November 9-11, 2021

www.dotnetconf.net

.NET Conf

Discover the world of .NET

Understanding Microservices

A guide for the monolithic developer

Layla Porter

Understanding Microservices

A guide for the monolithic developer

LAYLA PORTER

Developer Advocate, VMware
Microsoft MVP
GitHub Star
Former Director, .NET Foundation
MK .NET organiser

So you've decided you need microservices!

Are you sure?!

Okay, so you've definitely decided you need microservices!

What's all this gobbledy gook?
Where do I start?
How do I even go about creating a proof of concept?

Glossary of Common Terms

Service Registry
How applications and services locate each other

Load Balancer
Efficiently distributes network traffic across a group of backend services

Circuit Breaker
Stops an application from performing an operation that is likely to fail

API Gateway
A server that is the single entry point into the system

External Configuration
Keeps the configuration information in a centralised external store

Bounded Context
The boundary of a model that represents its concepts, relationships, and rules


Where to begin?
This is one of the hardest parts - figuring out what to do first!
In Stages
Refactoring
Fresh Start

Tacky Tacos

- E-commerce food ordering site
- Things go crazy on Tuesdays
- Want to scale to meet business demand
- Want to easily add in new services, like order tracking

Our Monolith

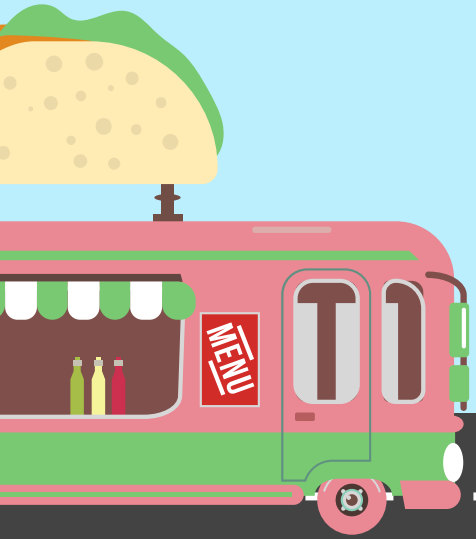
- User Interface
- Data access
- Ordering
- Payment
- Identity
- Notifications
- Admin
- Catalogue management

Our Monolith with updates

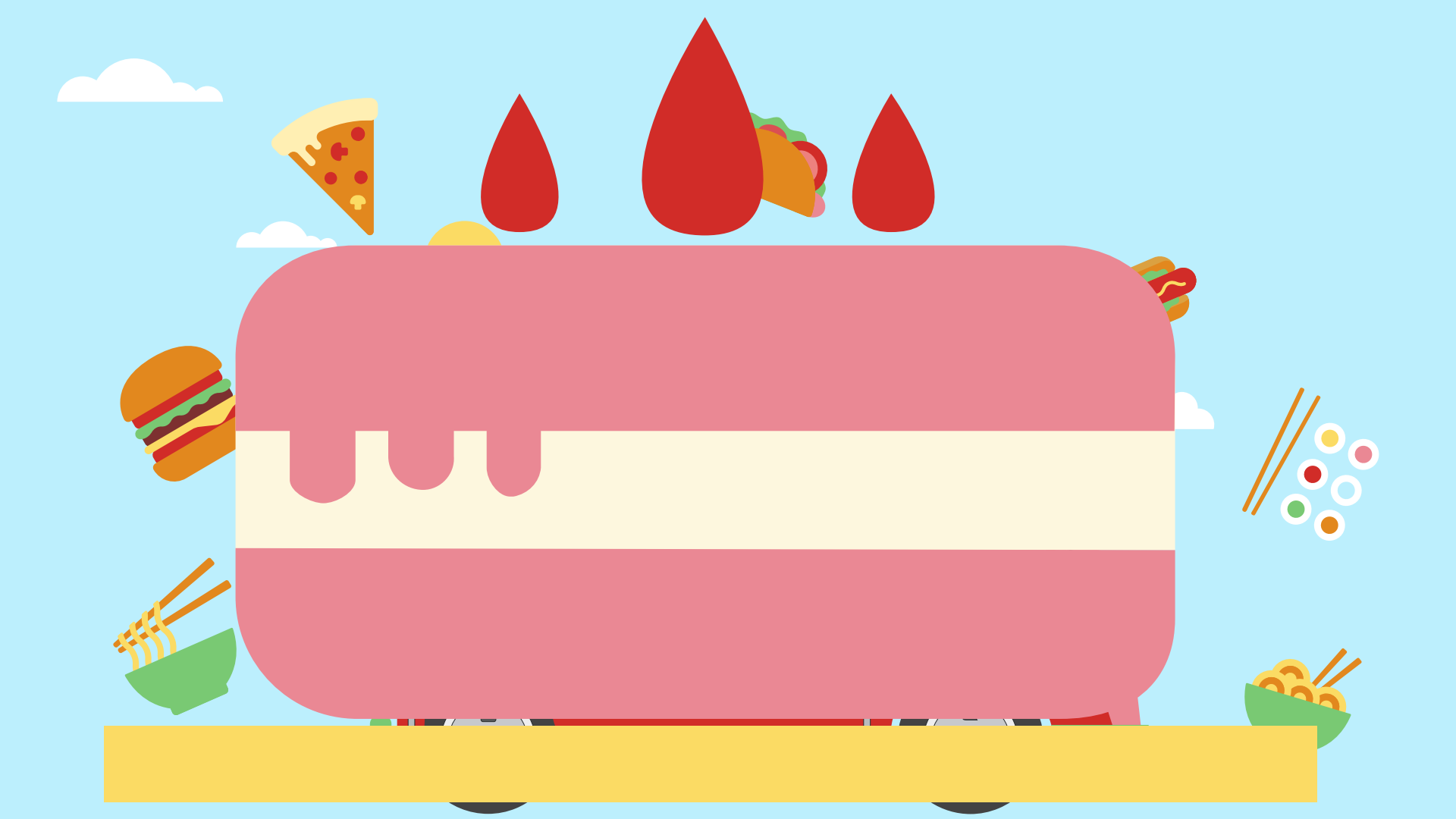
Customer Client

- User Interface
- Data access
- Ordering
- Payment
- Identity
- Admin
- Catalogue management

Notification service



Let's talk about scope!



A service should do one thing and do that one thing well!!

The scope of each service

Customer UI

Displays food items and collects orders

Order Service

Accepts orders and sends them out for payment collection

Payment Service

Contacts 3rd party payment provider and processes payments

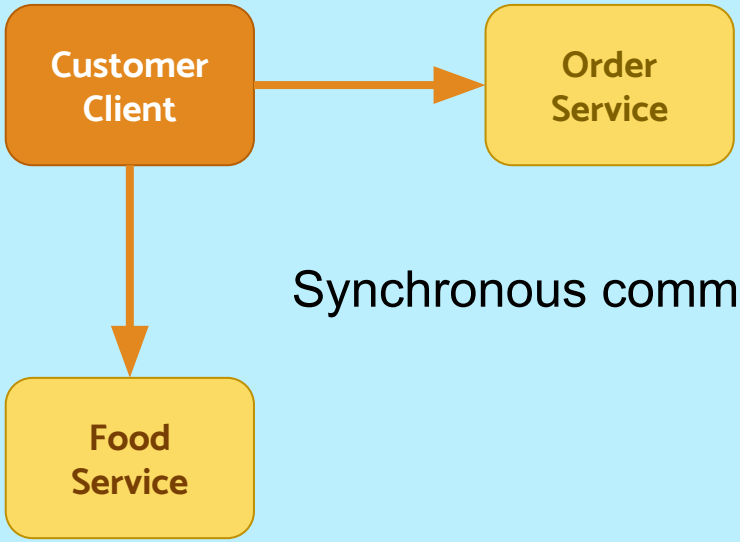
Food Service

Responsible for serving up all the available food and drink options

Notification Service

Handles all notifications from email to SMS

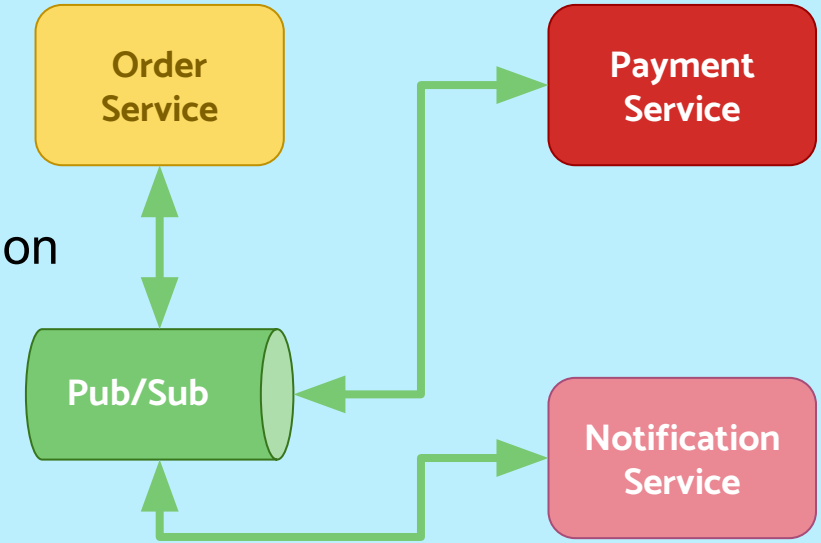
Our Services



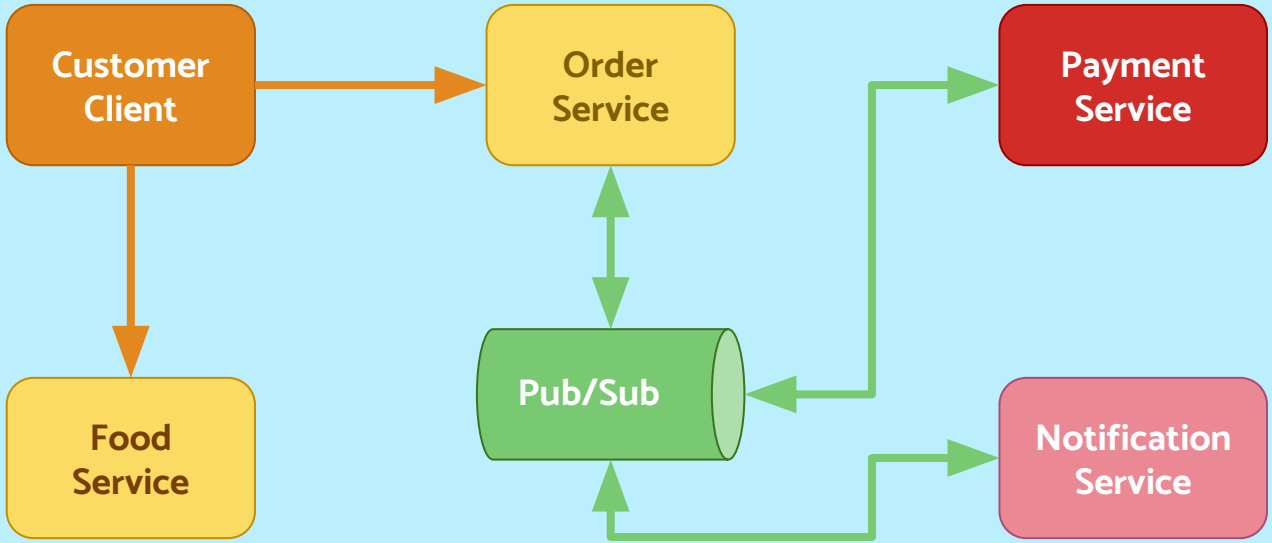
Synchronous communication

Our Services

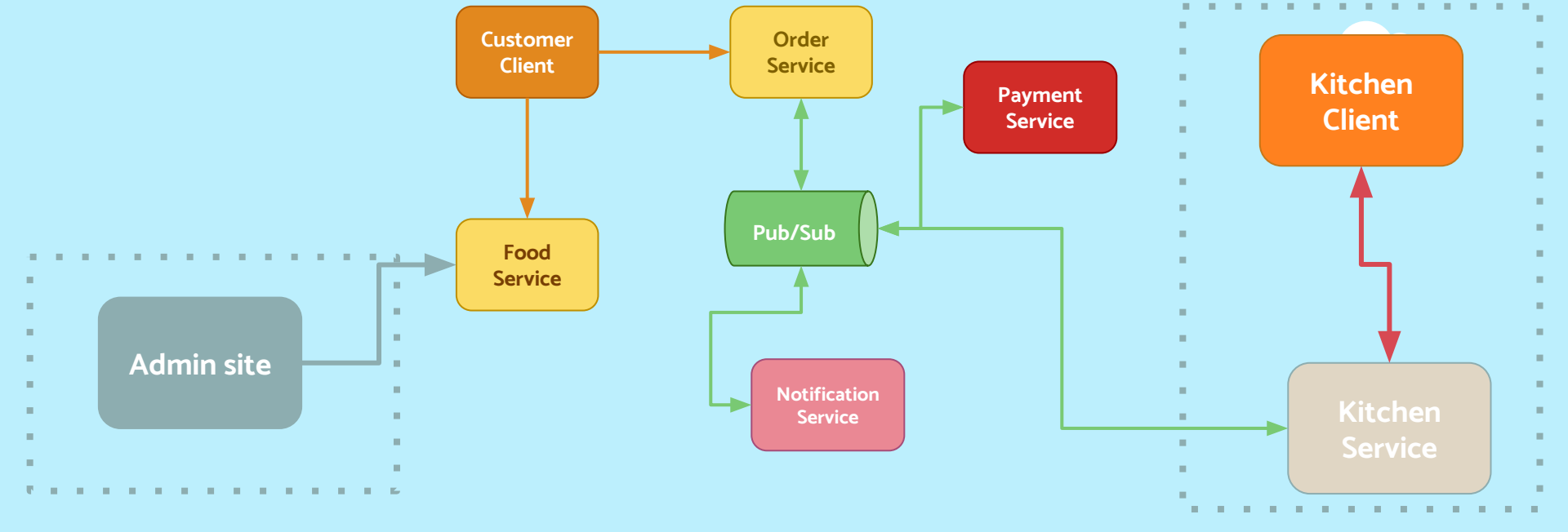
Asynchronous communication



Our Services



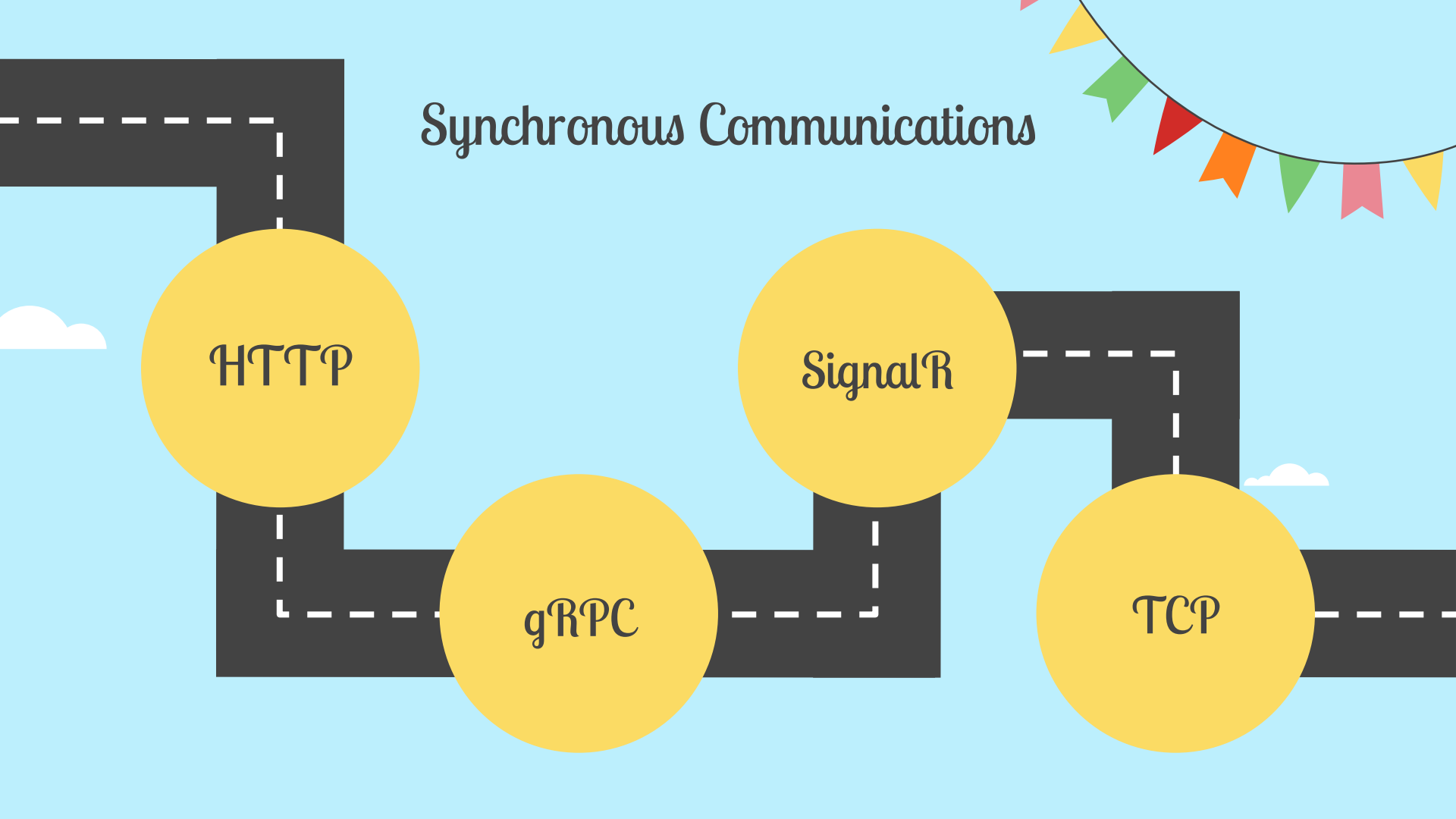
Our Future Services



01

Communication

How should our services communicate with each other?


Synchronous Communications
HTTP
gRPC
SignalR
TCP

Asynchronous Communication

Queues

one-to-one comms

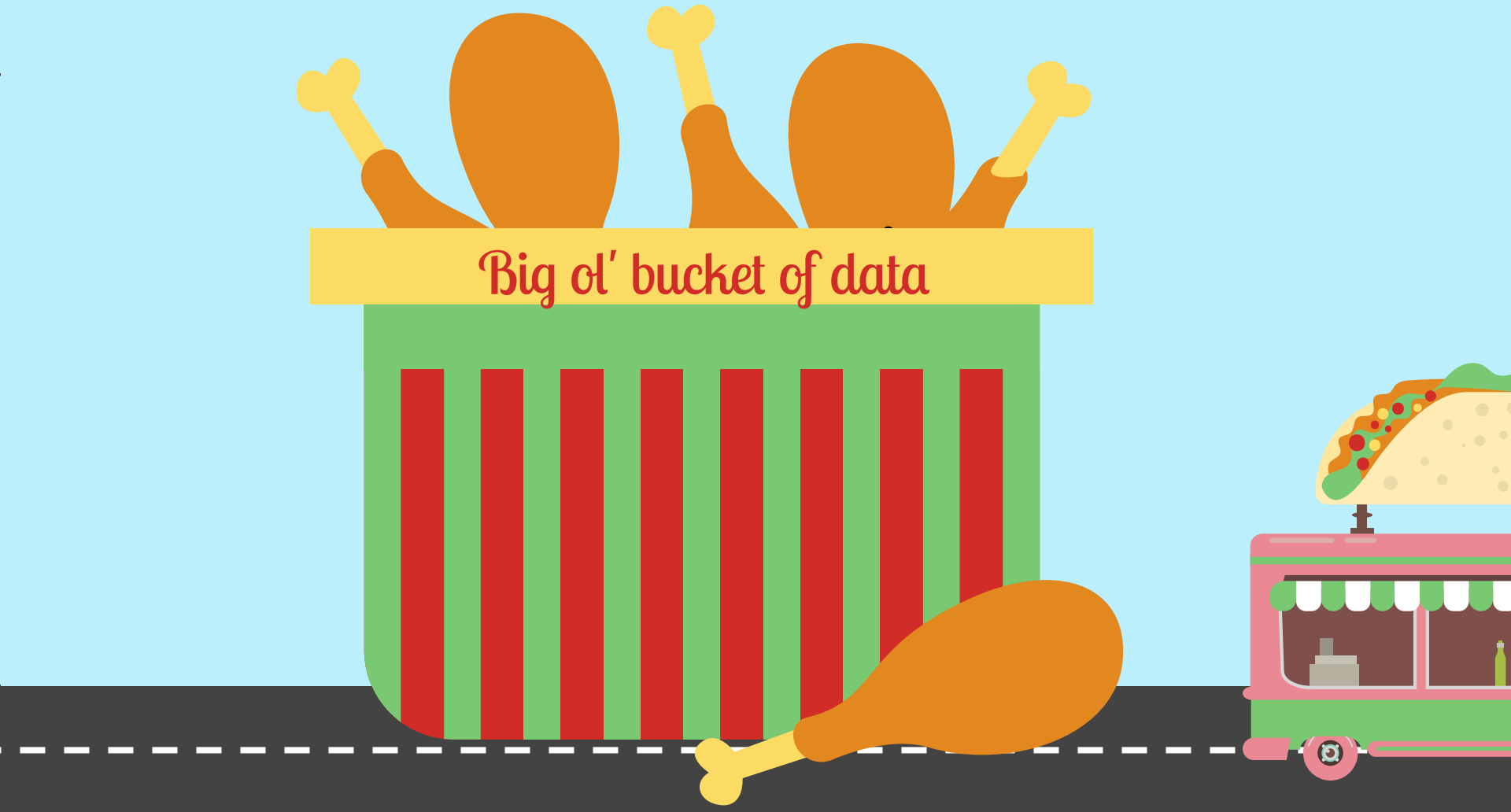
Buses

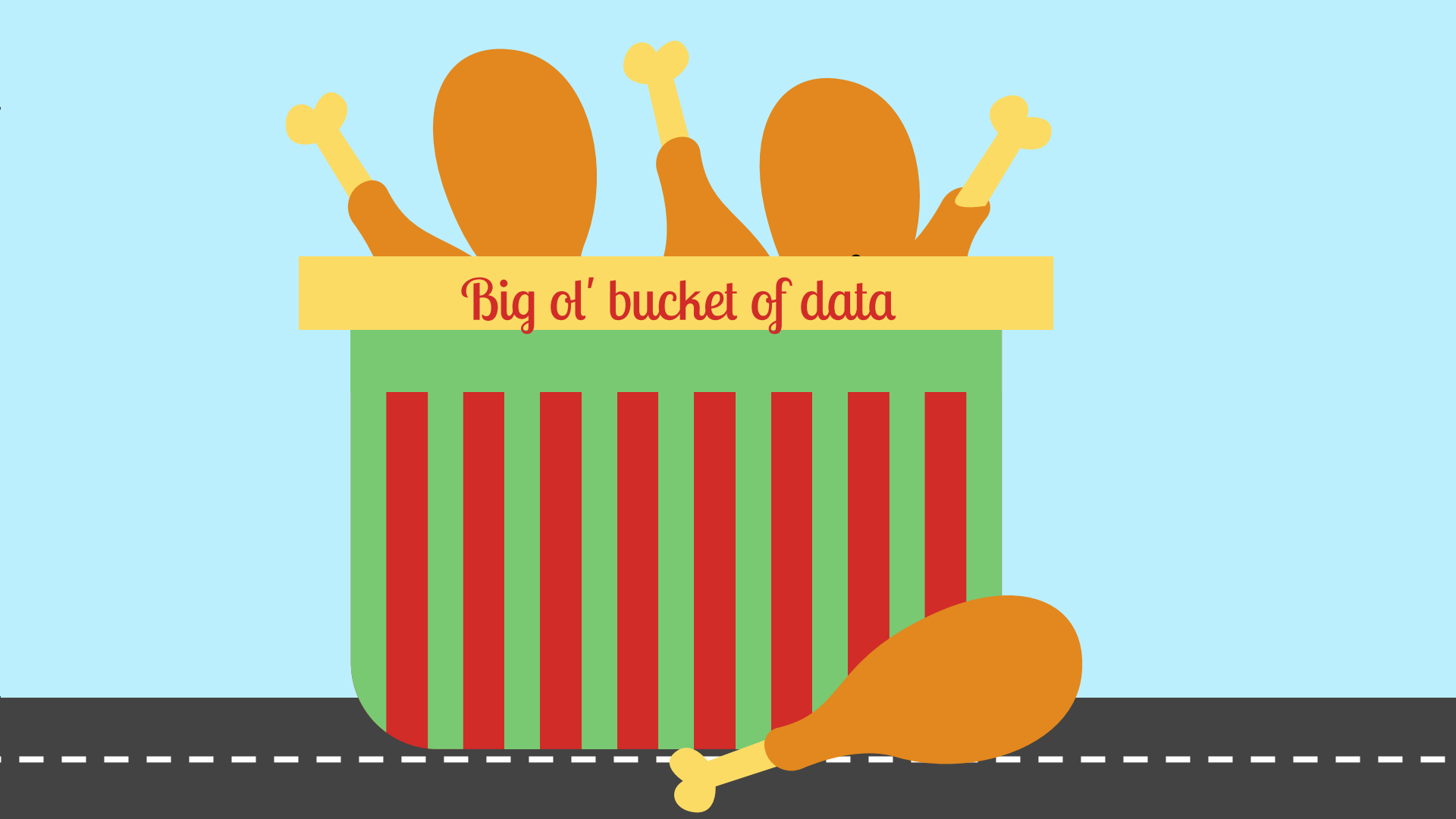
On-to-many comms

02

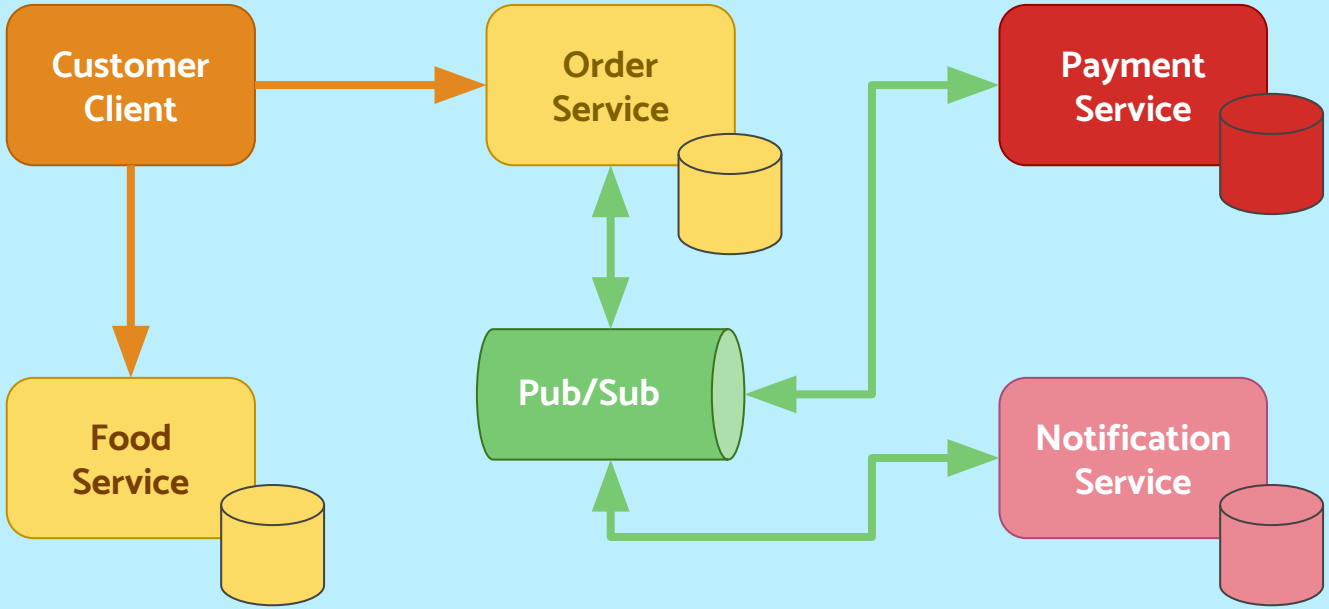
Data

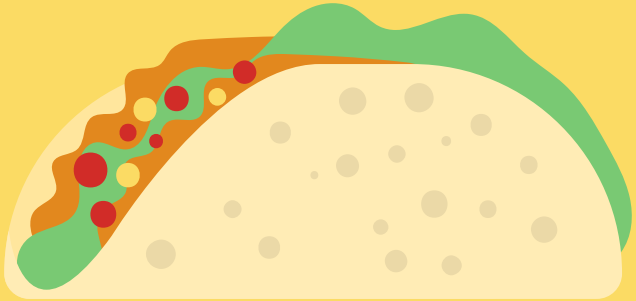
And what to do with it?


Big ol' bucket of data


Big ol' bucket of data

Our Services





Food Service

- Table Storage
- Cheap

Order Service

- Relational
- User Data
- Redundancy

Payment Service

- DocumentDB
- Highly scalable

Eventual Consistency

a consistency model used in distributed computing to achieve high availability that informally guarantees that, if no new updates are made to a given data item, eventually all accesses to that item will return the last updated value

Wikipedia

03

Security

Cue the silent groans!

Machine to Machine (M2M)

Ensuring that only approved apps, like the client app can access data that doesn't require user sign-in

User

Authorising and authenticating users via a sign-in

Security Options

Cookie

Client web app stores the cookie

Token/Claims

Used when no way to store a cookie

IAM

Such as OpenID, Oauth, Kerberos, Okta, AzureAD

Don't roll your own!

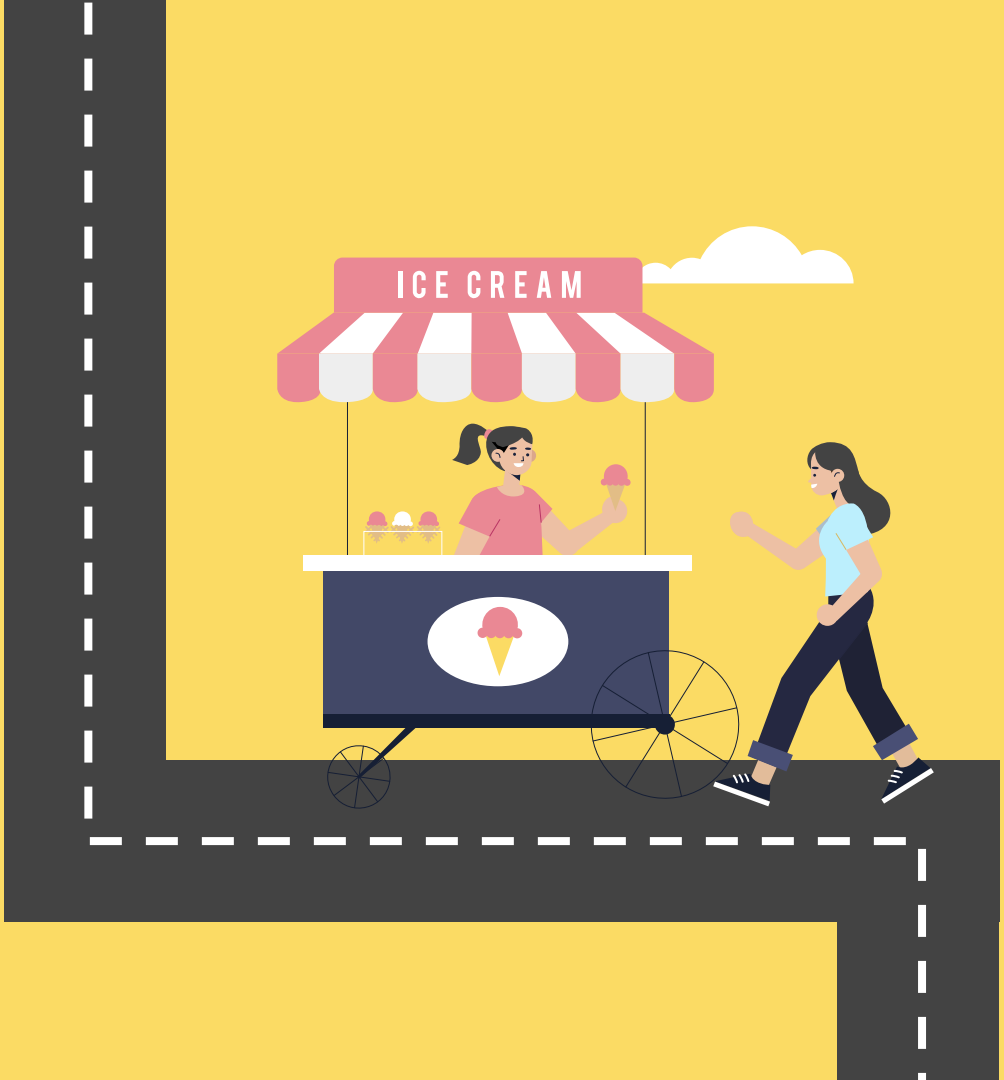
04

API Gateway

What is it?

An API gateway sits between the frontend client apps and the backend services

- Sometimes called a reverse proxy
- Single entry point so is well placed to control access to services
- Makes securing applications easier and more robust

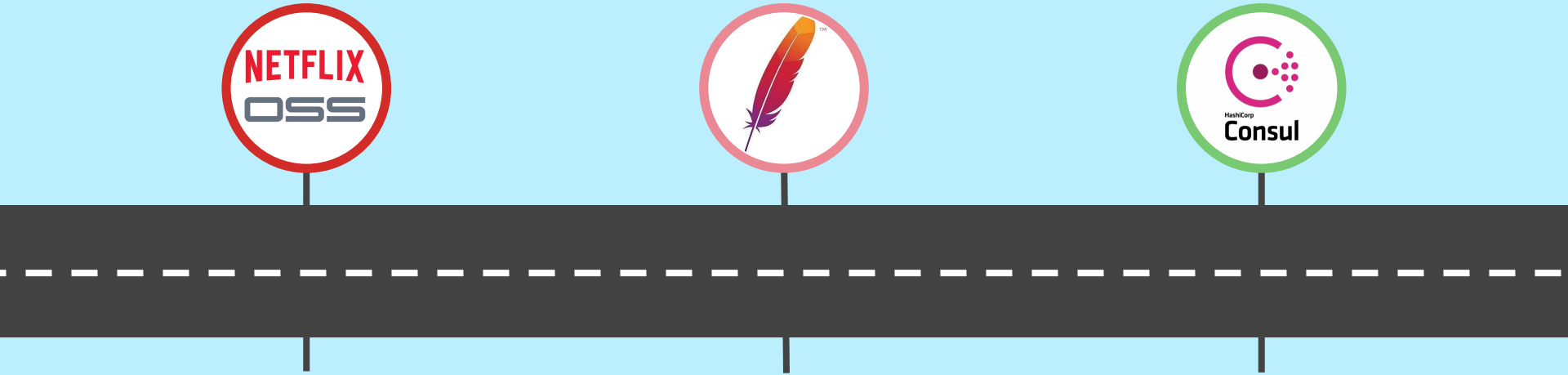


05 Service Discovery

Sometimes called a service Registry, too!


Where are
you again?

Some common Service Discovery options



Eureka

AWS Service registry for resilient mid-tier load balancing and failover.

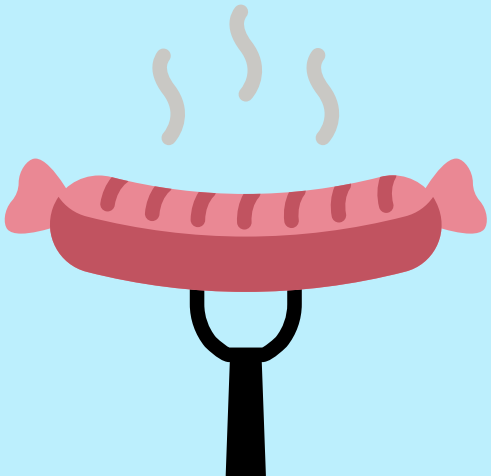
ZooKeeper

Centralized service for maintaining config, naming, distributed synchronization, etc

Consul

Distributed service networking layer to connect, secure, and configure services across any runtime platform

06
Actually writing code!



Things to bear in mind

Plan the scope of each project

Plan the right solution for each project

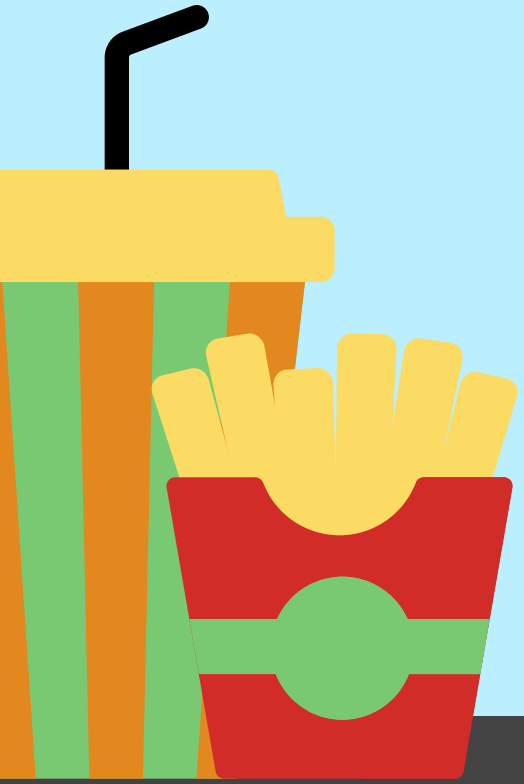
Layer in complexity

Be flexible

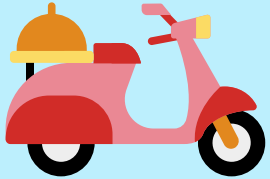
07

Other stuff

This topic is vast!



Things we haven't covered



Deployment

Deploying microservices is waaaaaaay beyond the scope of this talk!

Monitoring

Monitor all the things!

Availability

Scaling and load balancing

And probably a whole lot else....

08

Resources

Some cool stuff to check out!

Libraries etc

Steeltoe

A collection of libraries to help with connecting to services such as Eureka, RabbitMQ and a whole host of Observability endpoints!

DAPR

Helps developers build event-driven, resilient distributed applications

Project Tye

A developer tool that makes developing, testing, and deploying microservices and distributed applications easier

YARP

"Yet Another Reverse Proxy" project developed by some folks from Microsoft

Thank You!

@LaylaCodesIt

@Layla-P

@LaylaCodesIt

laylap@vmware.com

VMware.com

laylacodes.it

Tacky Tacos

Slides

Thanks for joining!

Enjoy the rest of the conference!