

Microservices with Service Fabric

Easy... or is it?

Goals

target

Stateless and **Stateful** Services

Partitioning of Business Data

Message Patterns and Partitioning

Premise

A tale of

chocolate

Karl's Sales Pitch

Our Chocolate Microservices

High Availability

Stateless Services

Automatic Rollback

Load balancing

Hyper Scale

Data Partitioning

Stateful Services

Rolling Upgrades

Replication & Failover

Self-healing

Health Monitoring

Container orchestration & lifecycle management

On-premises or in the cloud

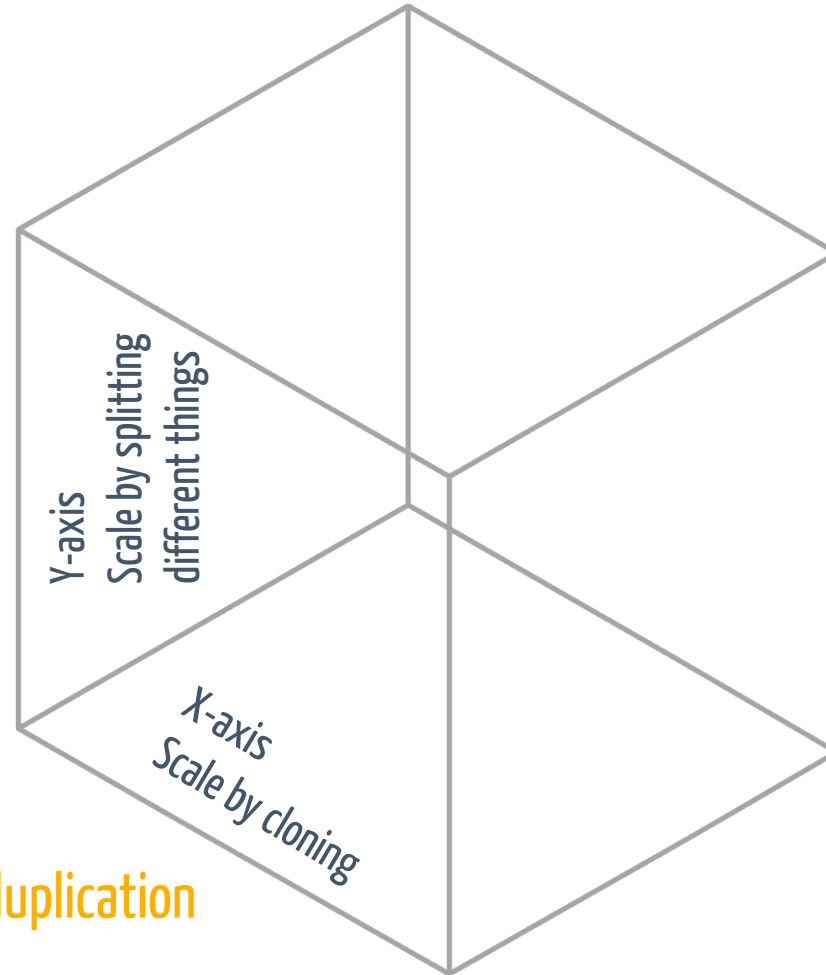
<https://channel9.msdn.com/Blogs/Azure/Azure-Service-Fabric>

<https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-overview>

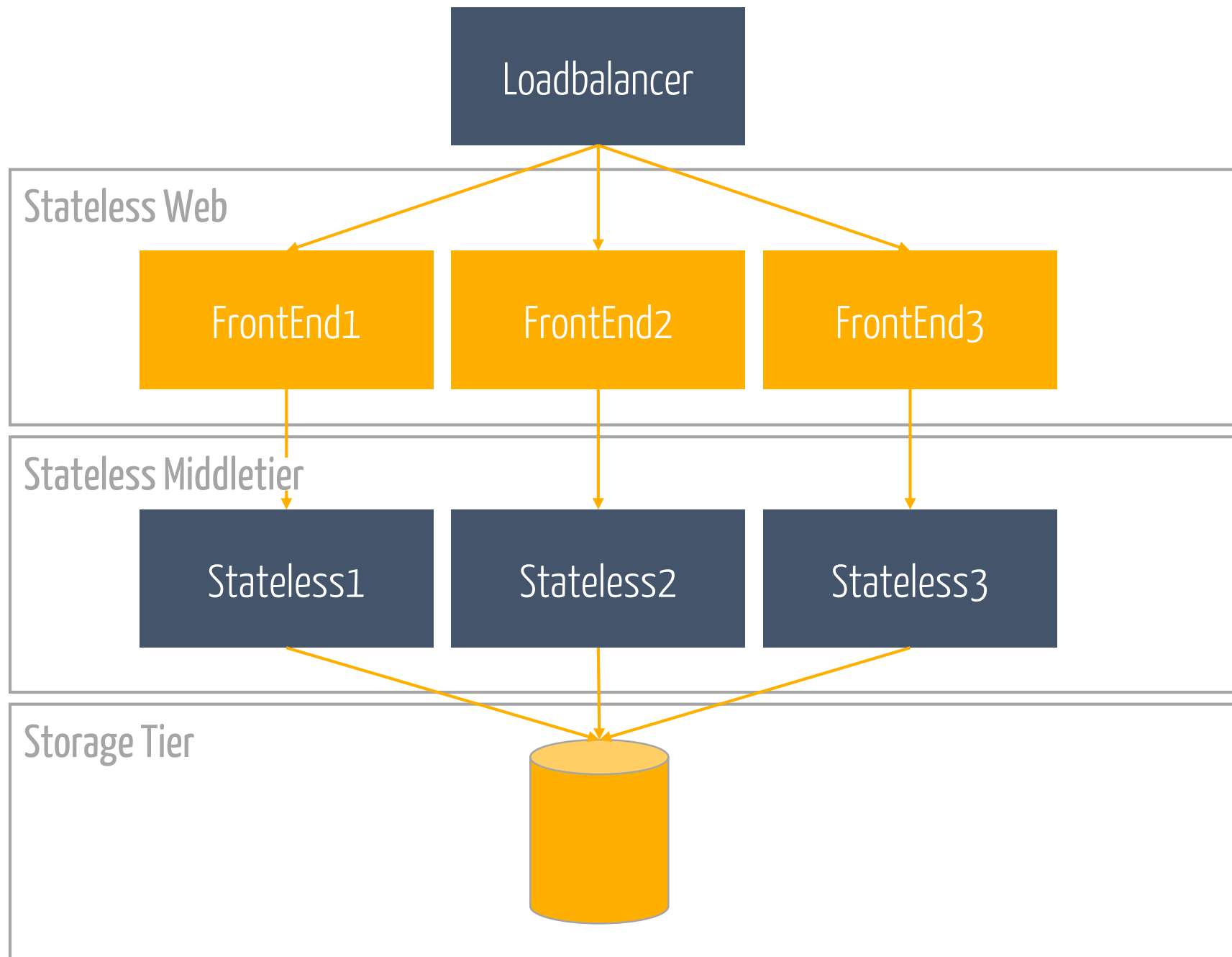
Demo

Let's scale
chocolate

Functional decomposition

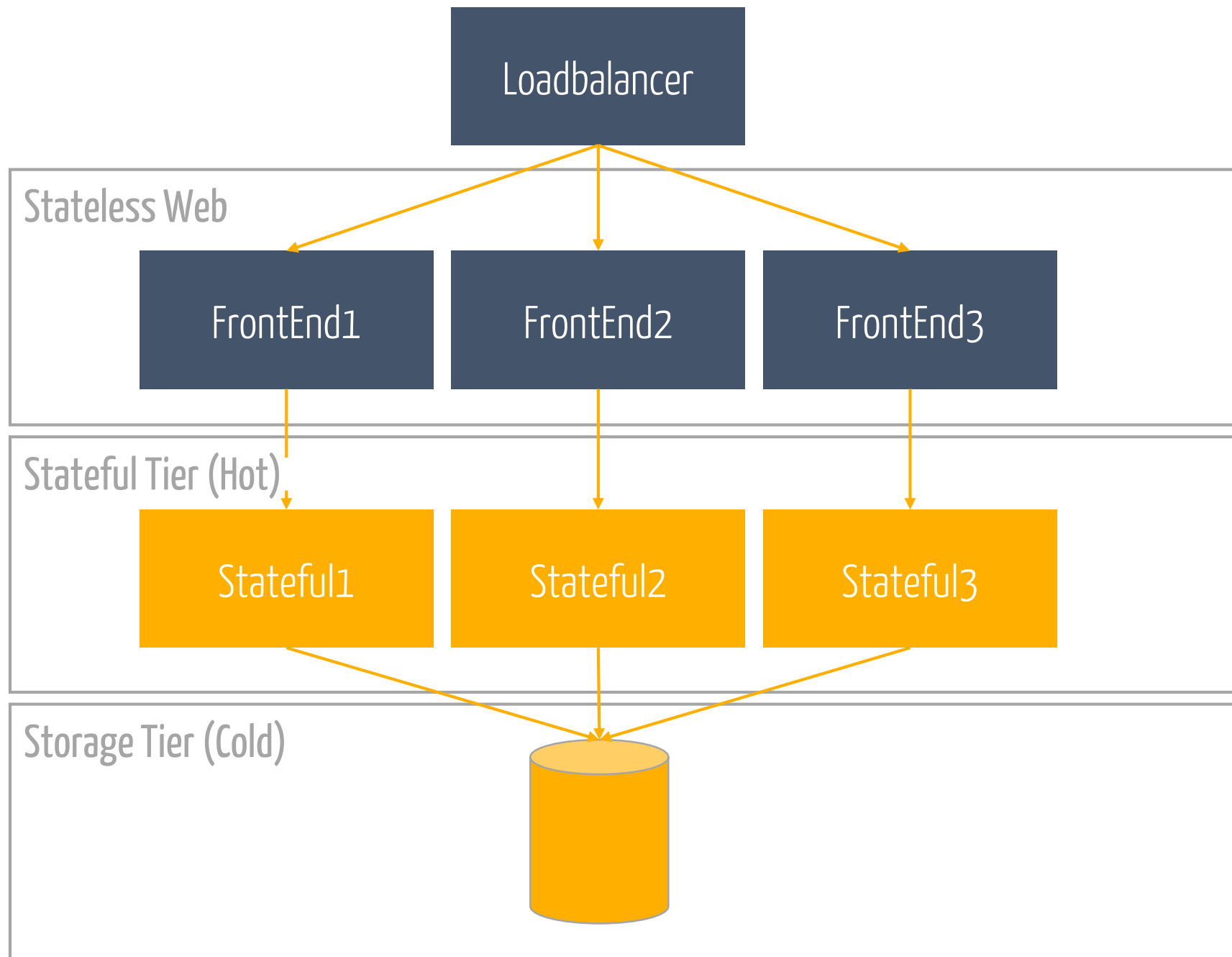


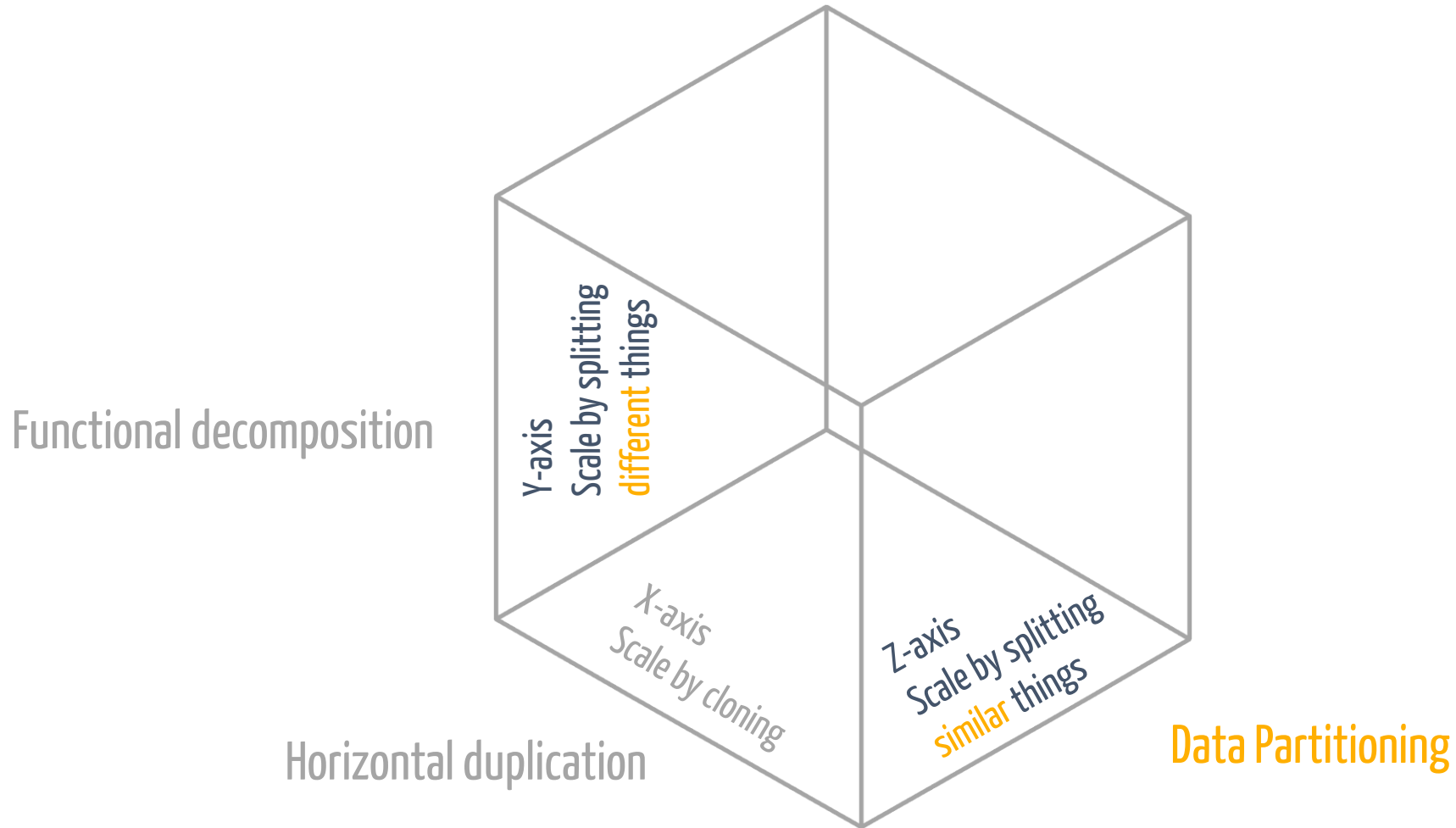
Horizontal duplication

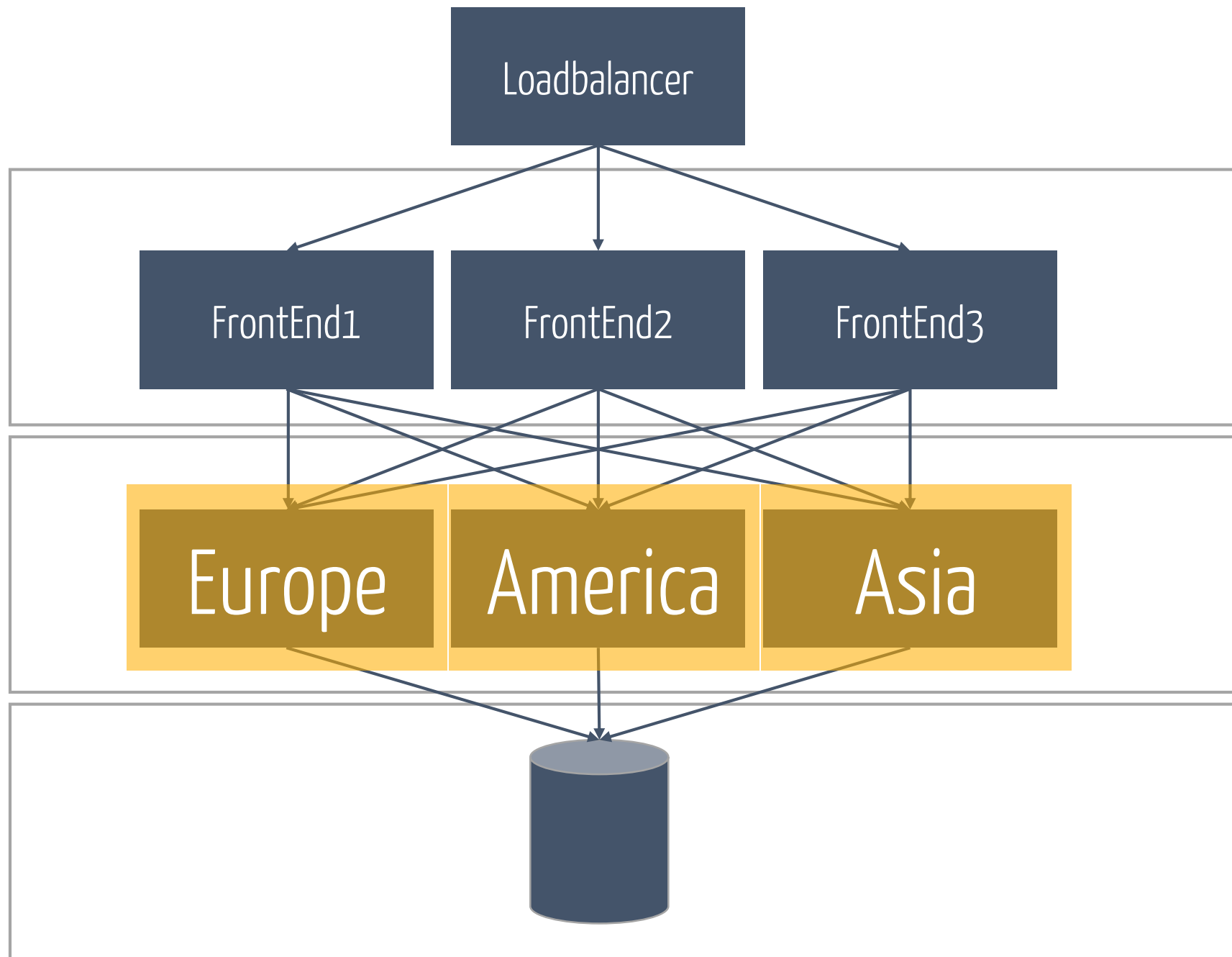


and then

Mandy spoke up

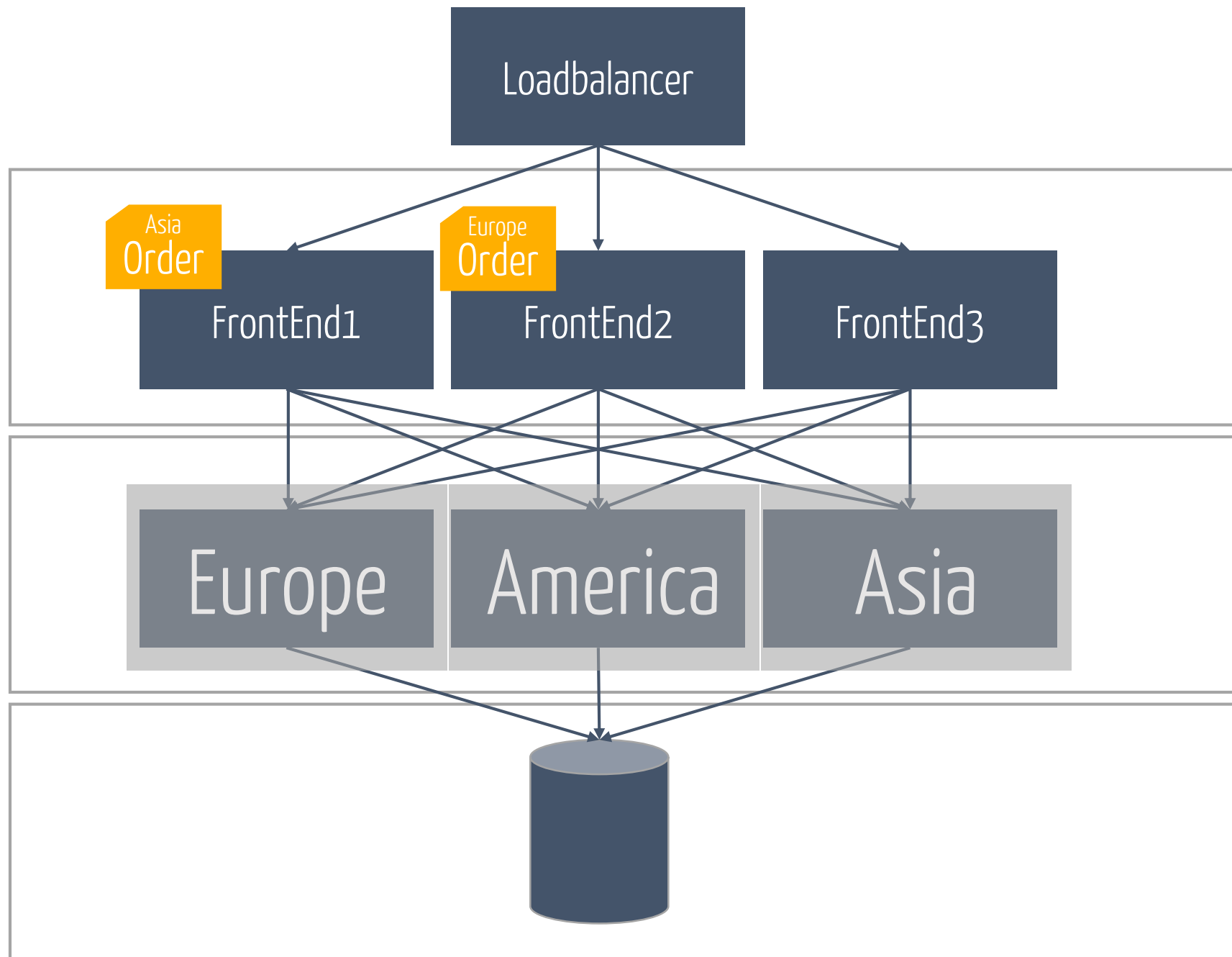






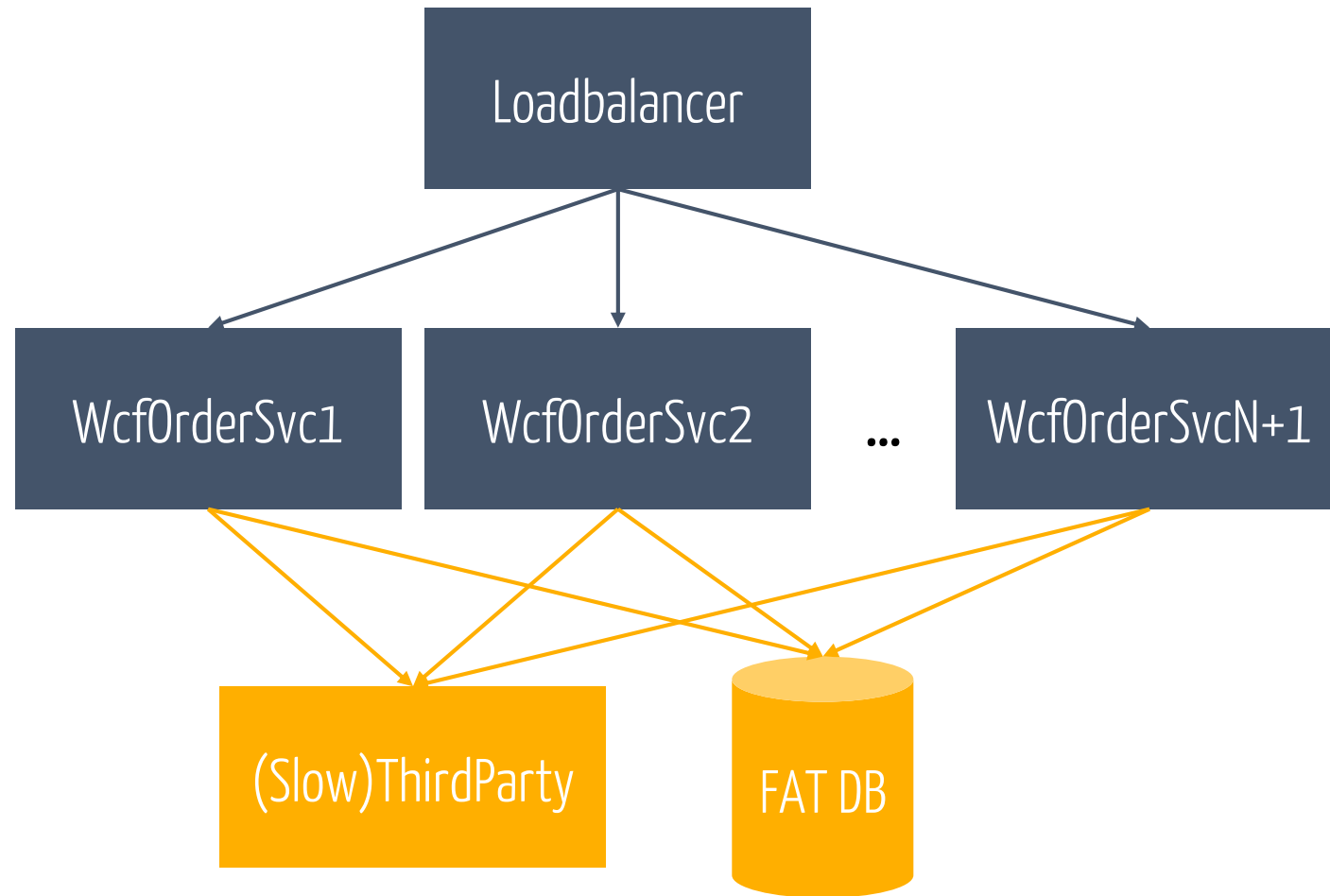
However

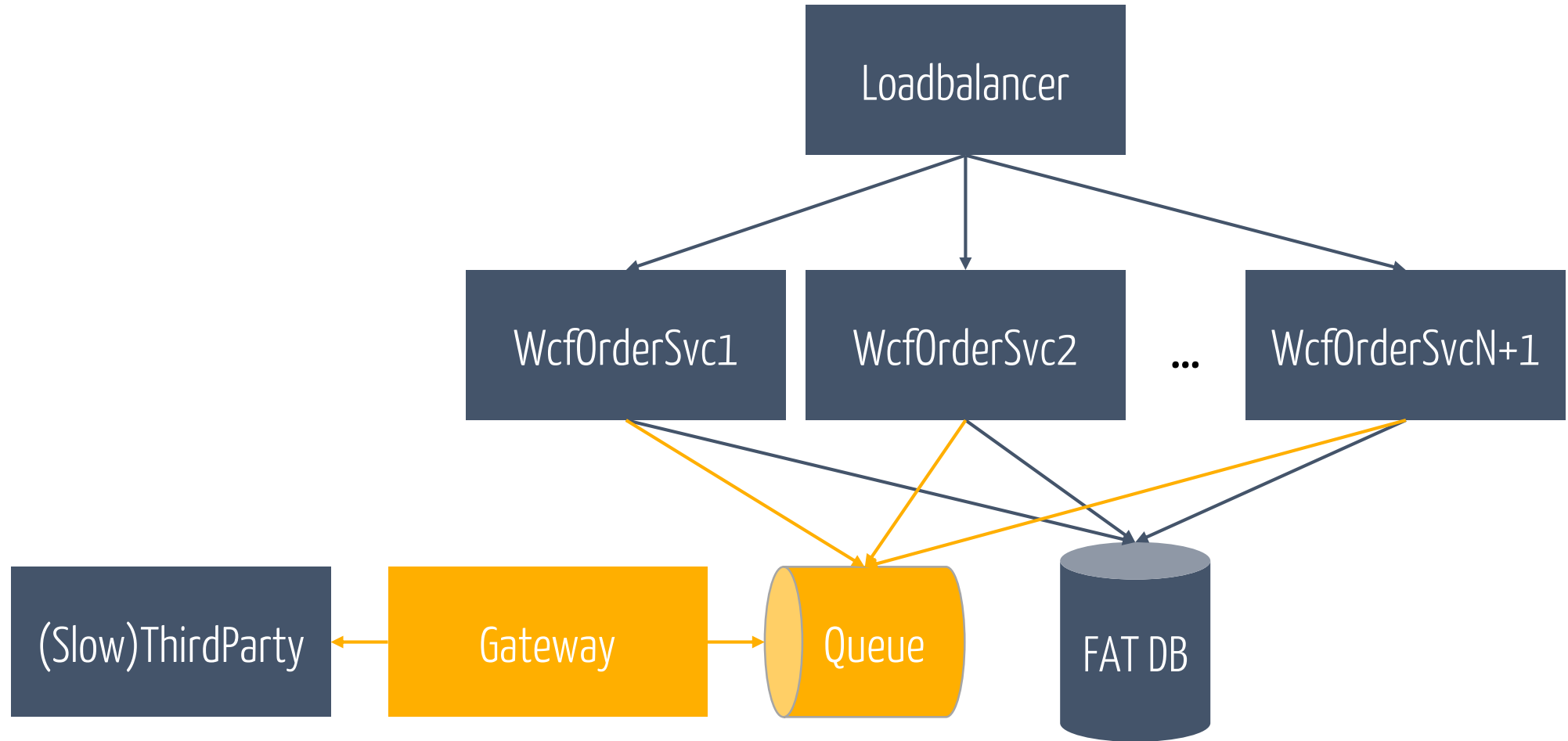
Joe couldn't understand it

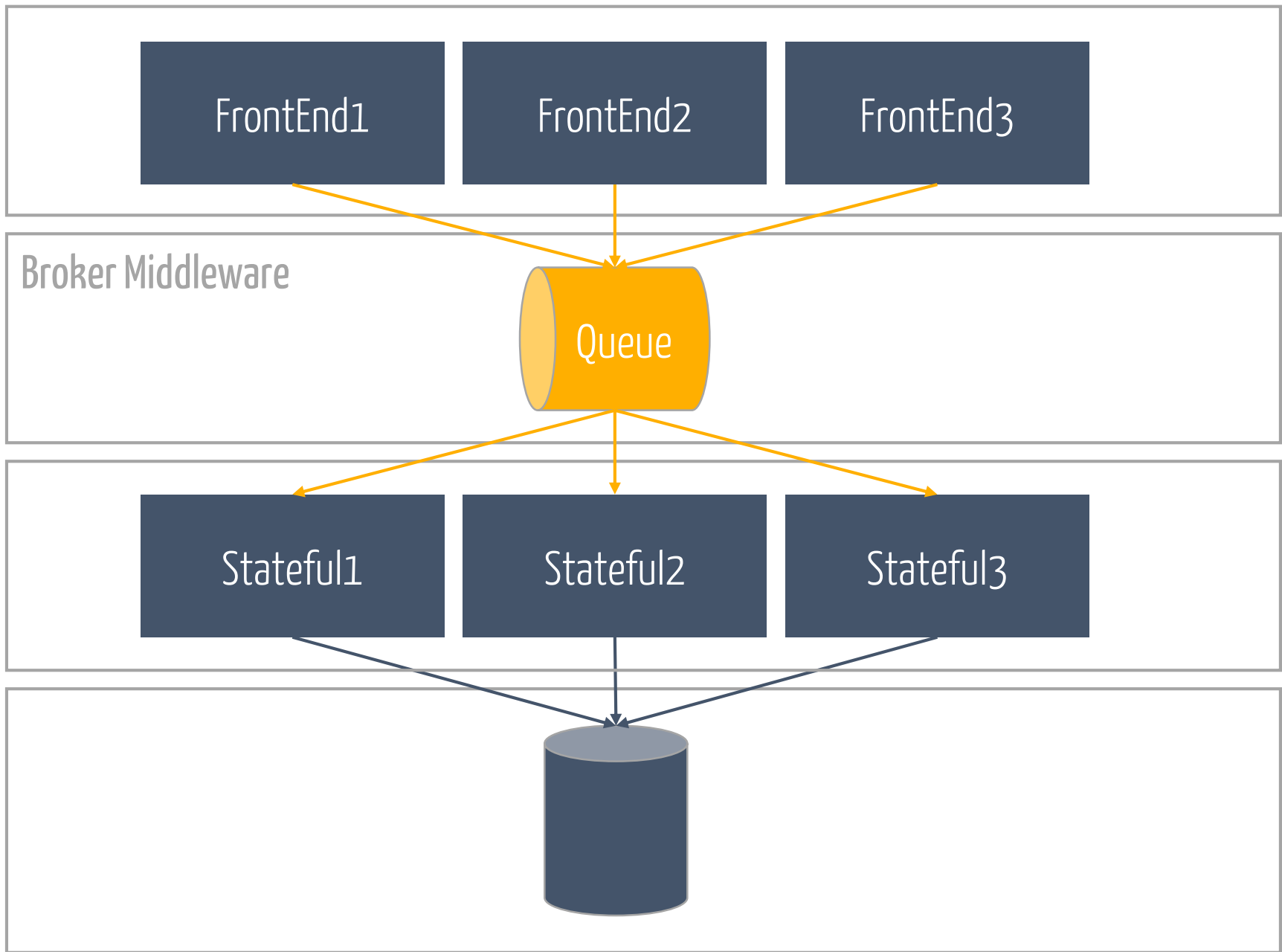


And then

Sophia threw a grenade





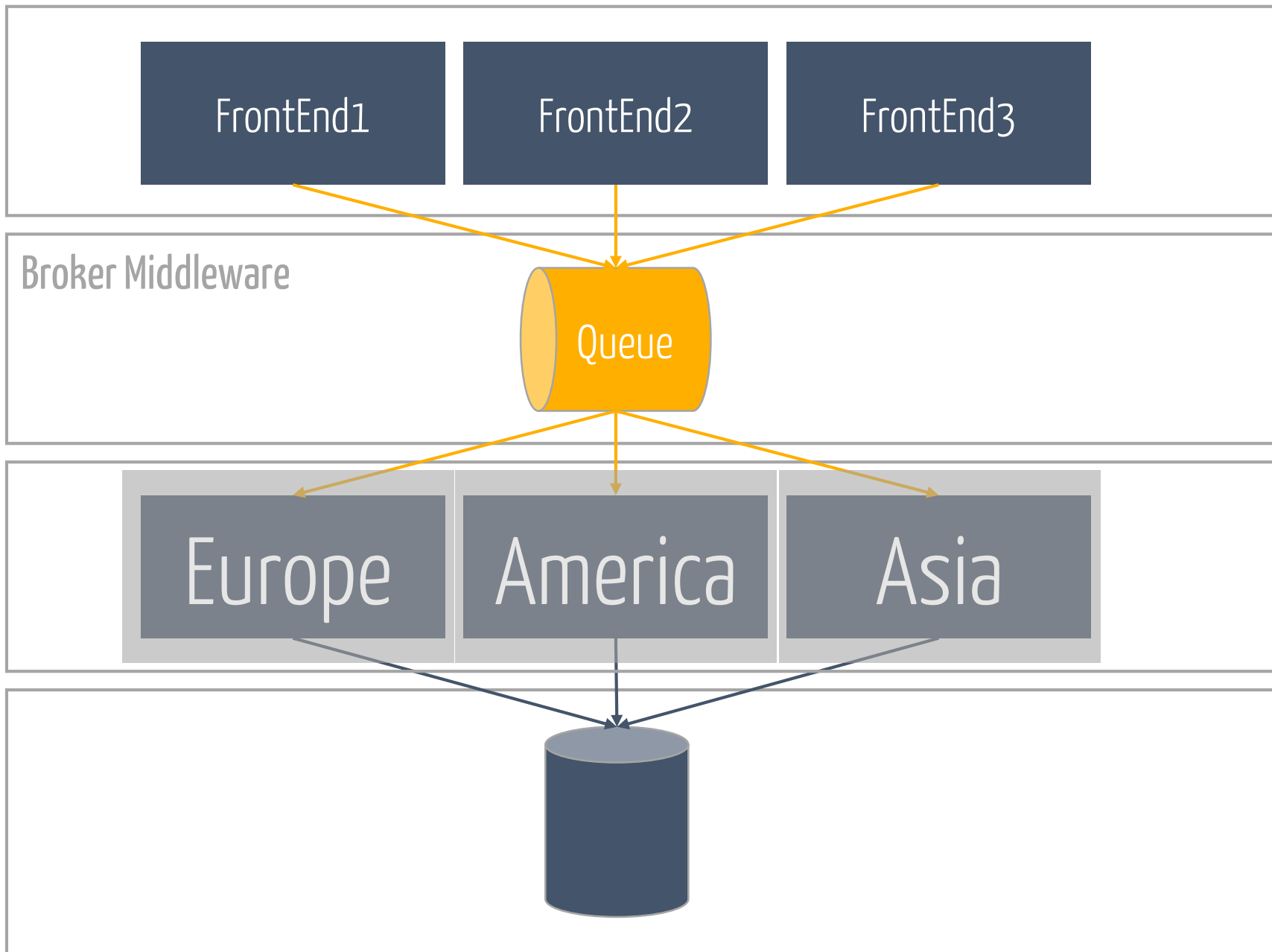


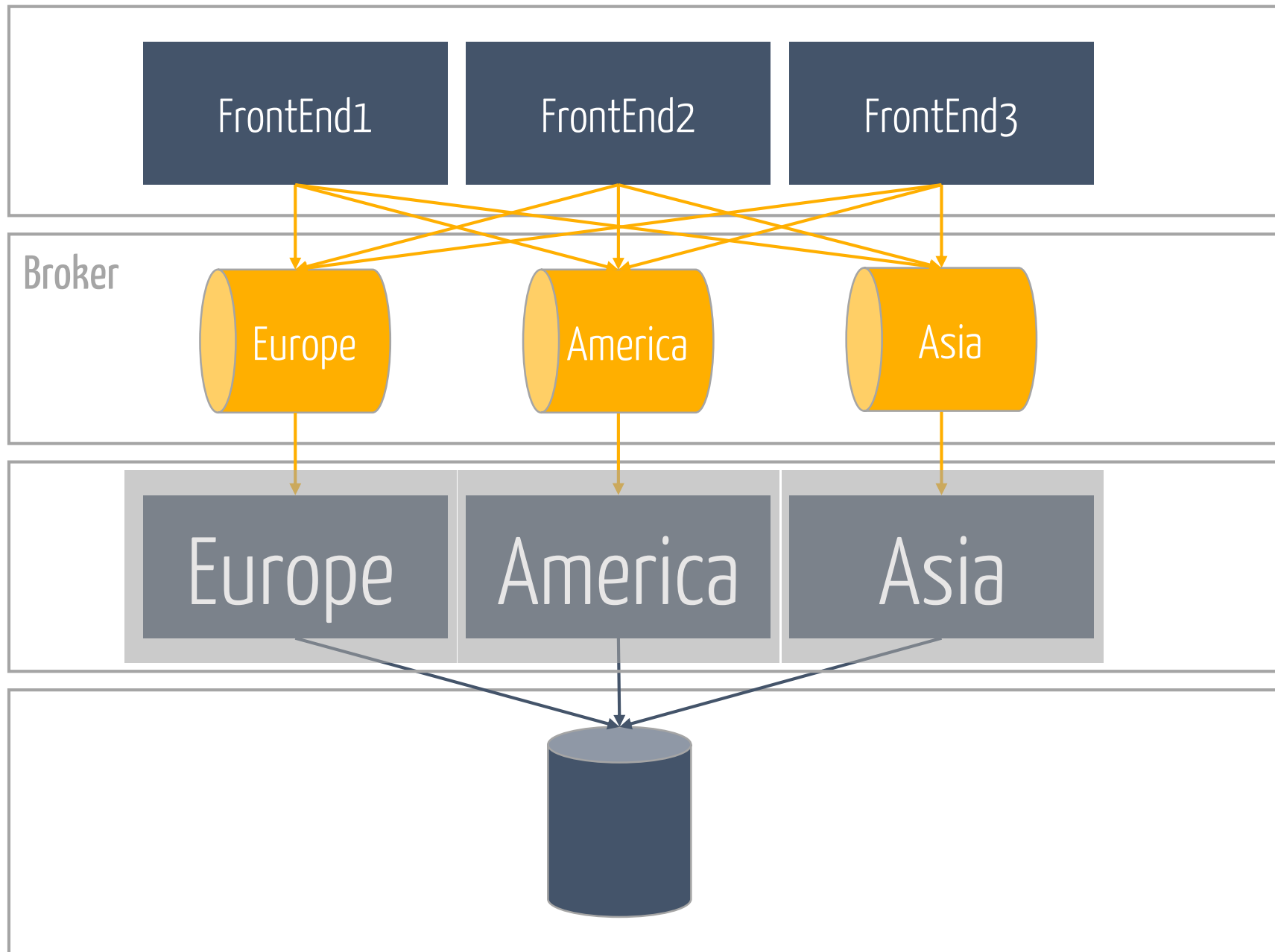
Peter snatches

the whiteboard markers

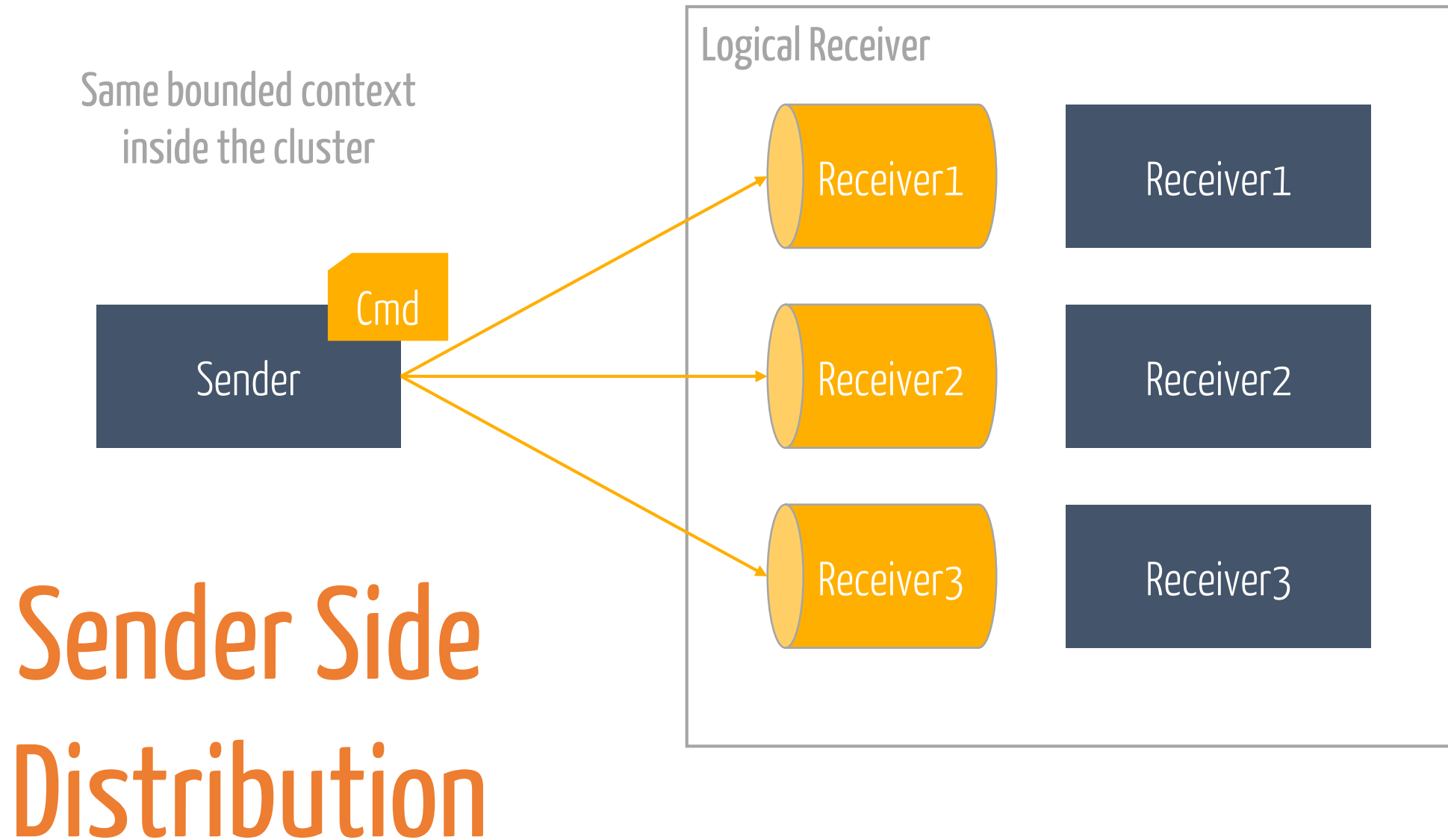
and furiously

screams





Commands

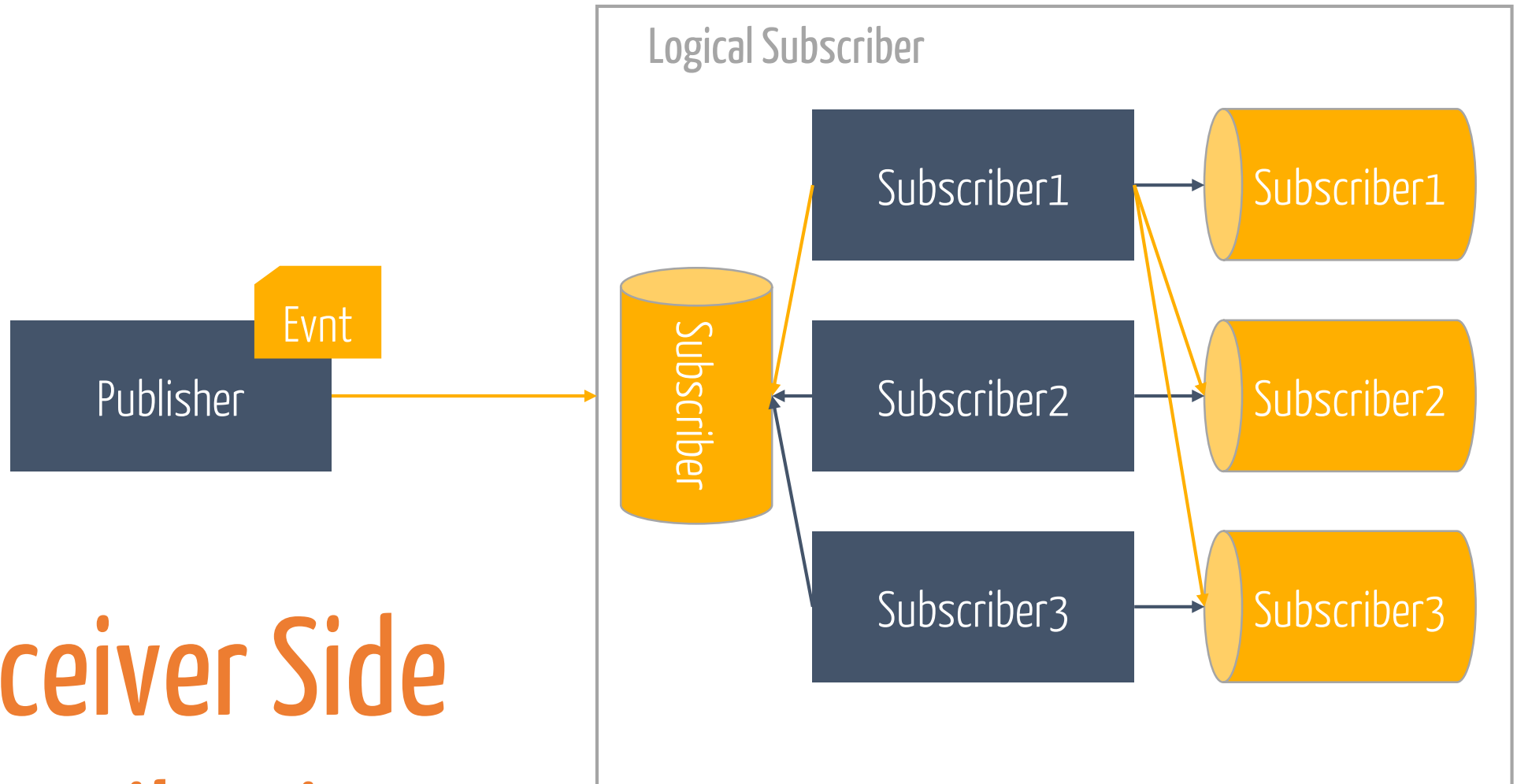


The Phd dude

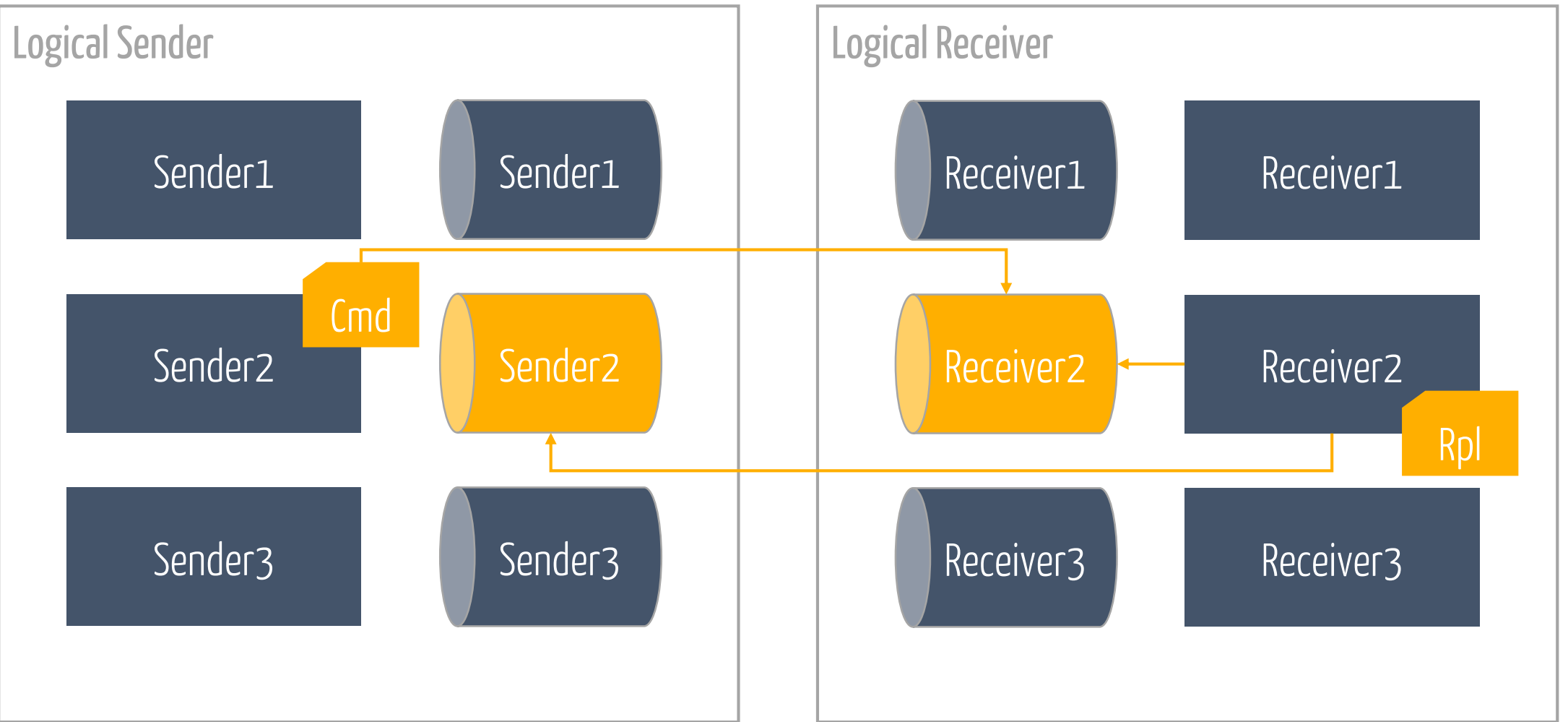
acts like a smart ass

Events

Receiver Side Distribution



Request / Reply



Demo

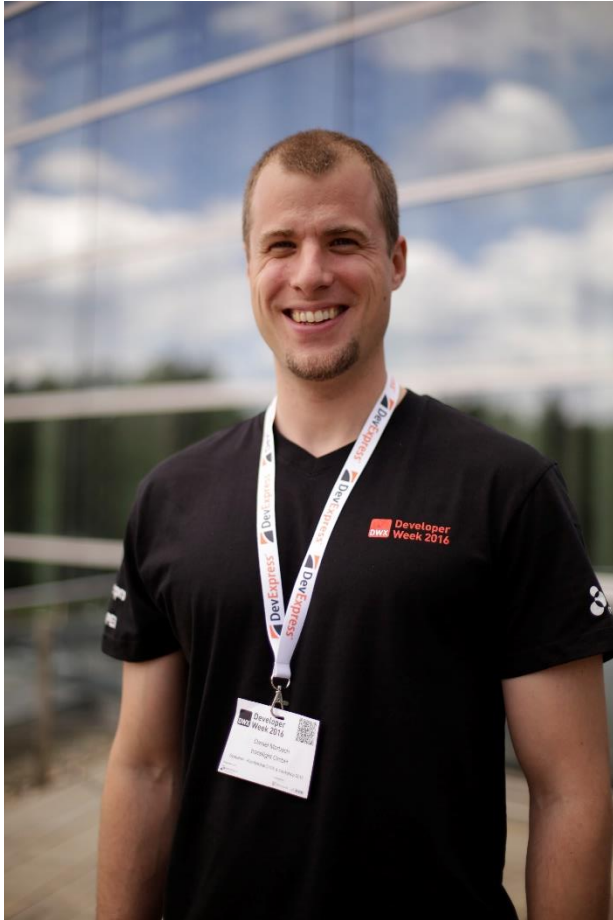
Recap

reminder

Slides, Links...

[github.com/danielmarbach/](https://github.com/danielmarbach/Microservices.ServiceFabric)Microservices.ServiceFabric

Q & A



Software Engineer
Enthusiastic Software Engineer
Microsoft MVP

@danielmarbach
particular.net/blog
planetgeek.ch

Thanks