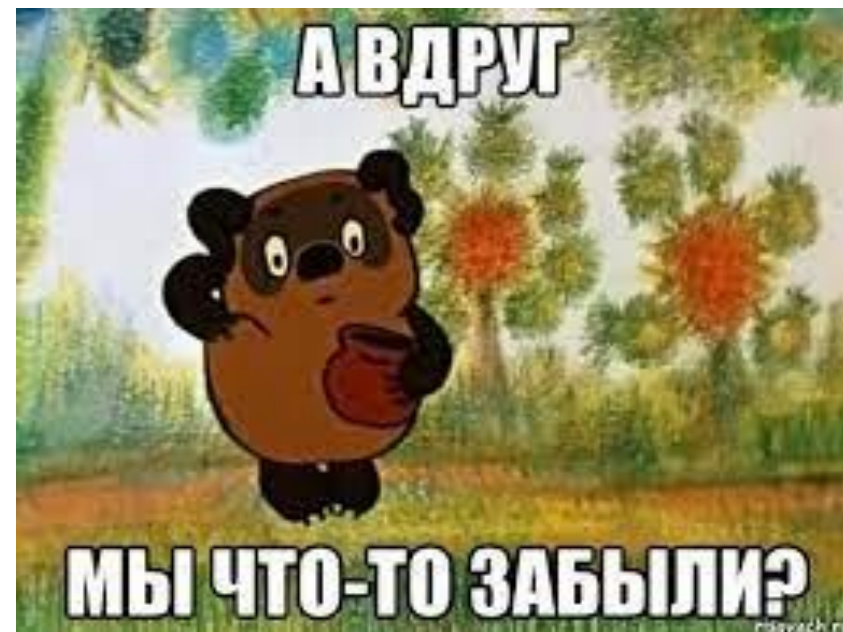


Проверяем требования

- Время «холодного» запуска расчета не более 30 мс

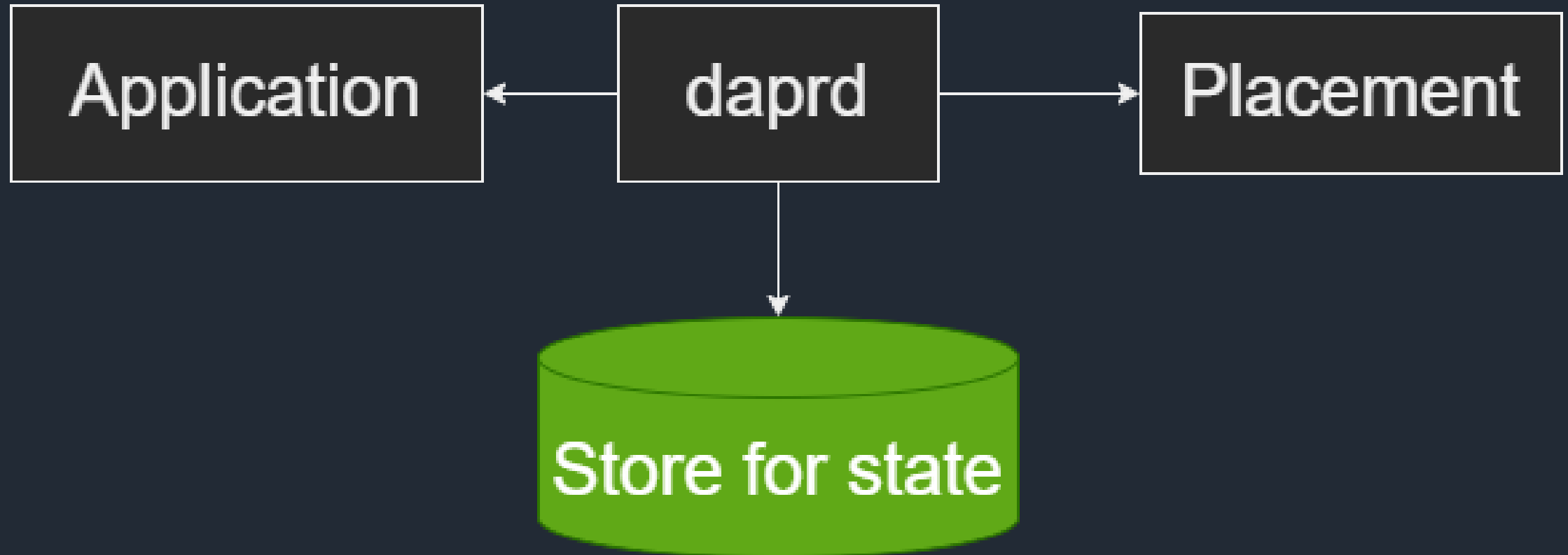


Еще раз проверяем требования



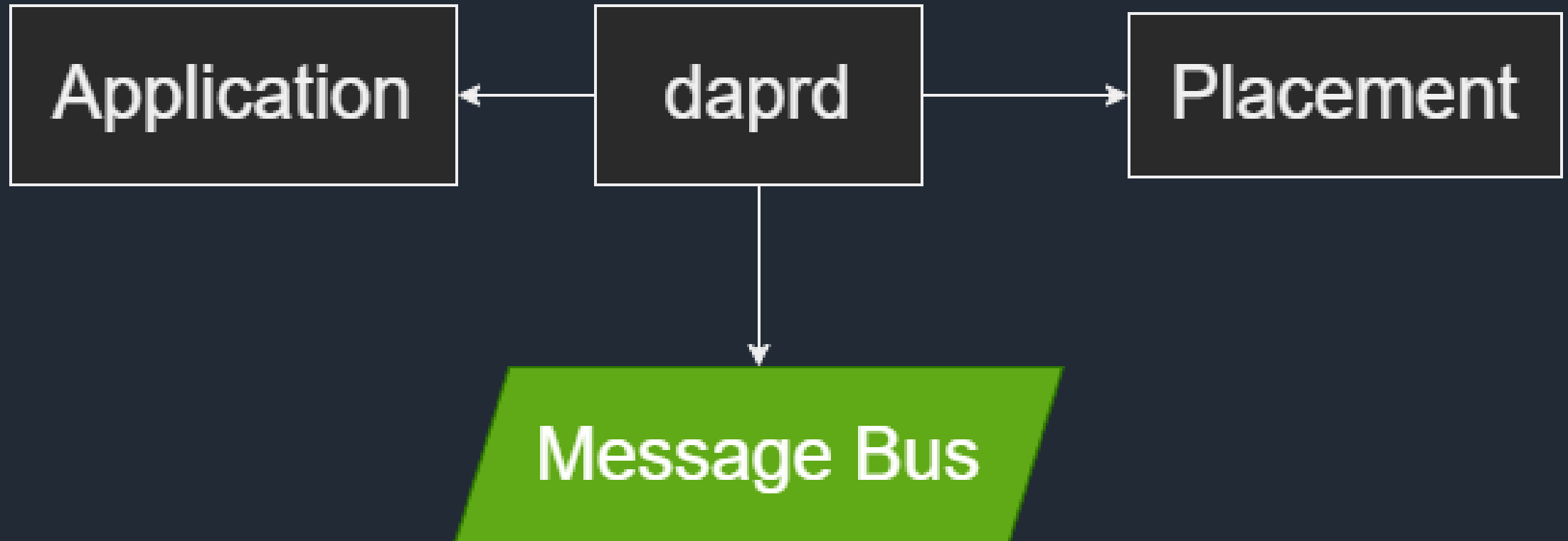
Неучтенные требования

- Необходимо хранить состояние. Например расчеты нарастающим итогом.



Неучтенные требования

- Получение/отправка данных через шину



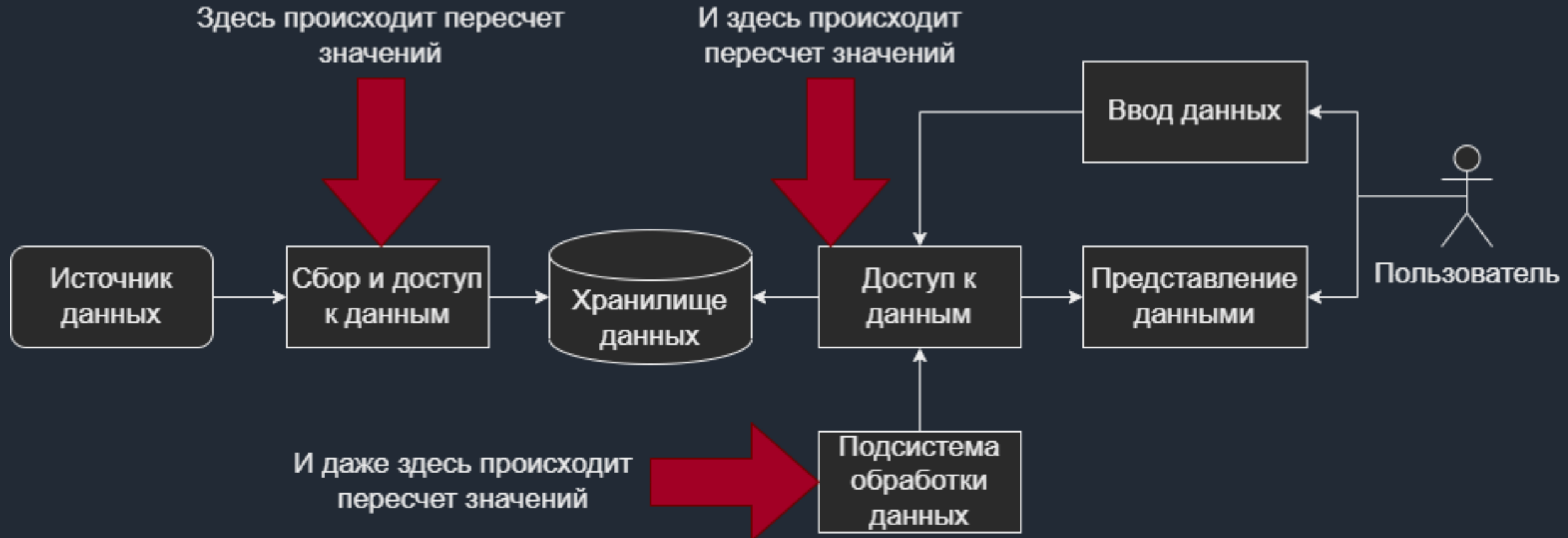
Еще пара моментов

- Подготовлена хорошая документация
- Поддерживается сильным вендором (Microsoft)

Итоговое решение



Вернемся к началу?



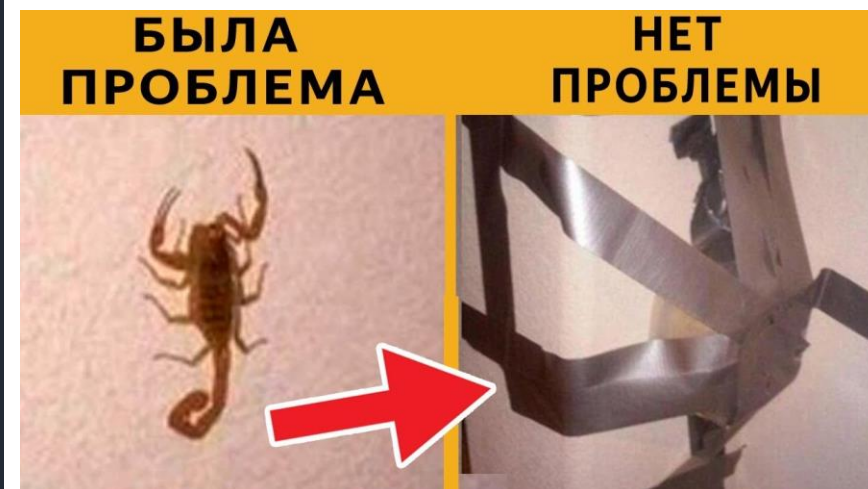
Из чего состоит решение?



Где же DAPR?



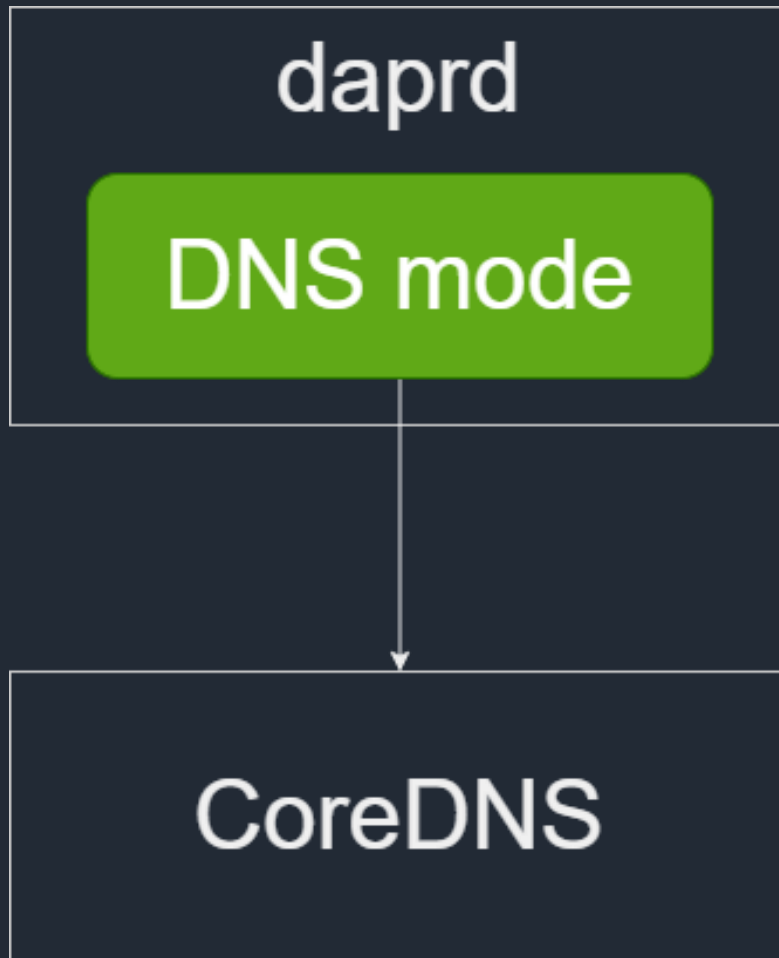
Проблемы и решения



Развертывание в Kubernetes



Развертывание в Kubernetes. Костыль



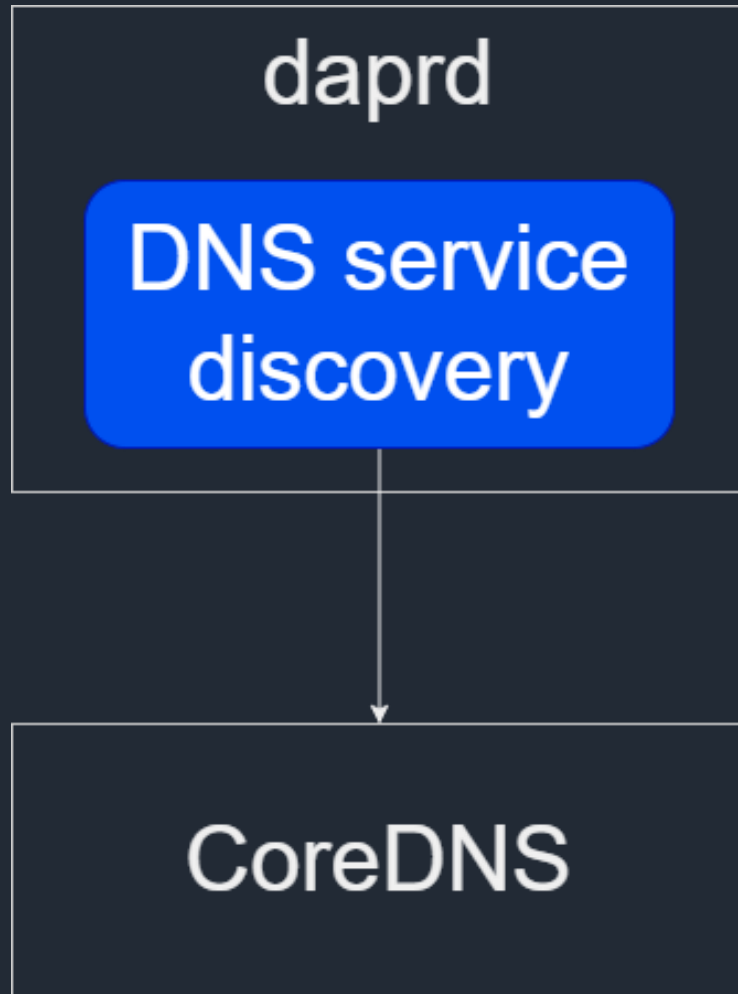
Быстренько запилим свое решение и предложим его сообществу.

Сделали 2 PR

<https://github.com/dapr/dapr/pull/2903> и

<https://github.com/dapr/components-contrib/pull/858>

Развертывание в Kubernetes. Решение



А вот правильное решение

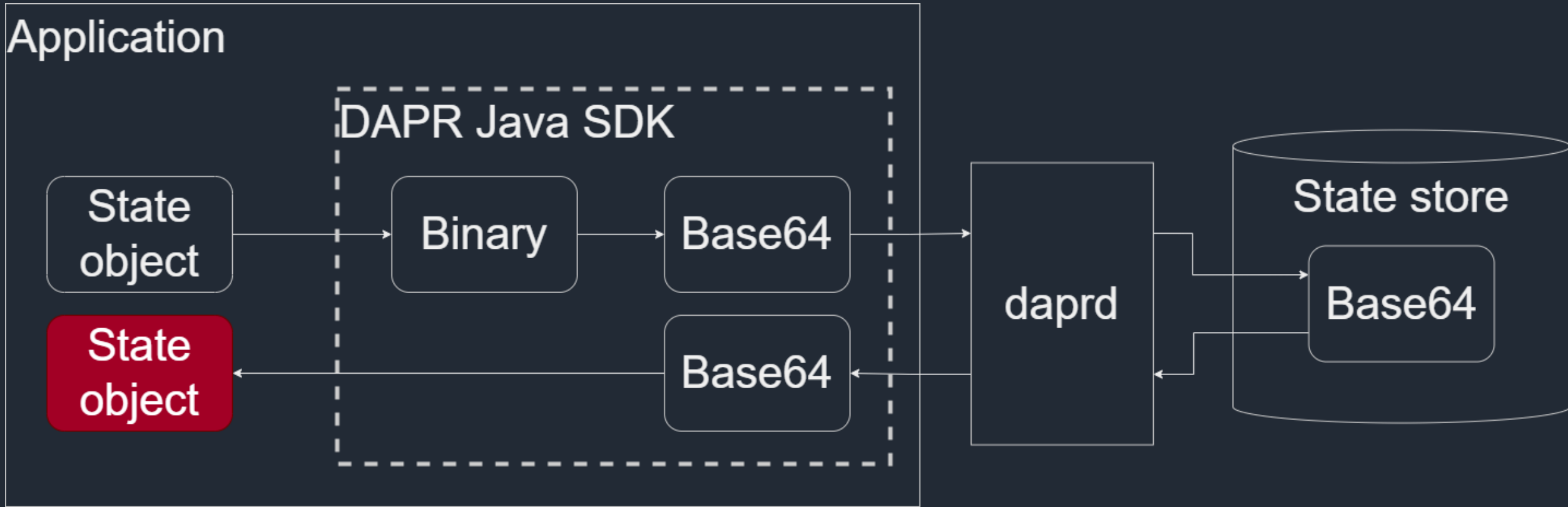
<https://github.com/dapr/dapr/pull/2904>

04 жаль не наше (

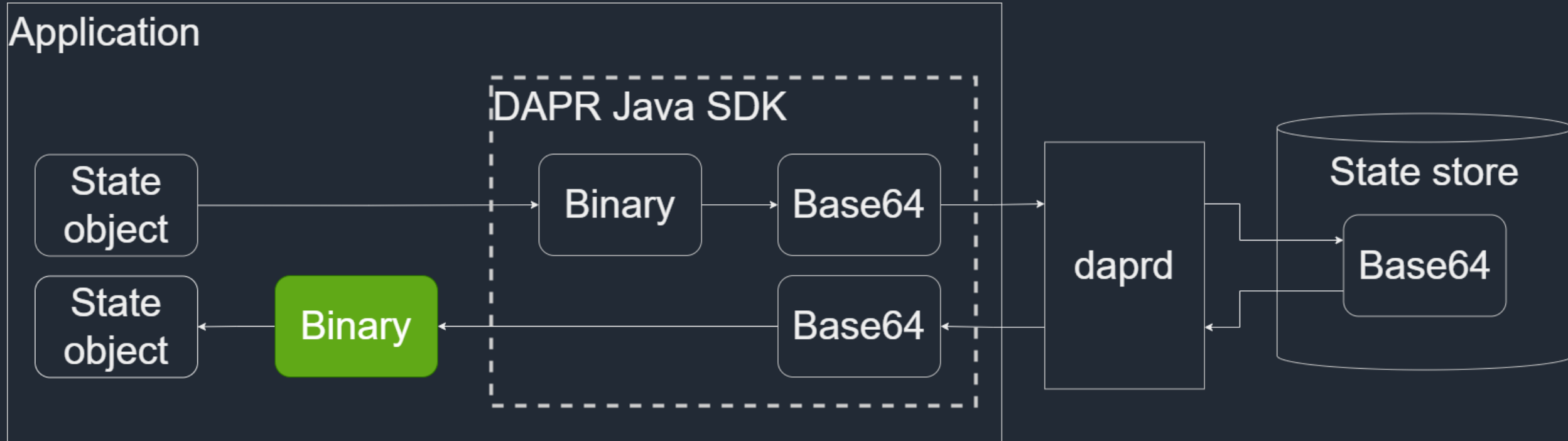
Десериализация сохраненного состояния



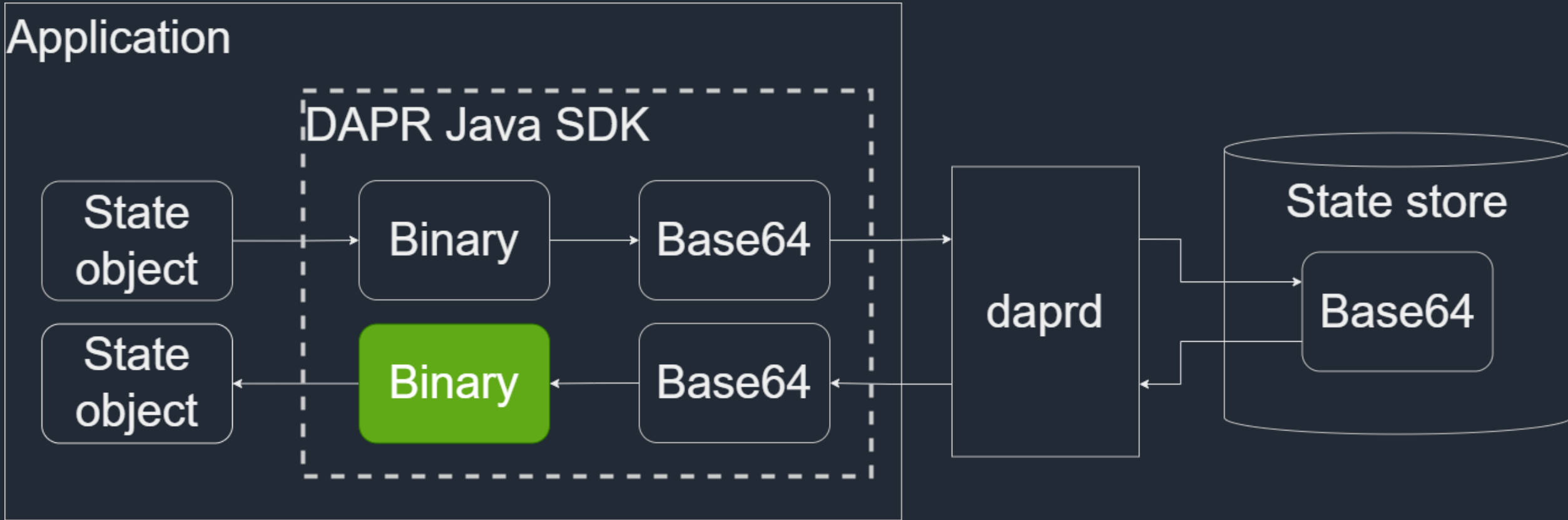
Десериализация. Проблема



Десериализация. Костыль



Десериализация. Решение

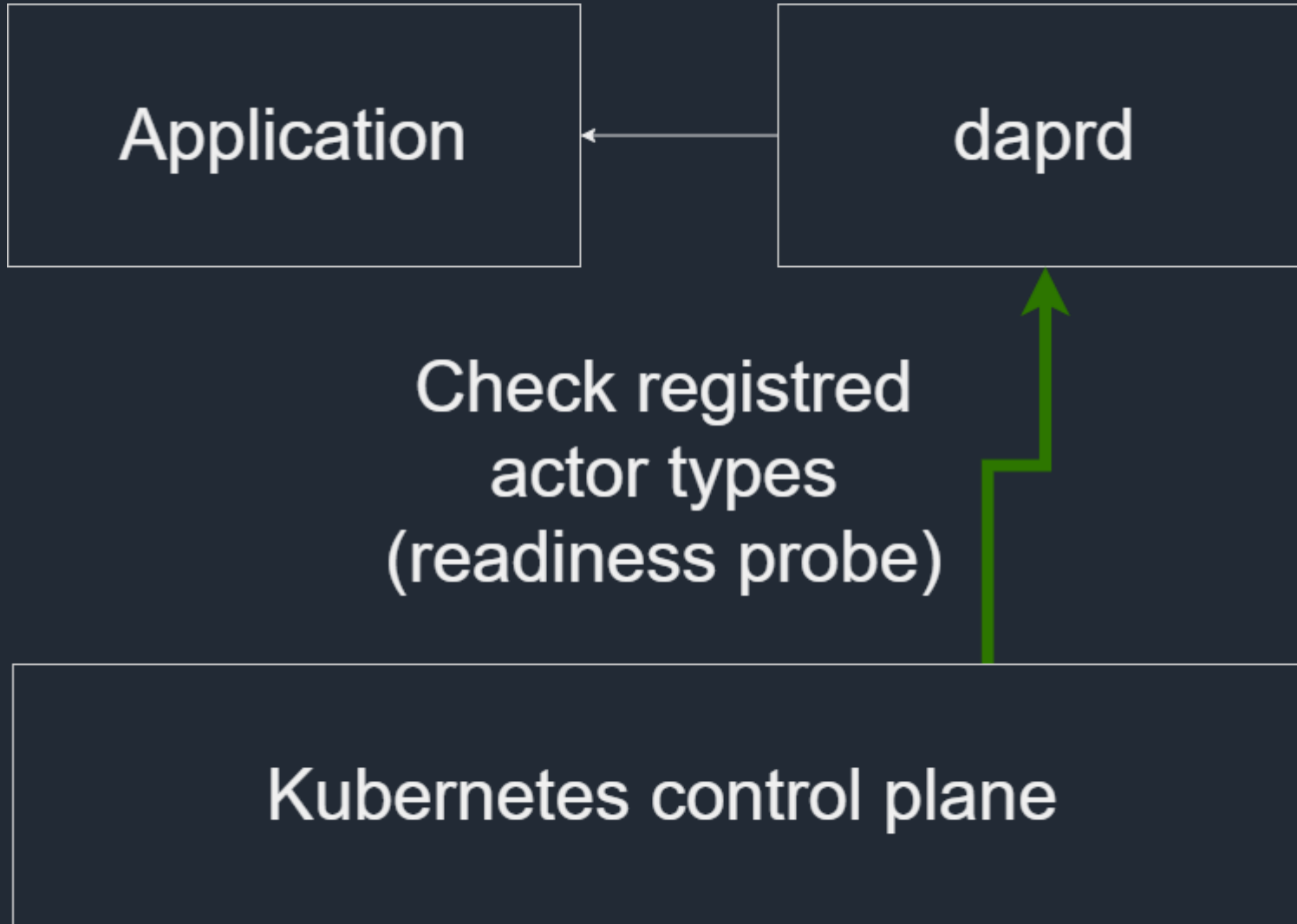


<https://github.com/dapr/java-sdk/pull/539>

Регистрация типов Акторов

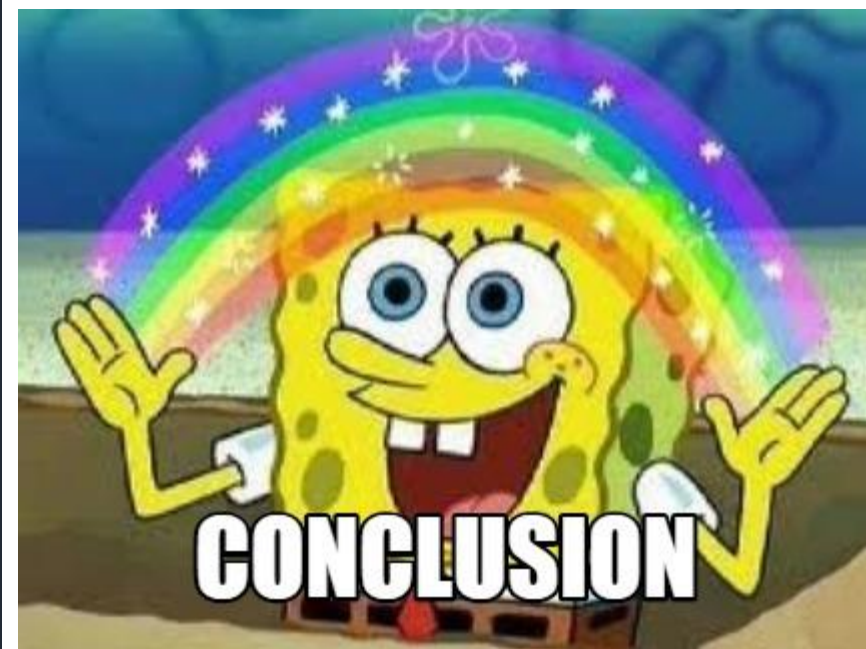


Регистрация типов Акторов. Решение



Как водится сделали «костыль» и завели issue
<https://github.com/dapr/dapr/issues/3132>

Выводы, куда же без них



Что мы поняли

Уделите достаточно времени поиску
готового решения вместо написания
очередного велосипеда

В нашем случае мы сократили разработку примерно на
6 месяцев для команды из 3 человек таким образом
сэкономив 18 человекомесяцев или 1,5 человекогода

Что мы поняли

Смотрите на задачу под разными
углами

В нашем случае пересмотр модели с **FaaS** на
Virtual actors помог нам избежать построения
костылей поверх FaaS

Что мы поняли

Популярность решения не самый
важный критерий

Если есть решение которое хорошо подходит для вашей задачи, но о нем не пишут в каждой второй статье это вовсе не значит что оно неприменимо.

Интересные ссылки

<https://github.com/dapr/dapr> - основной репозиторий.

<https://github.com/dapr/components-contrib> - репозиторий где лежат все компоненты

<https://dapr.io/> - сайт

<https://docs.dapr.io/> - документация

<https://discord.com/invite/ptHhX6jc34> - Канал в Discord, где можно задать интересующие вопросы (осторожно английский)

<https://github.com/man-group/dapr-sidekick-dotnet> - Проект sidekick позволяет запускать без sidecar. Развивается энтузиастами и только для .NET



Спасибо за внимание!



Богданов Андрей Сергеевич

andrei.bogdanov@vk.team / +7(920)3507781

Старший архитектор/  VK Cloud

