



Alexey Shcherbak

“Boring Stack”

- .NET 4.6, C#
- Cloud Service @ Azure
- RESTful WebAPI with Swagger
- Orleans

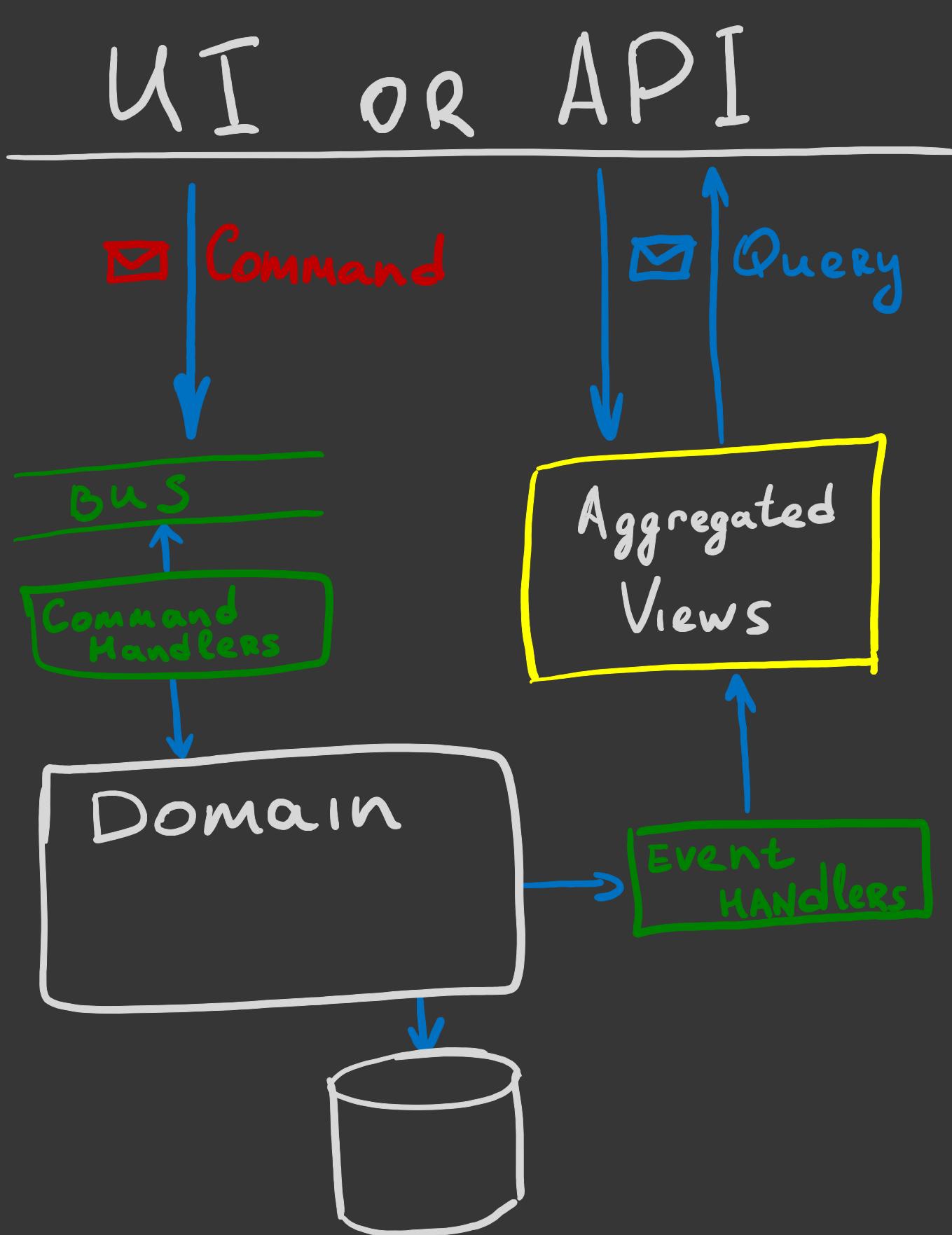
Service Fabric

- Closed by default
- Private docs
- Closed community

Orleans

- Open by default
- Detailed docs
- Thriving community

CQRS & Event Sourcing



• Refactoring-to-CQRS
is Painful

START
WITH
ES+CQRS

Uniform Contract (API)

```
interface ICounter
{
    Task Handle(CountCmd cmd)
}

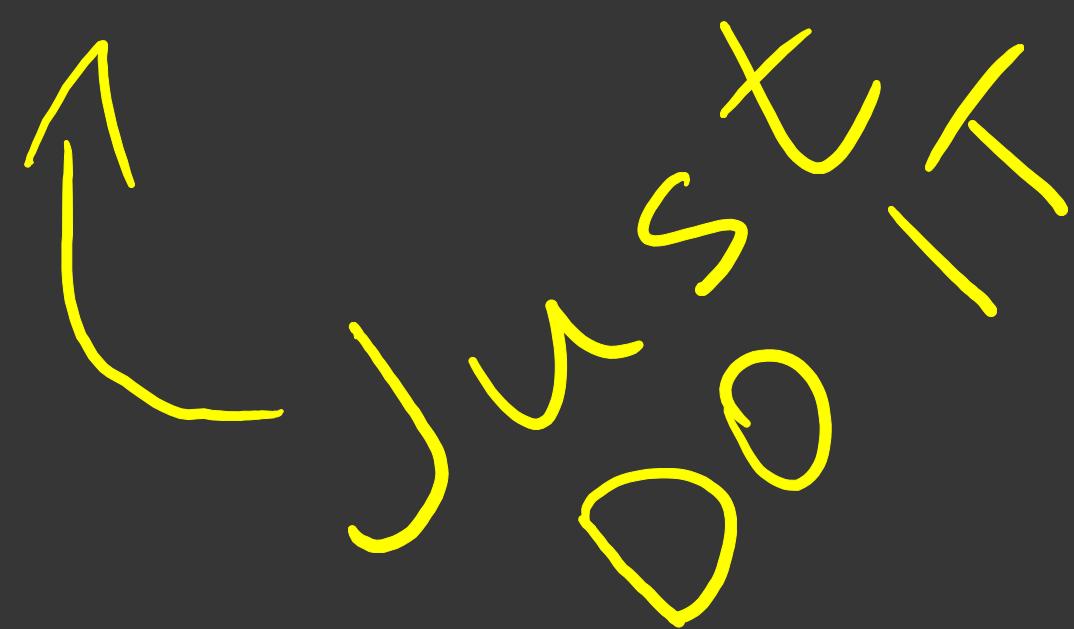
class CountingGrain : ICounter
{
    public Task Handle(CountCmd cmd){
        return performCommand(cmd as dynamic);
    }

    async Task performCommand(AddCmd cmd){
        // Addition
    }

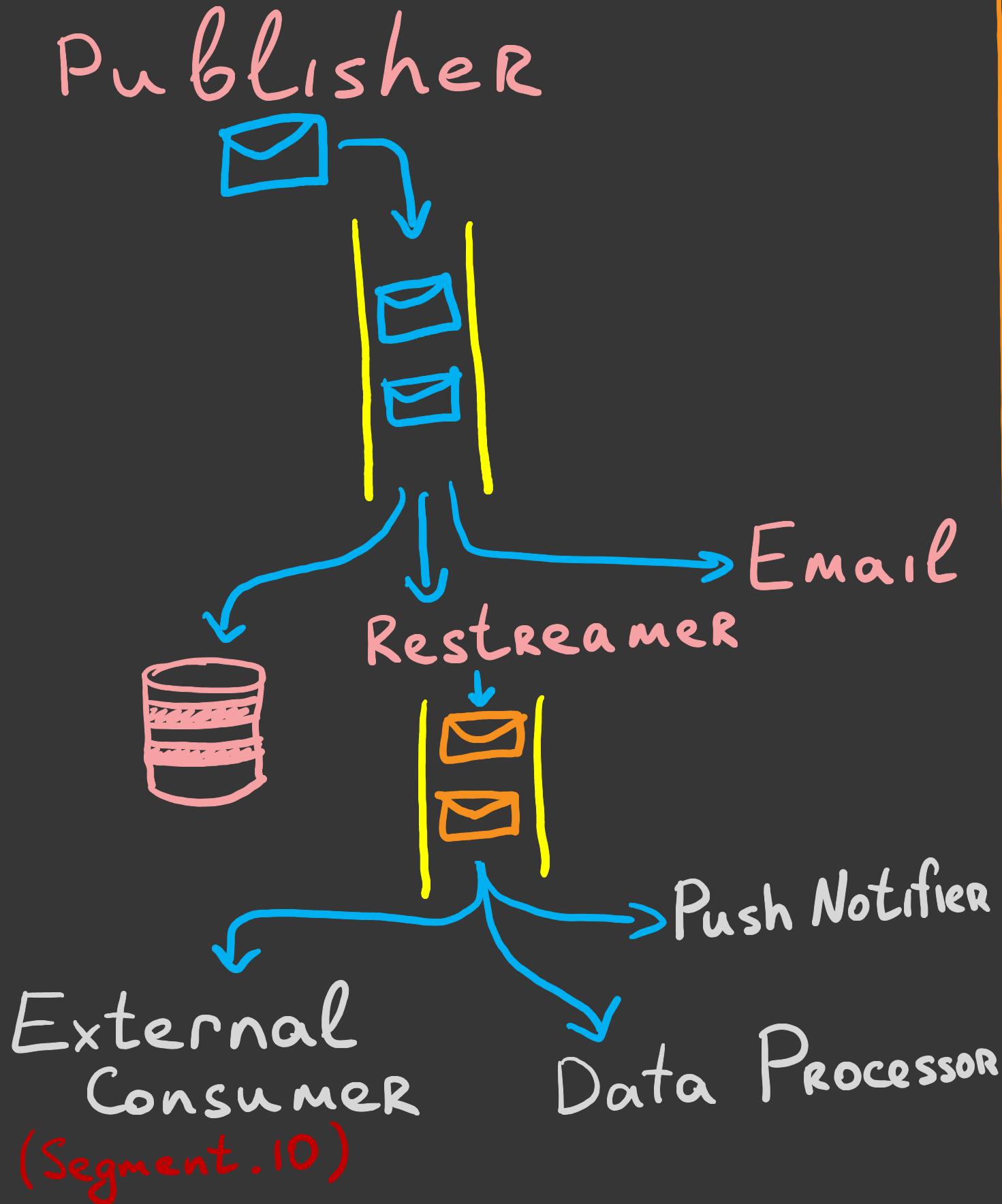
    async Task performCommand(SubtractCmd cmd){
        // Subtraction
    }
}

abstract class CountCmd { }
class AddCmd : CountCmd { public int Value; }
```

- Great even w/out ES
- Helps with cross-cutting code
- Type-matching in C#
- Data-driven tests



Orleans Streams

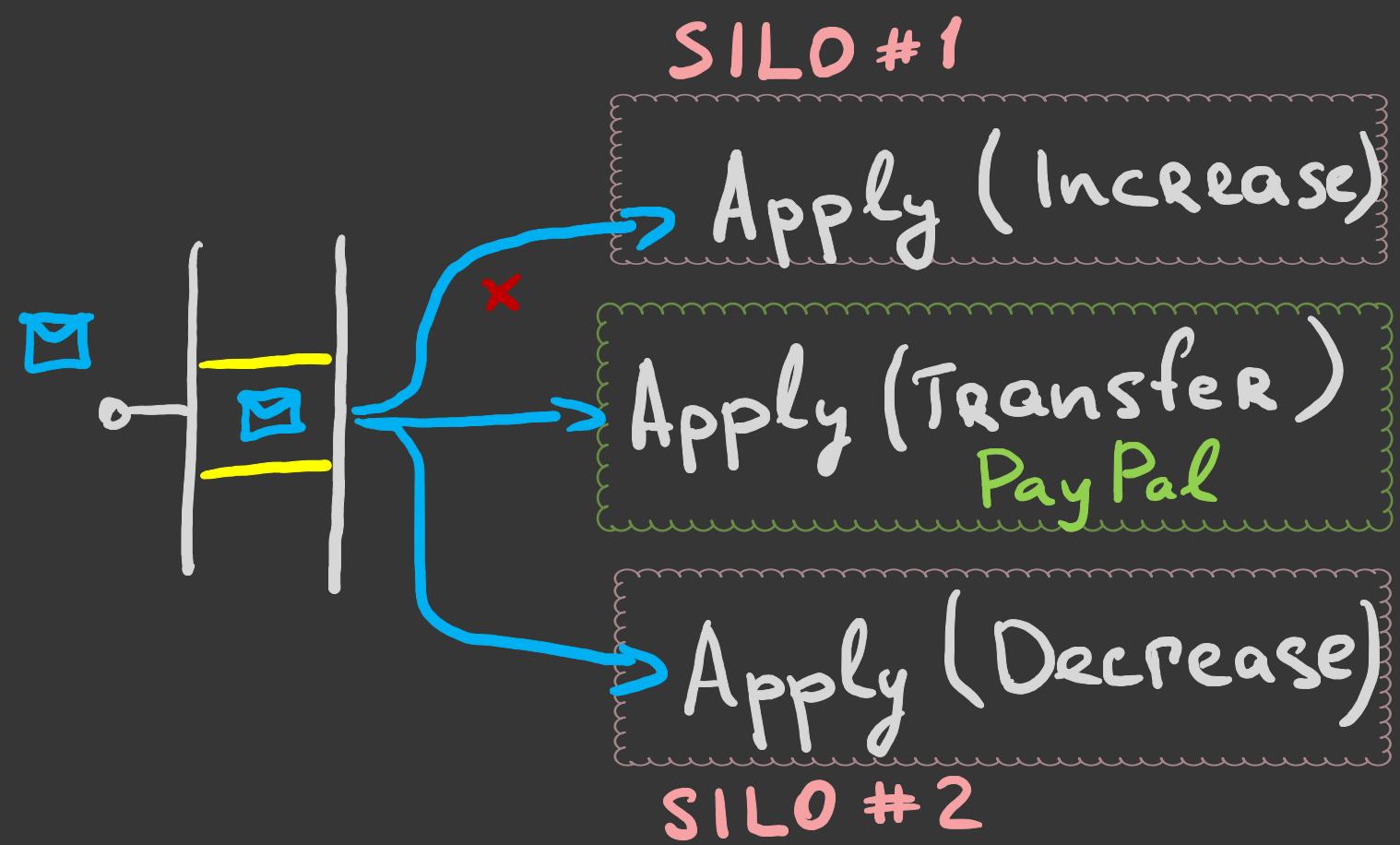


- PubSub model for code
 - Client side subscription + Websockets 
 - Mapping & filtering instead of grouped method calls
- HOOKED ON THE FIRST TRY

chALLENGEs

~~ROADBLOCKERS~~

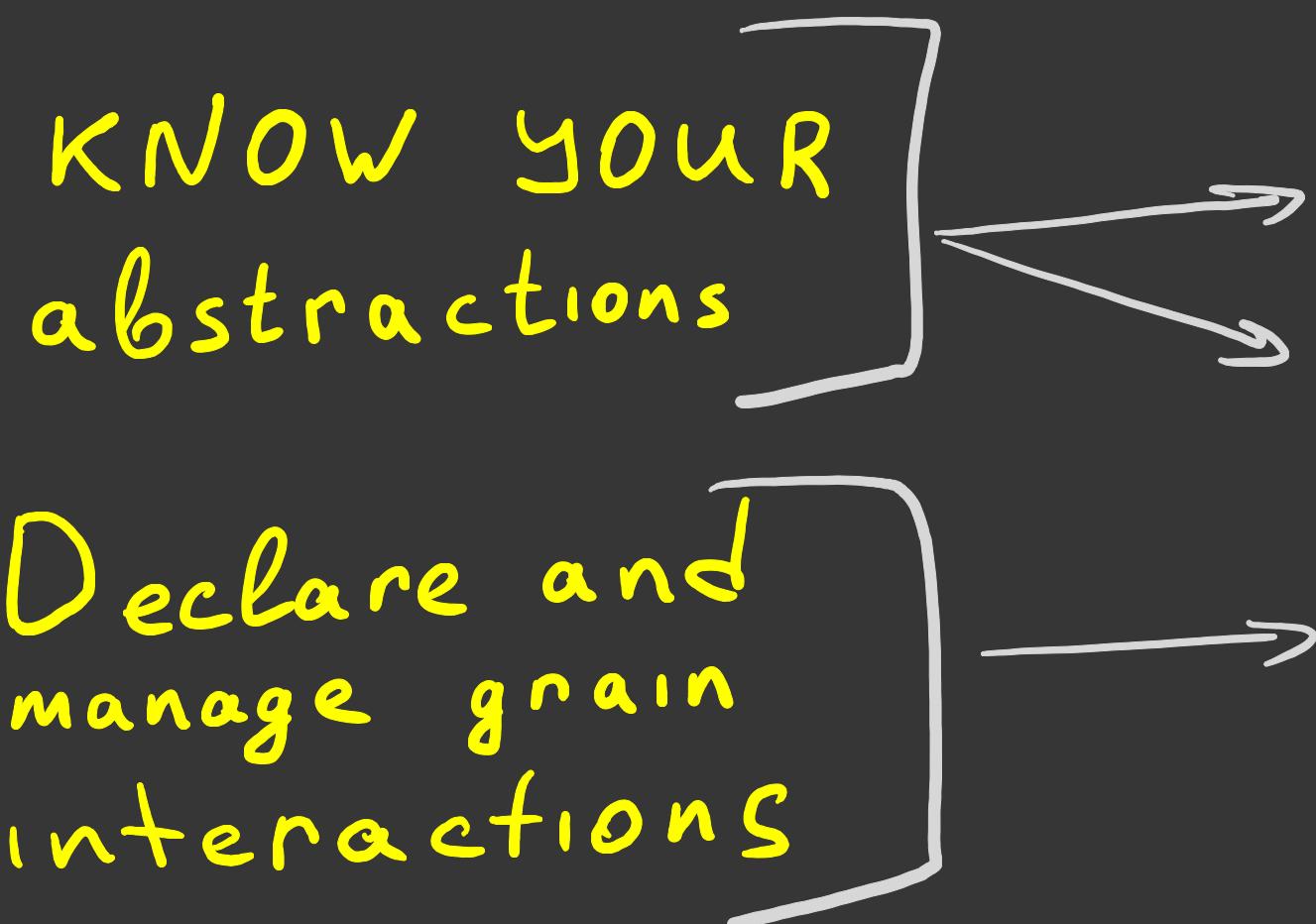
Business Logic Transaction



- Typical problem for distributed systems
- Sagas != Transactions
- Multi Actors changes
- General purpose
SOLUTION

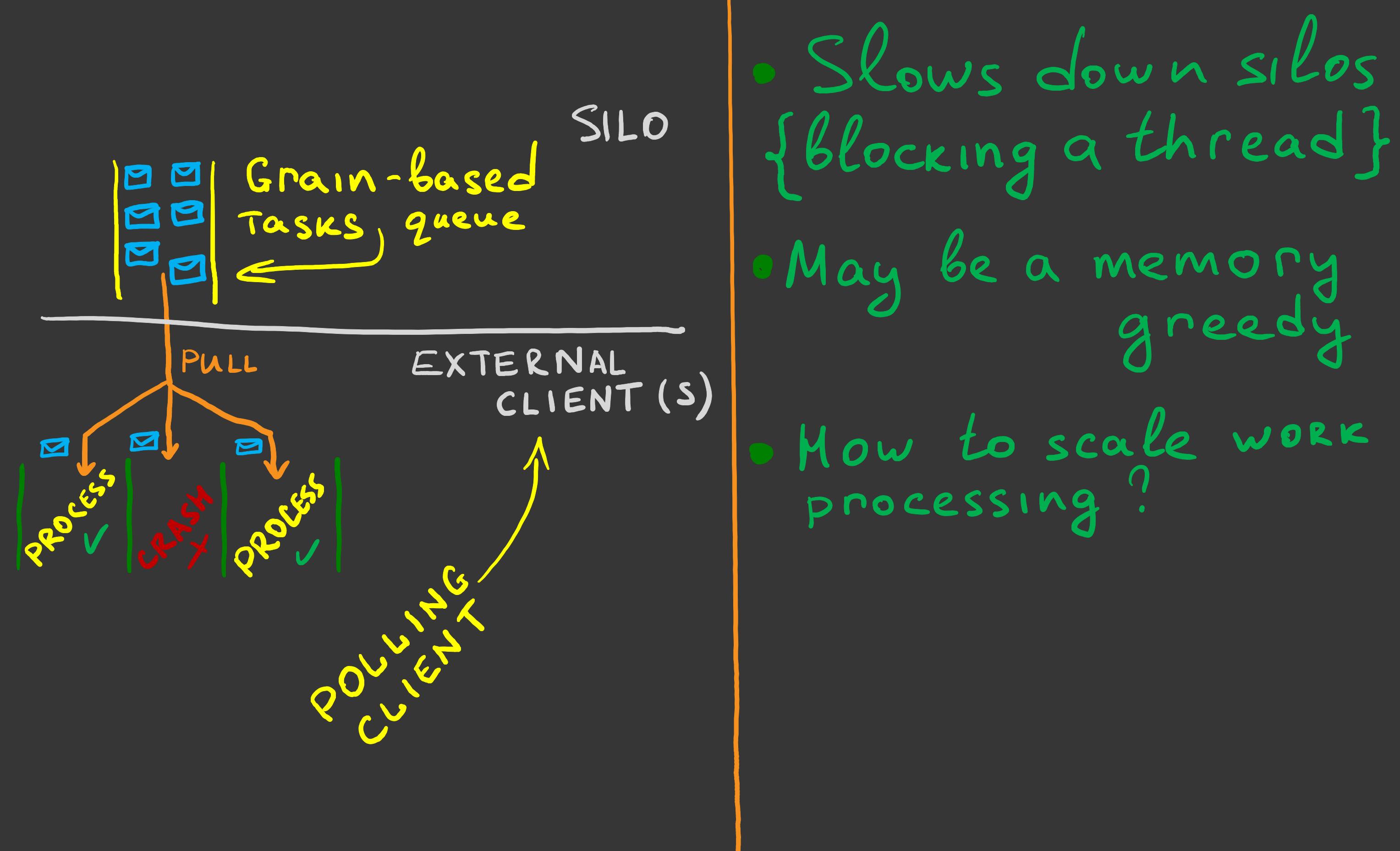
???

LEAKY ABSTRACTIONS



- Implementation-specific:
 - Limitations
 - Behaviours
- Grain Factory = Service Locator !!

Long running tasks



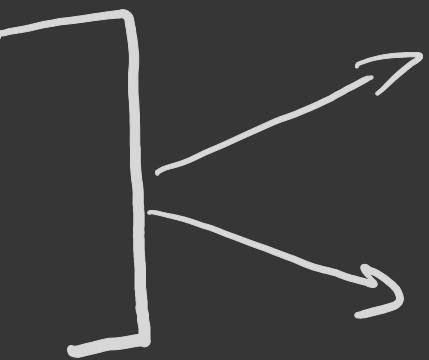
BE AWARE

ALWAYS RUN
A CLUSTER

EXPECT IT
FROM START

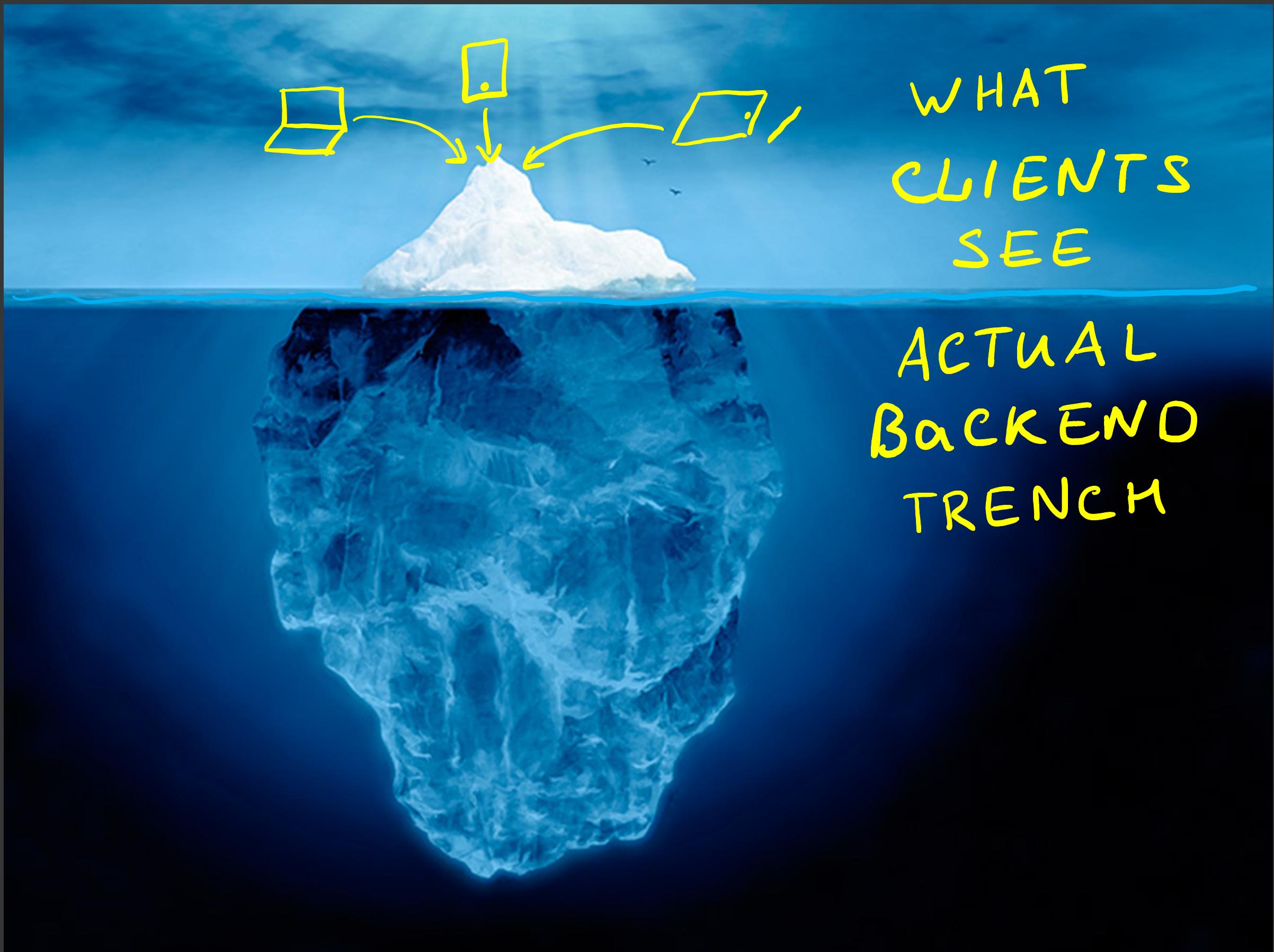
NO HARM IN
ASKING OTHERS

gitter.im/dotnet/orleans



- Serialization
- Delivery failures
- Out-of-order messages
- Other distributed system pitfalls

Powering our clients



Q

&&

A

THANK you :)

come to our workshop
and learn