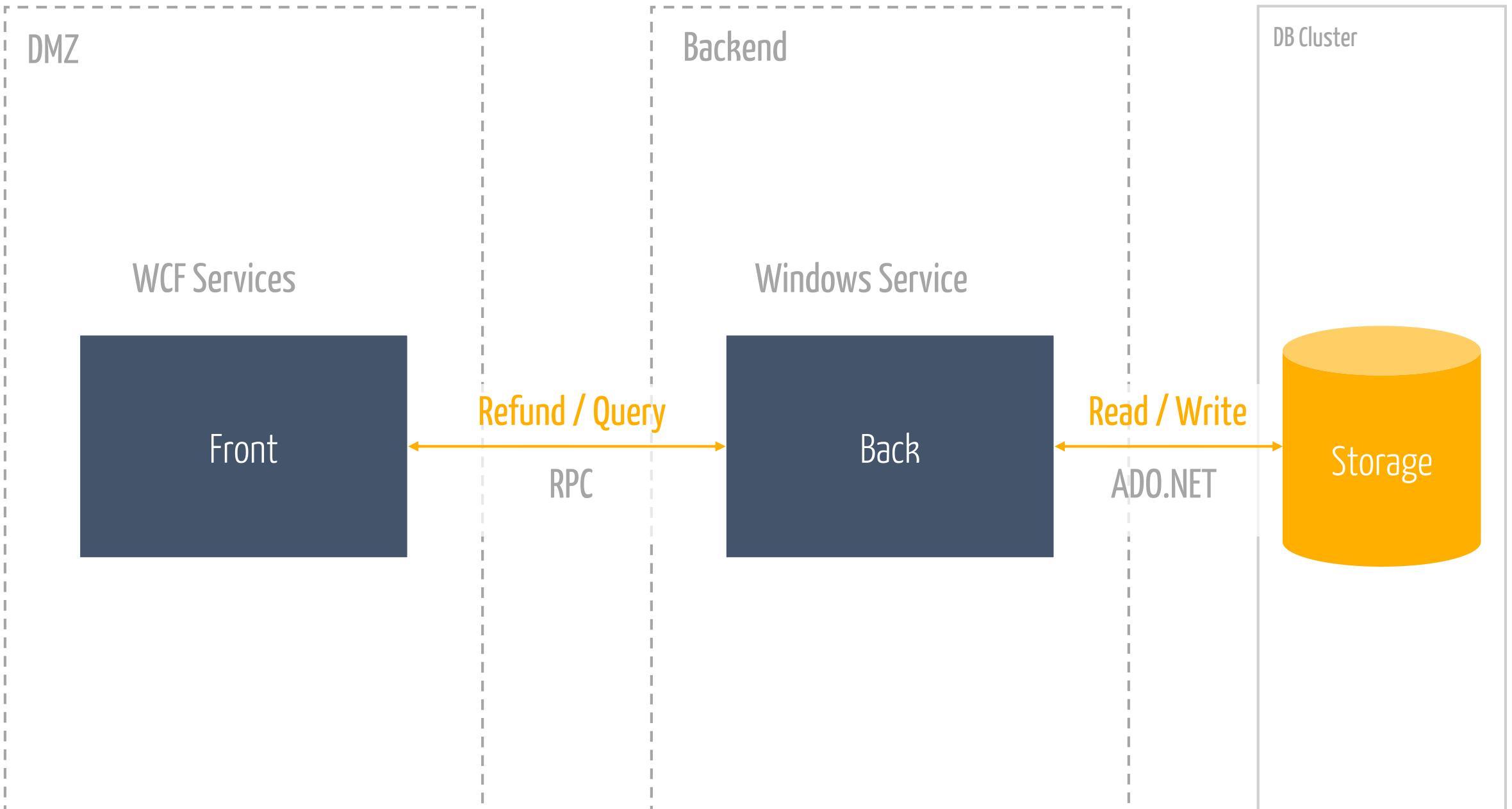




DIY Async Message Pump

Lessons from the trenches

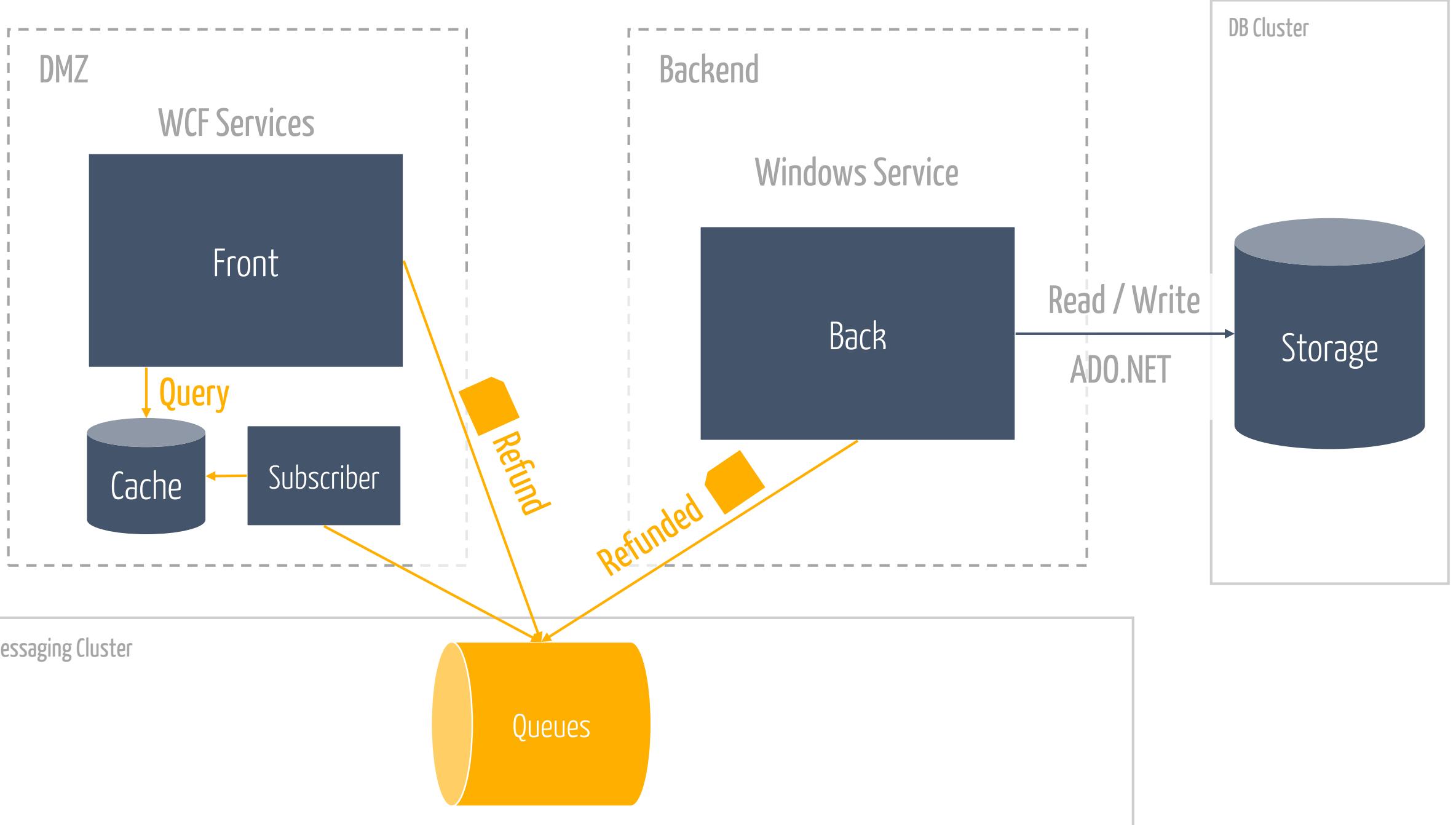


The RPC callstack of

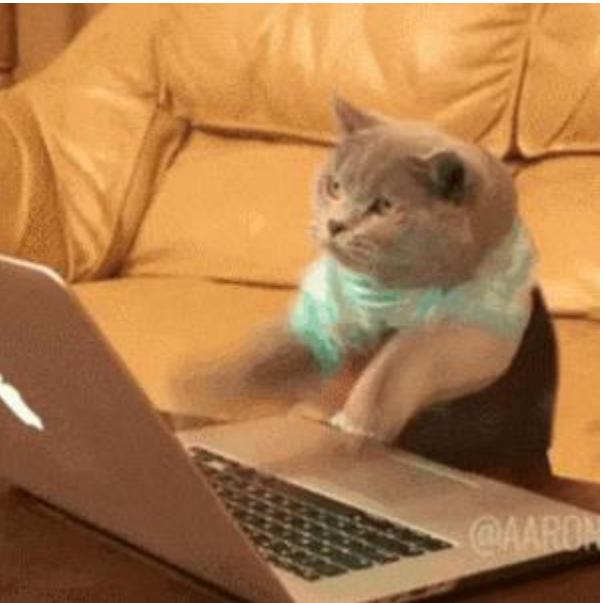
doom

praise the lords of

messaging



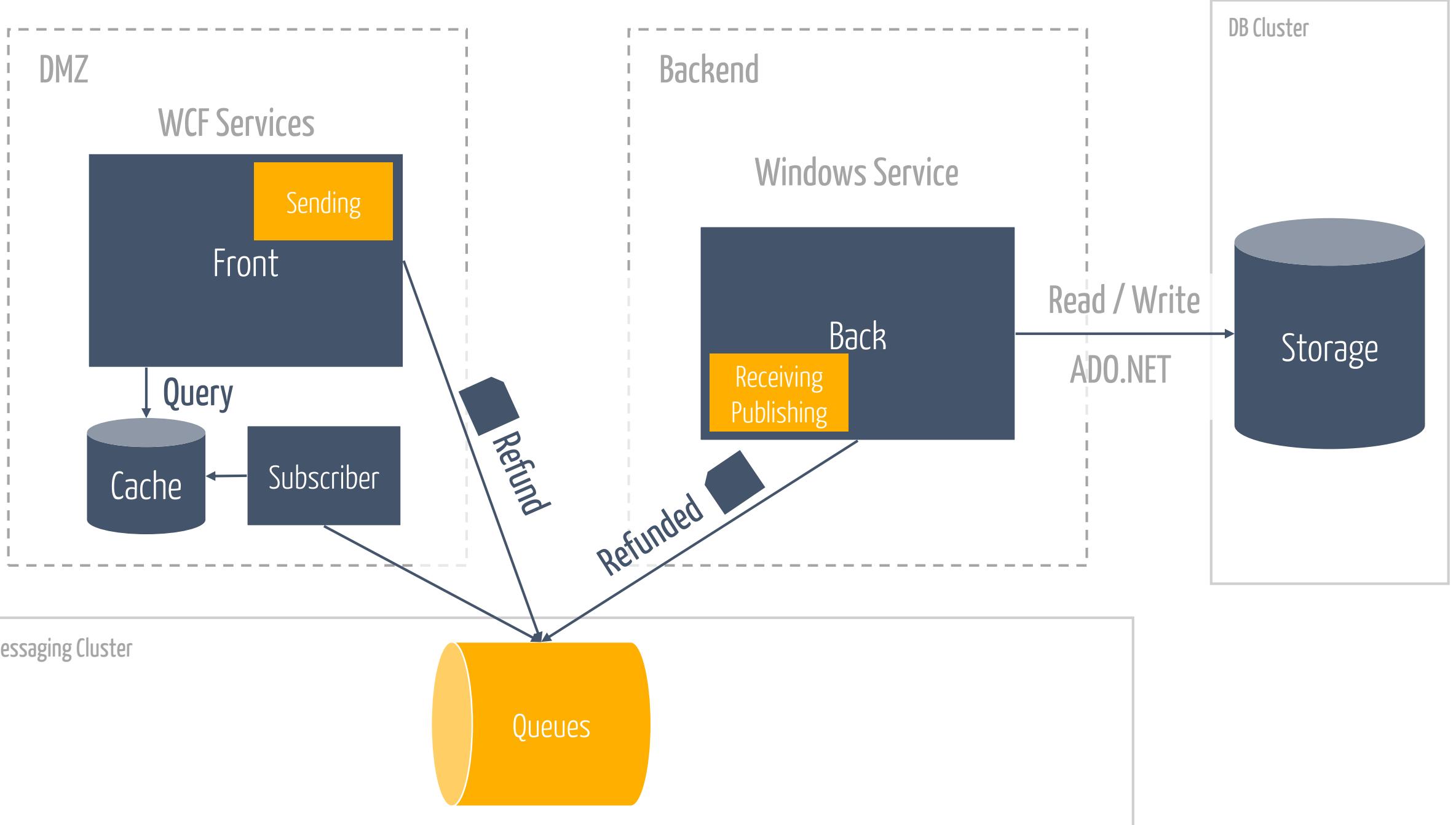
Build



or



Buy



Building the

pump



TPL handwaving

Cooporative cancellation 101

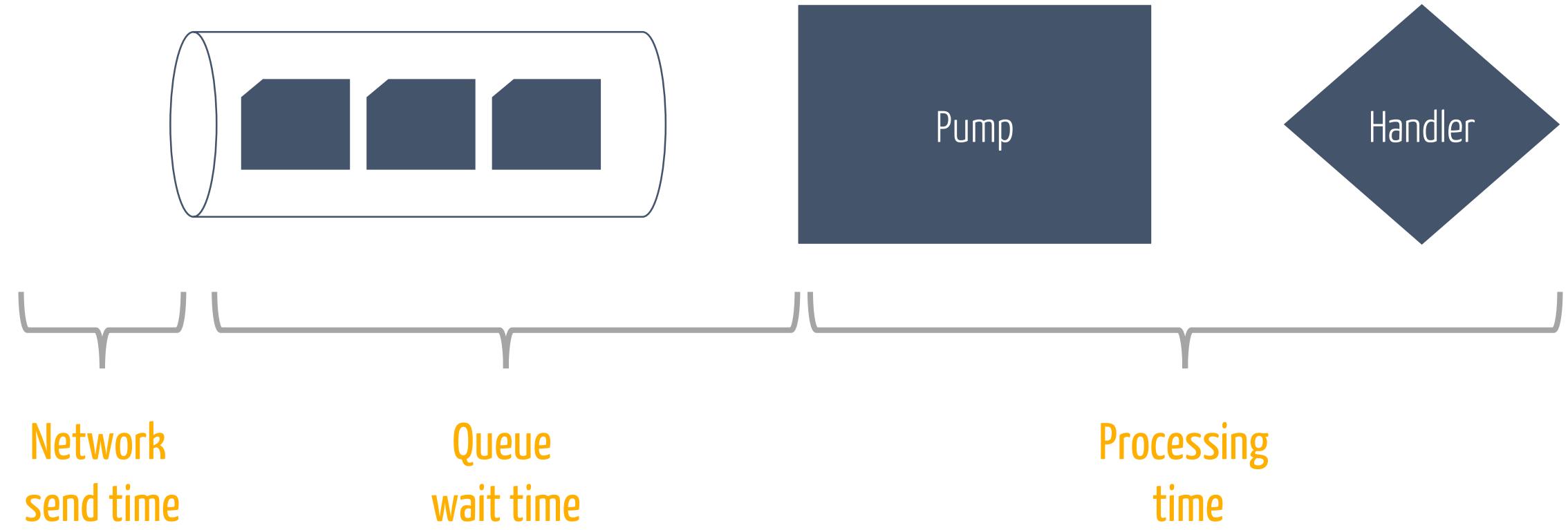
Async / Await

Ship it!

It worked until



Rush hour



Let's throw in some

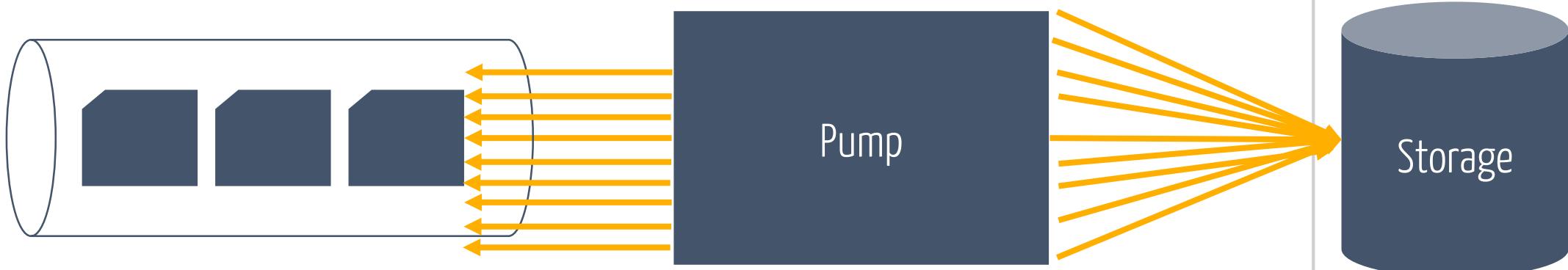
concurrency

Just a tiny change...
Introduce Fire & Forget
Ship it!

It worked until



Rush hour



DB Cluster

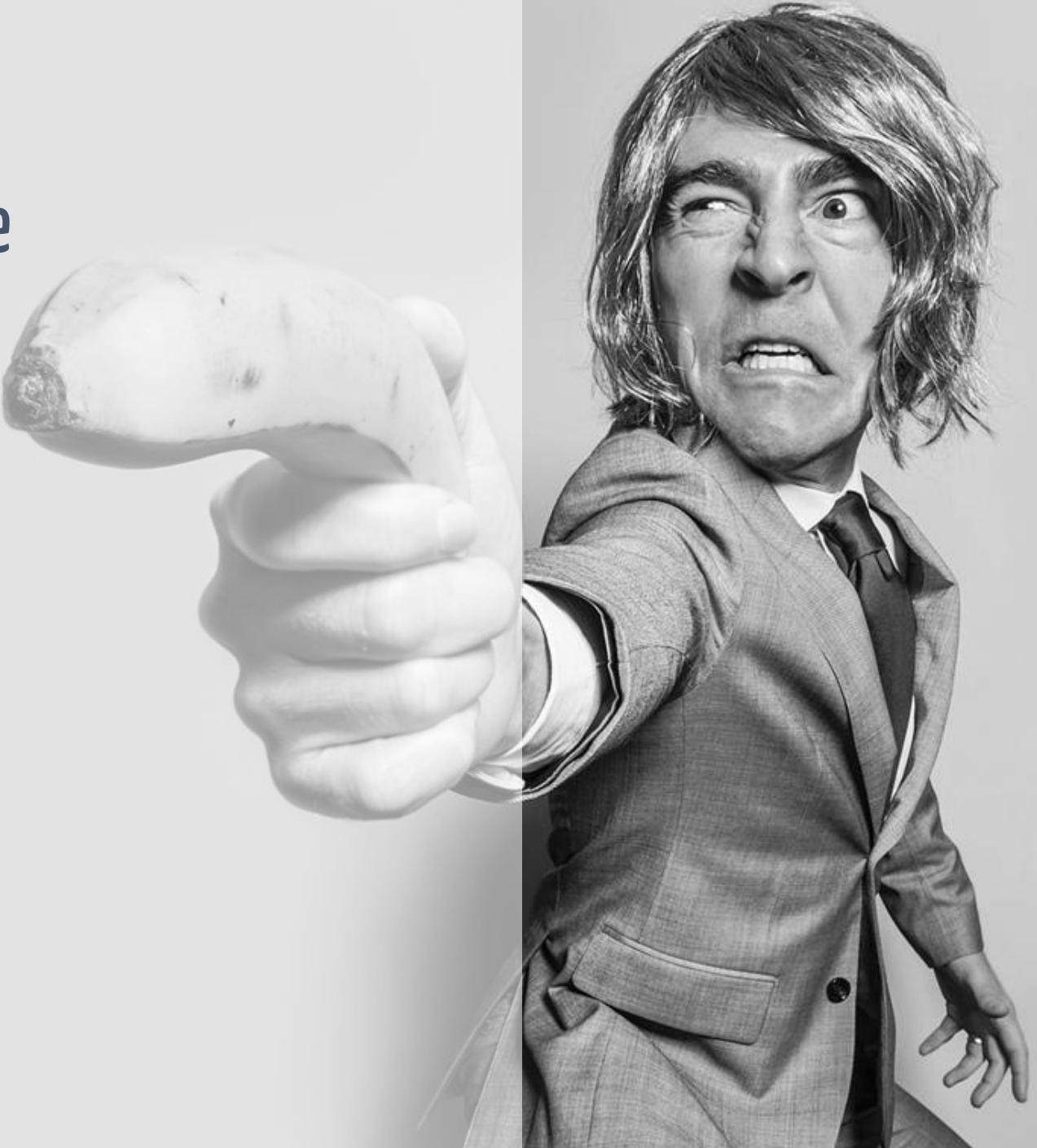
Storage

Better limit

concurrency

Semaphore controls floodgate
Ship it!

It worked





As time passes

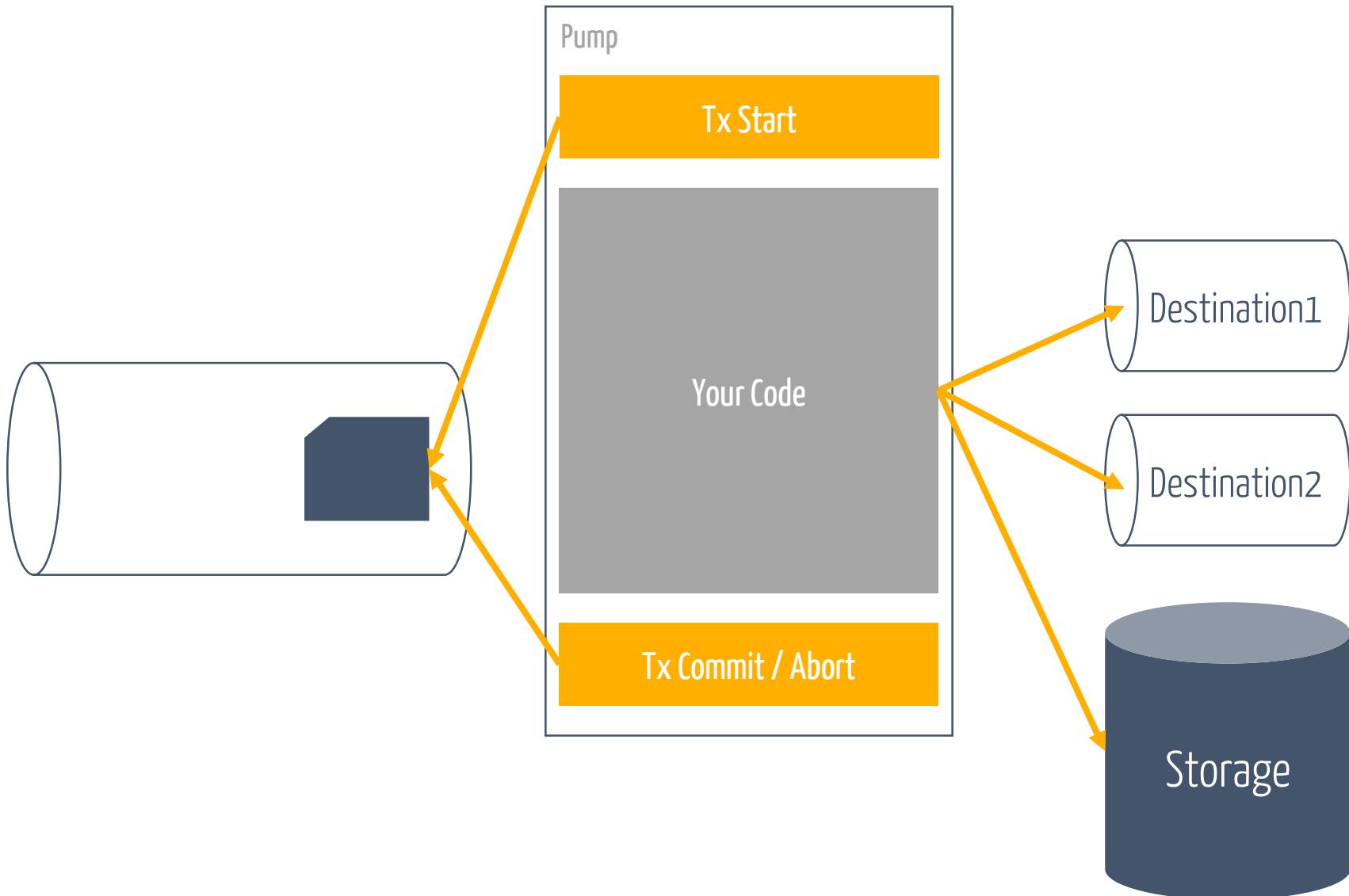
Make it

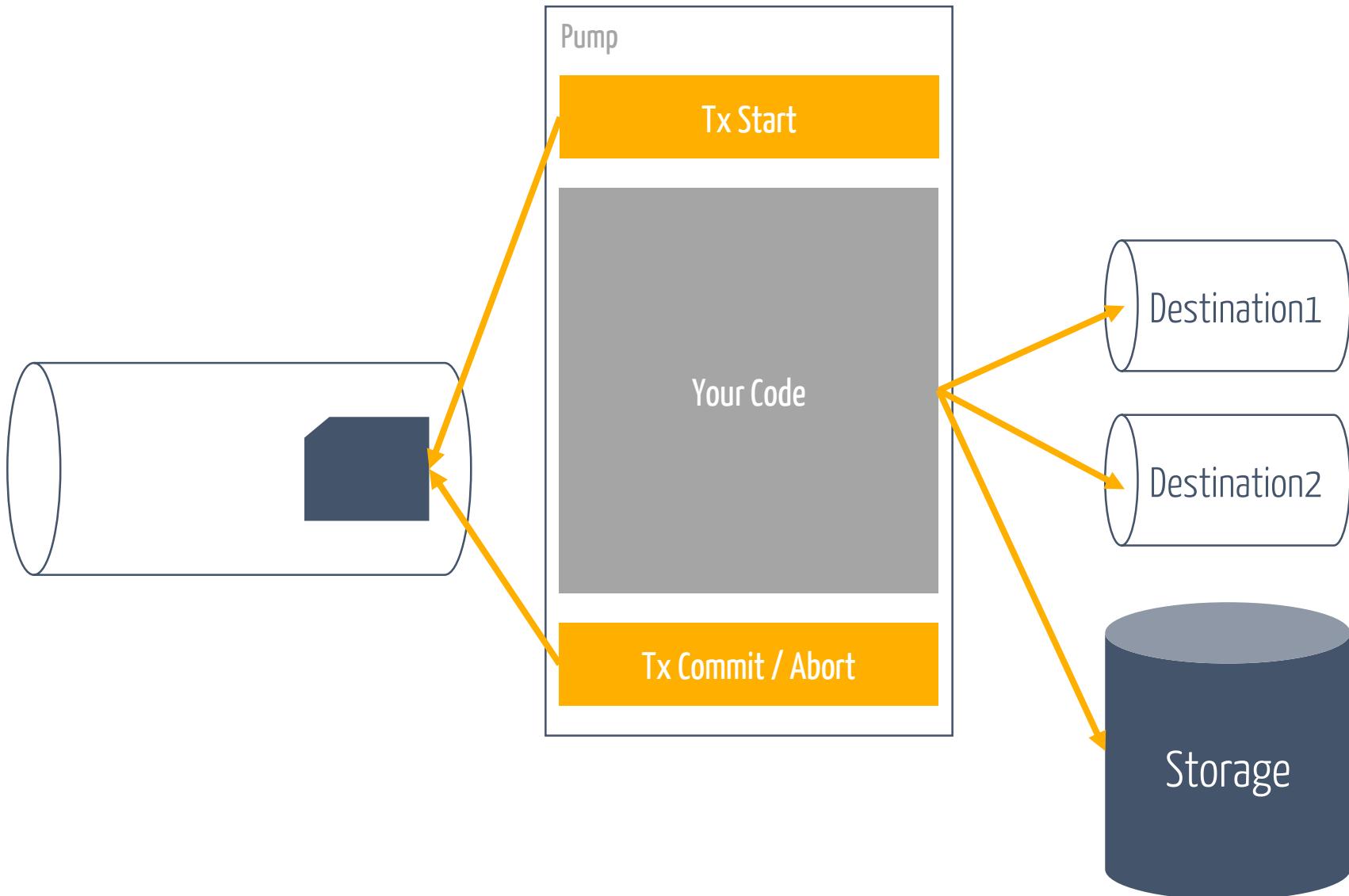
flextensible

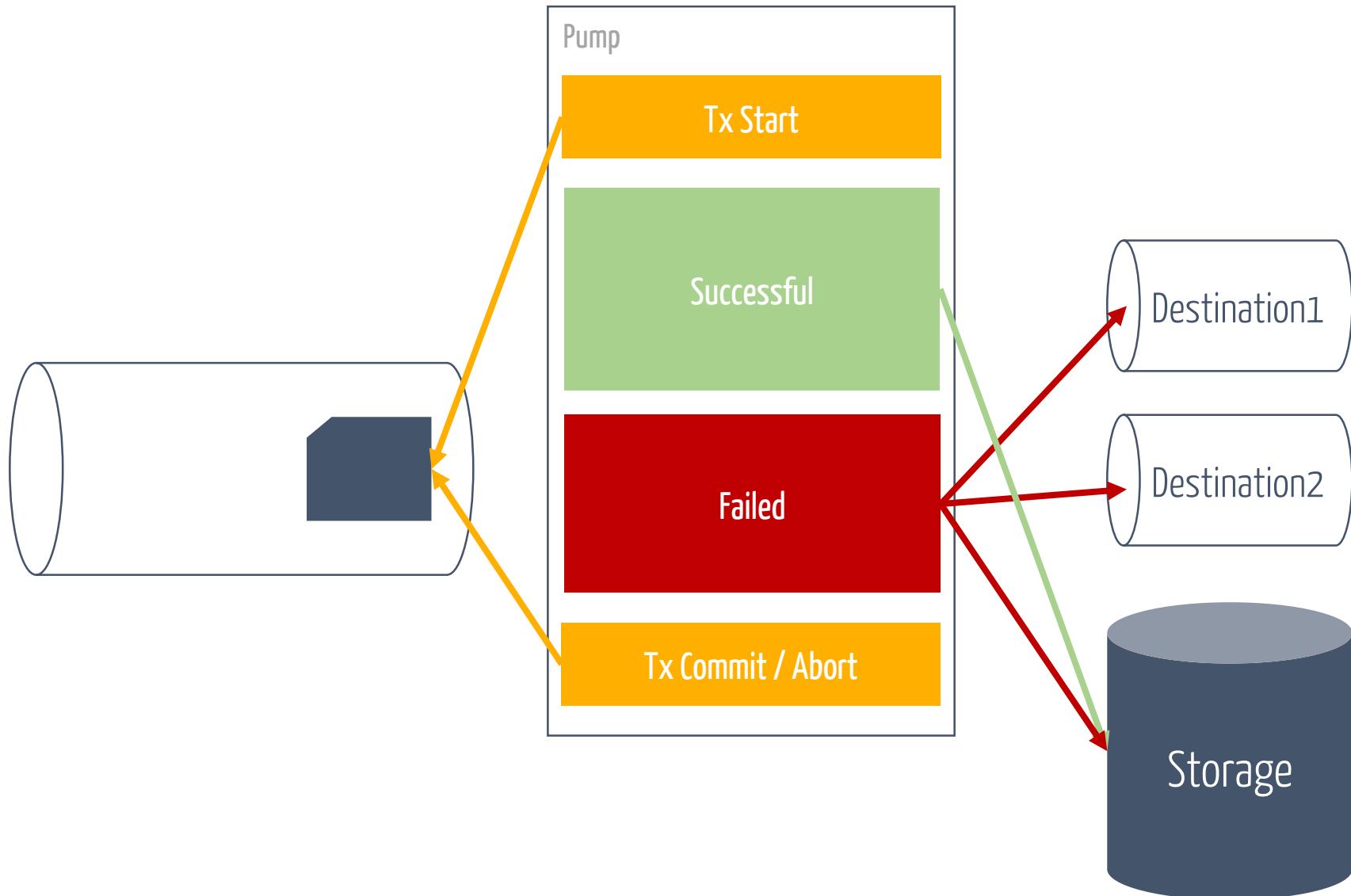


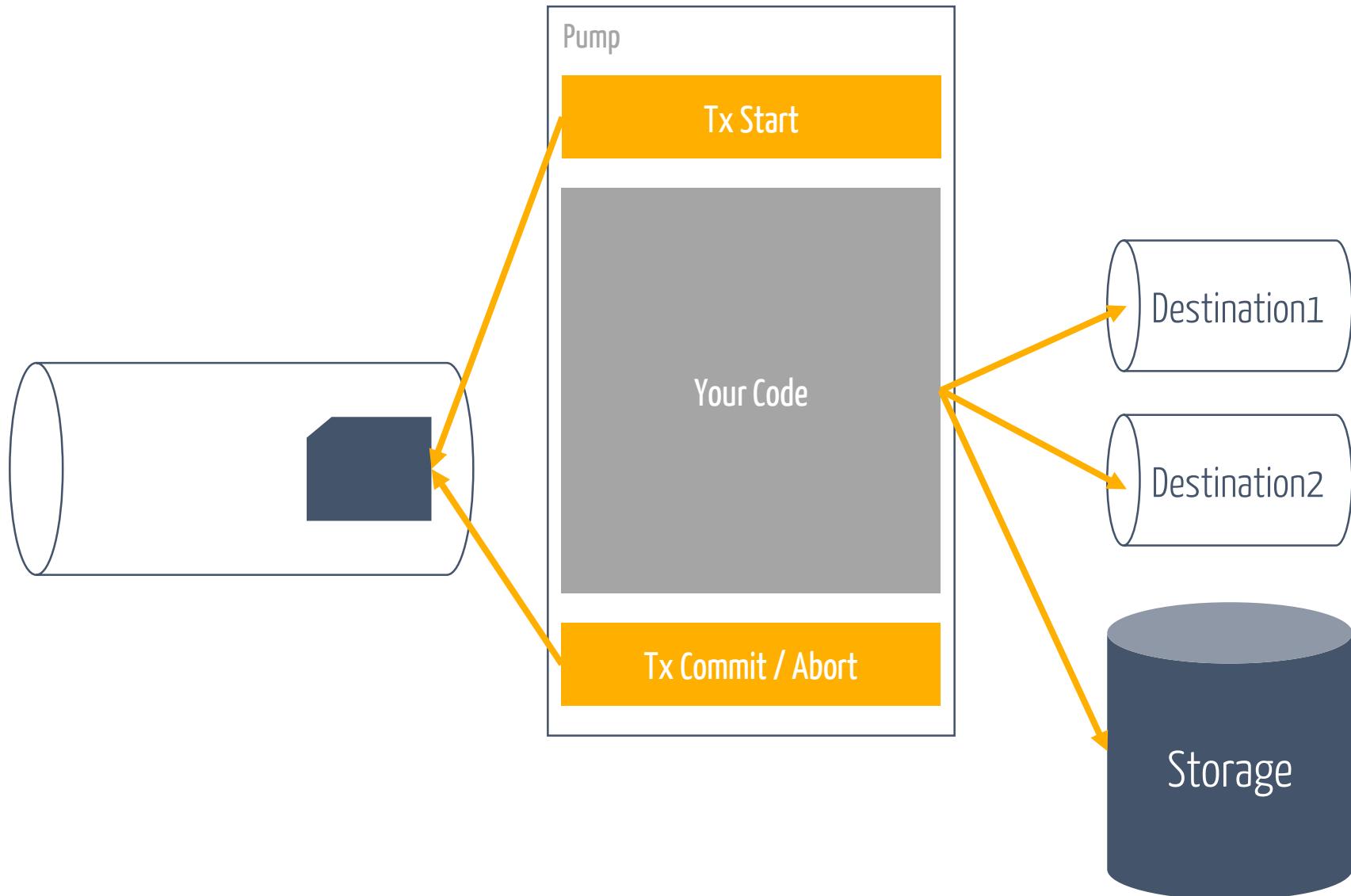
Life beyond

transactions



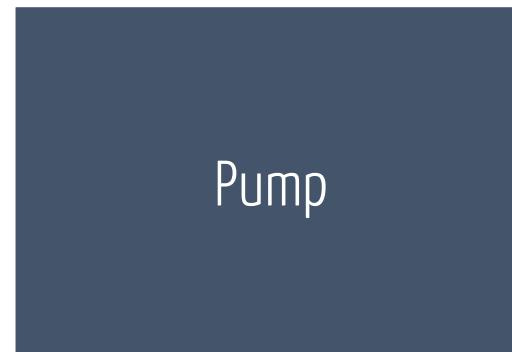


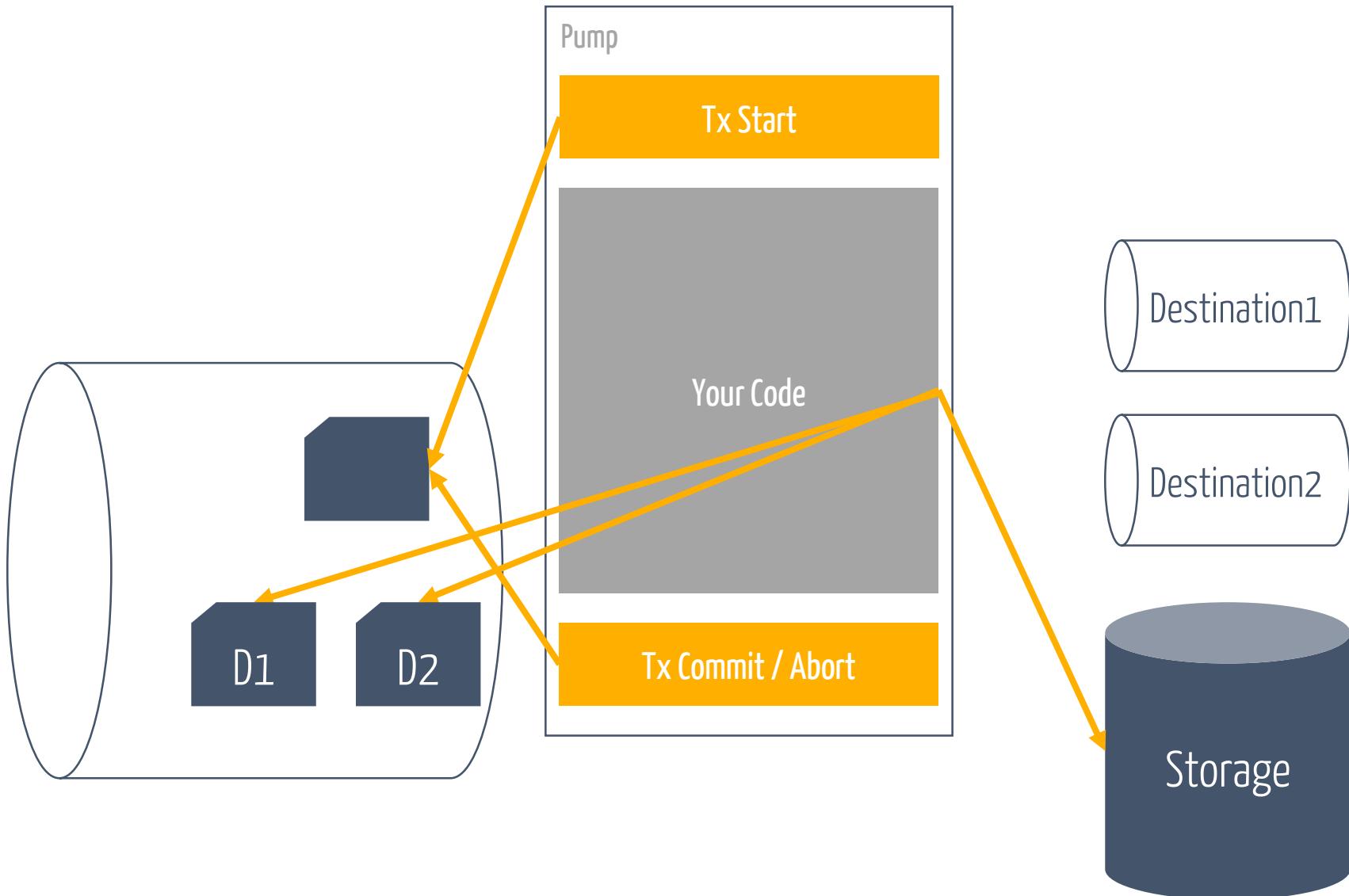




Cloudy with a chance of

failura



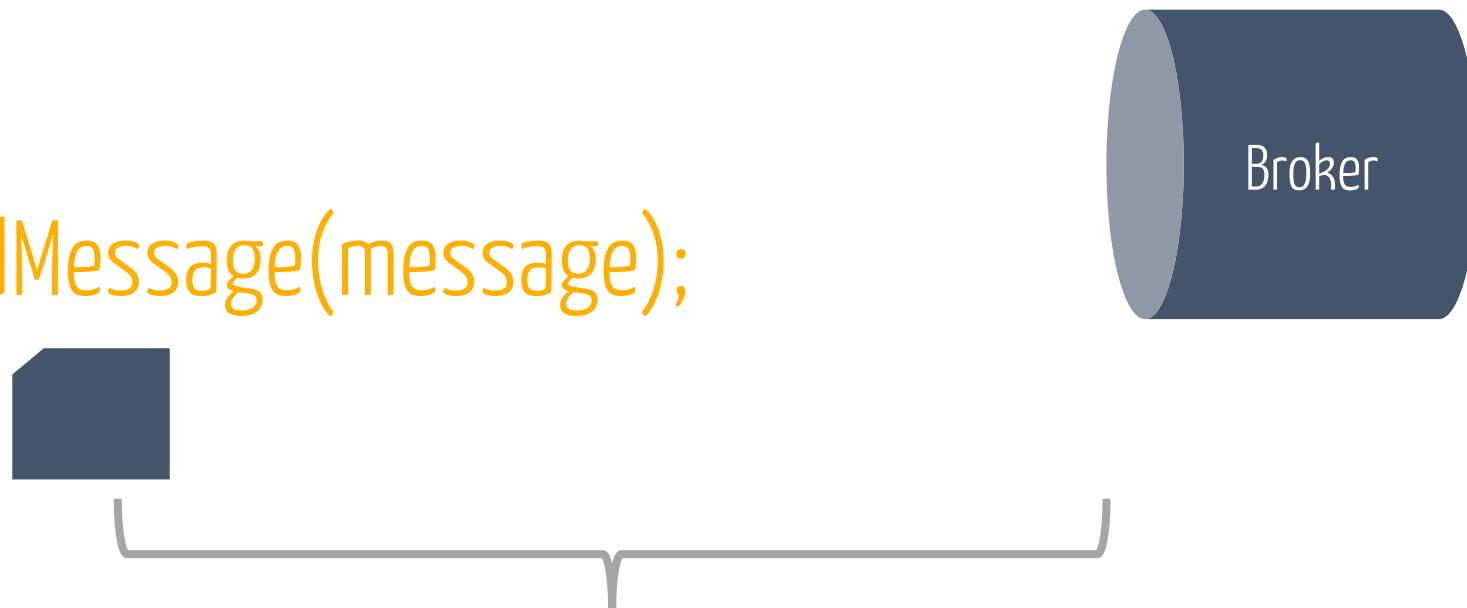


Penny Pinching

```
foreach(var message in messsagesToBeSent) {
```

```
    await SendMessage(message);
```

```
}
```

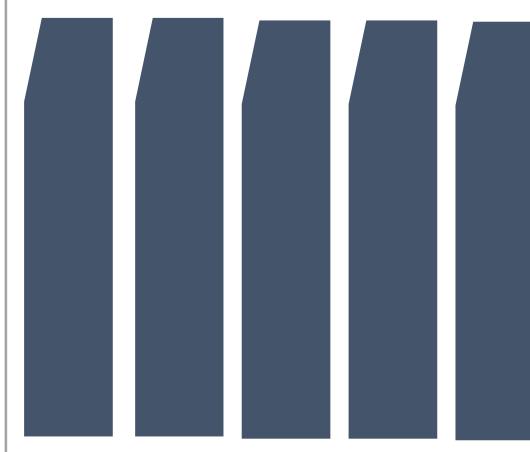
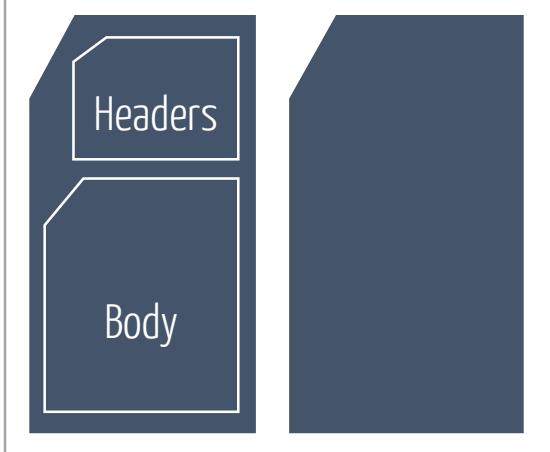


Latency / Operation costs

Destination 1

Destination 2

256 KB



All the messages to be sent

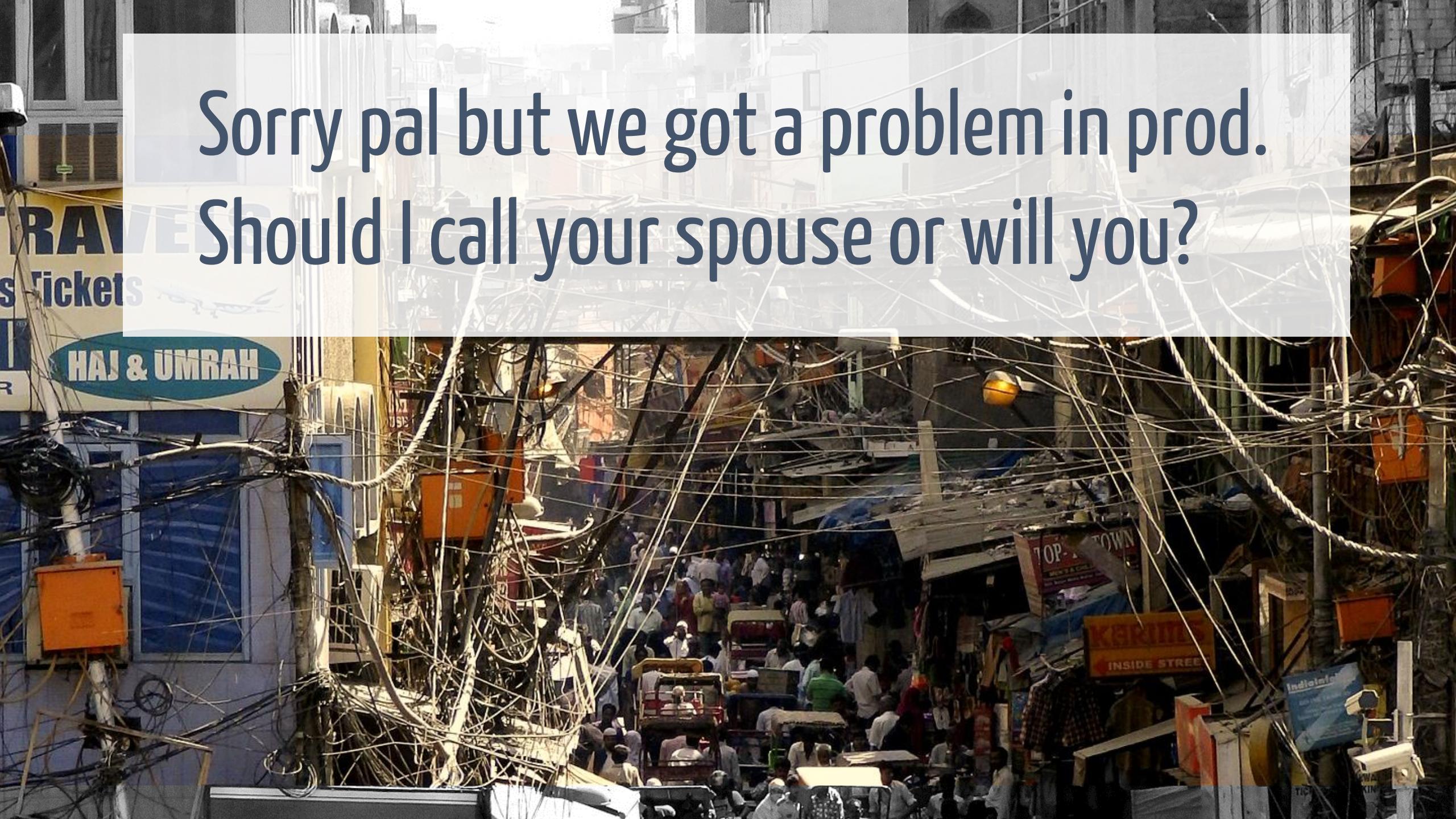
Now we have the

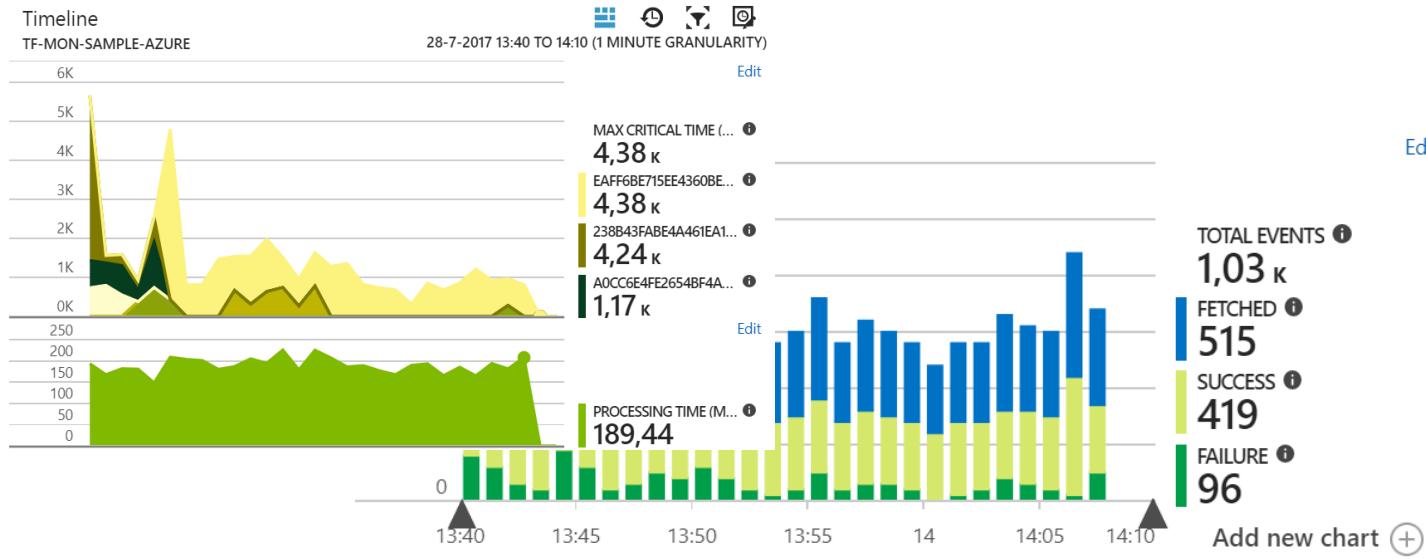
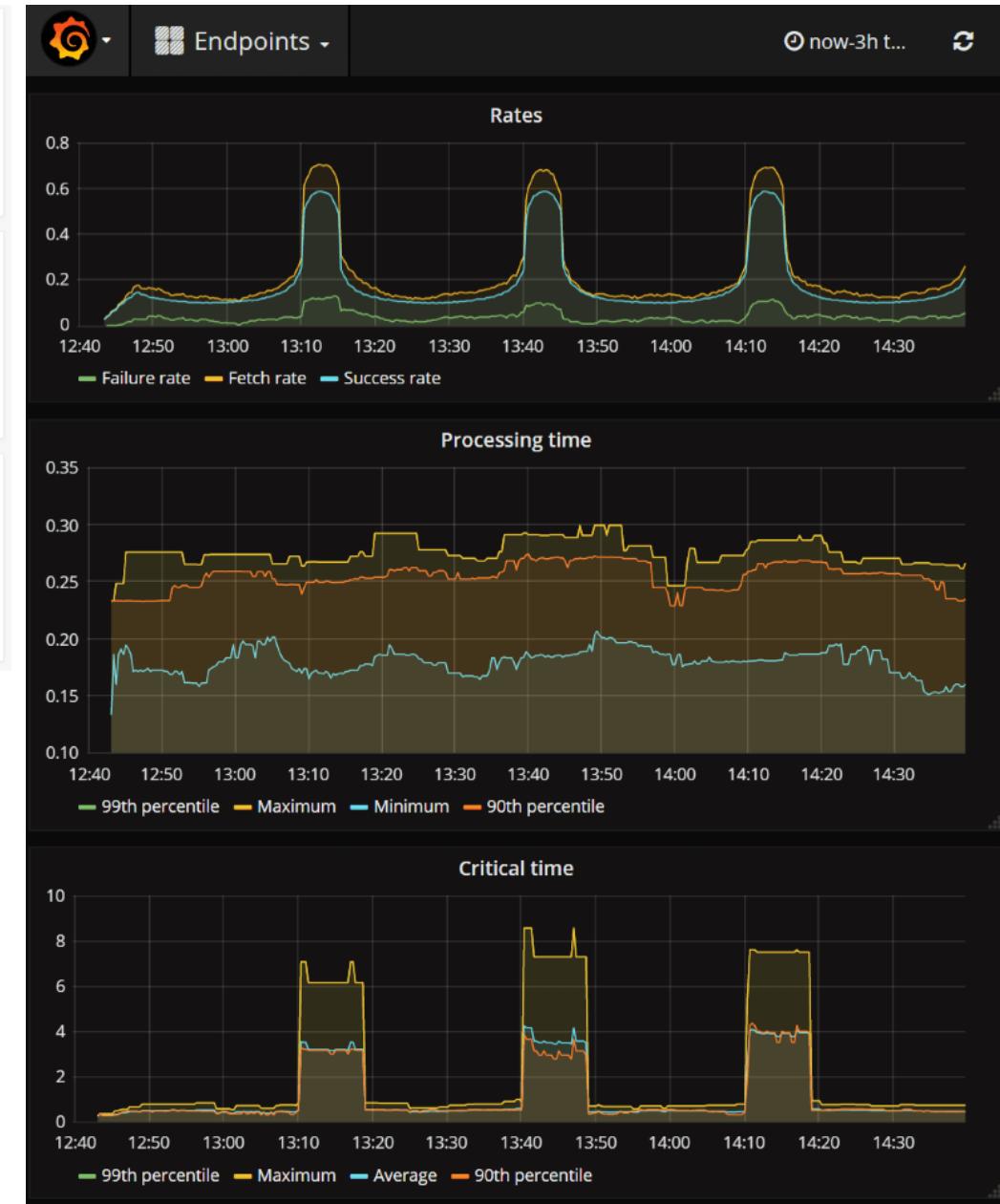
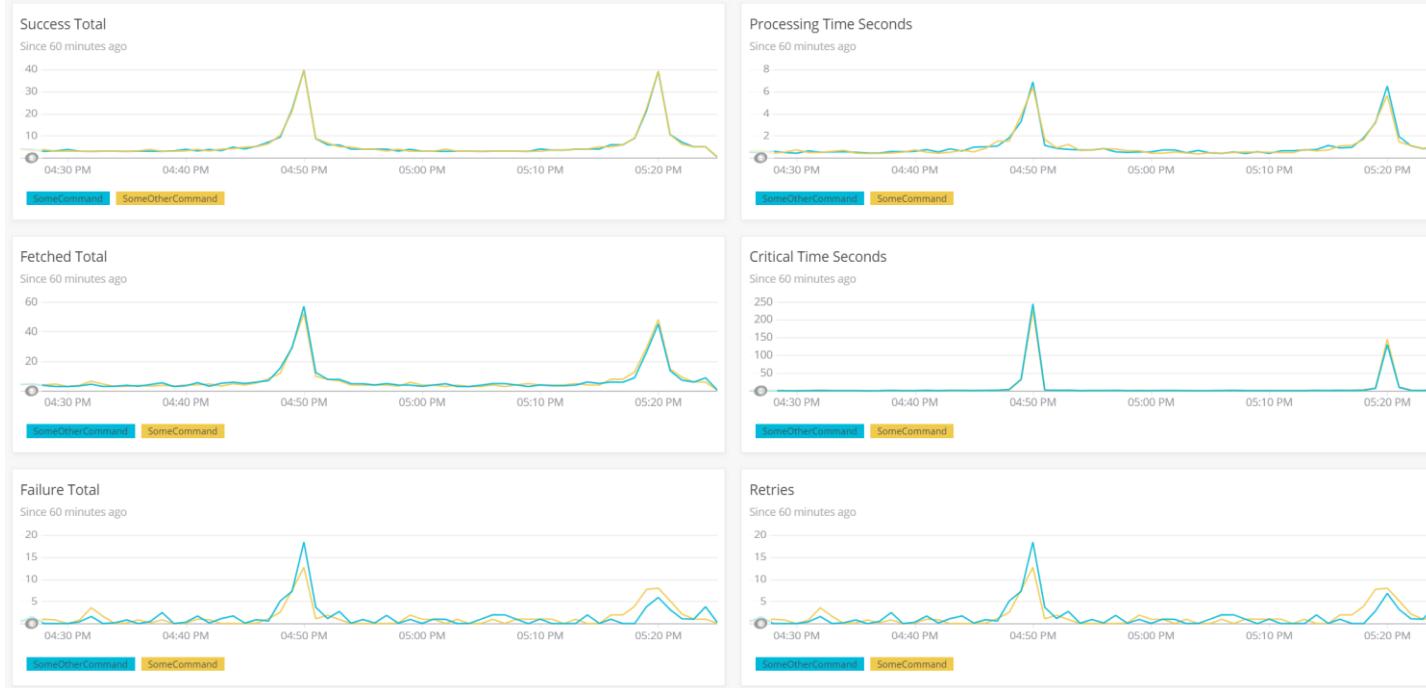
basic bits

Living on the edge of sanity



Sorry pal but we got a problem in prod.
Should I call your spouse or will you?



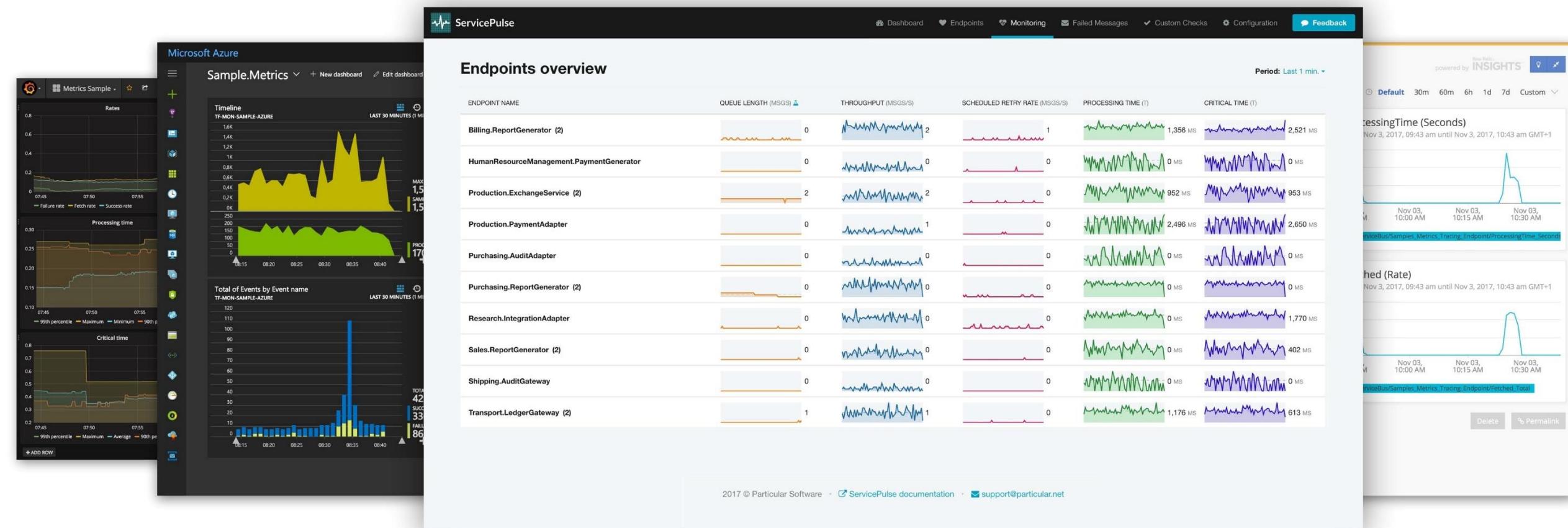




Try it yourself



Thanks



go.particular.net/demo

Slides, Links...

github.com/danielmarbach/MessagePump



Q & A



Software Engineer
Microsoft MVP



@danielmarbach
particular.net/blog
planetgeek.ch



Thanks