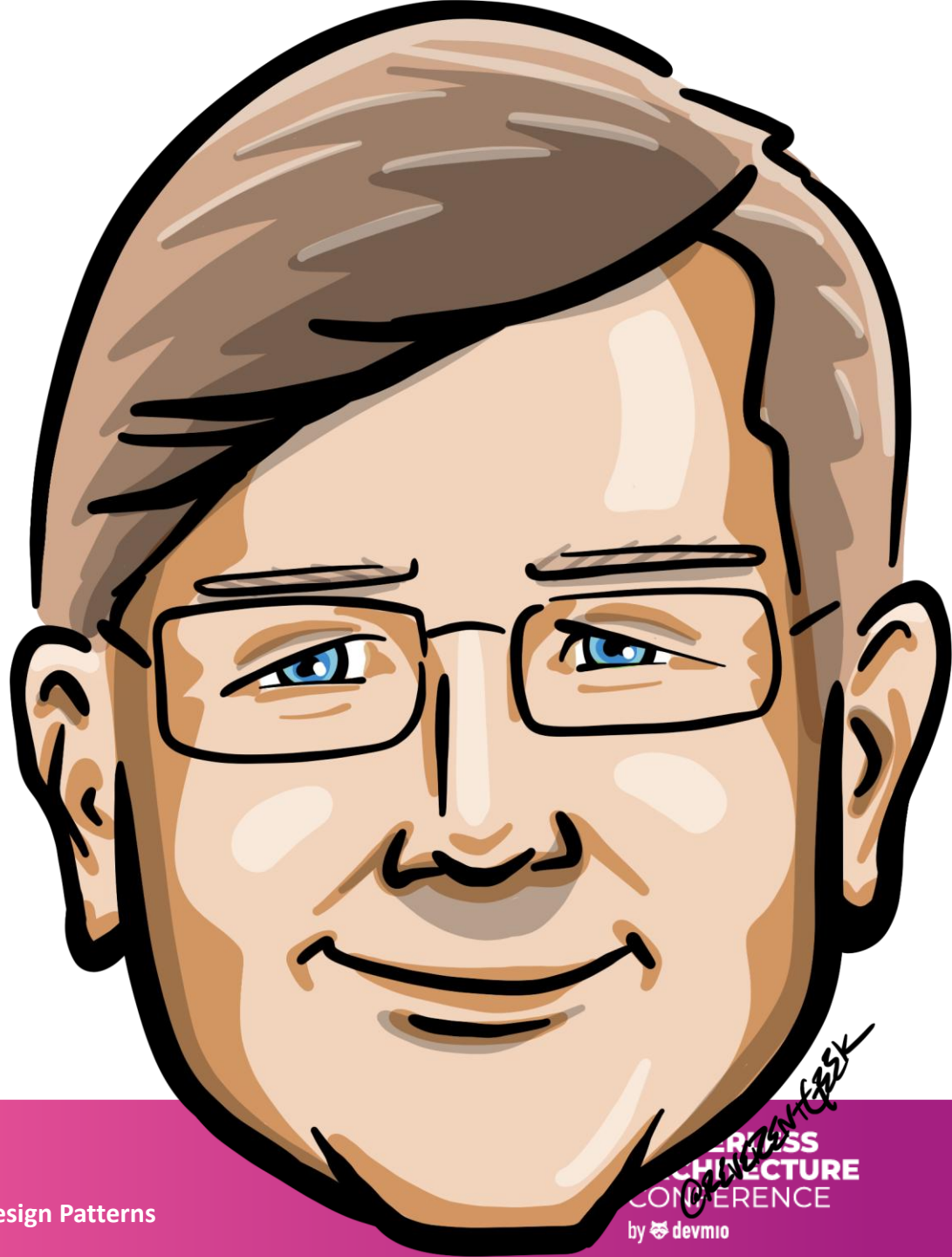
Architect Like a Boss

Mastering Serverless Design Patterns

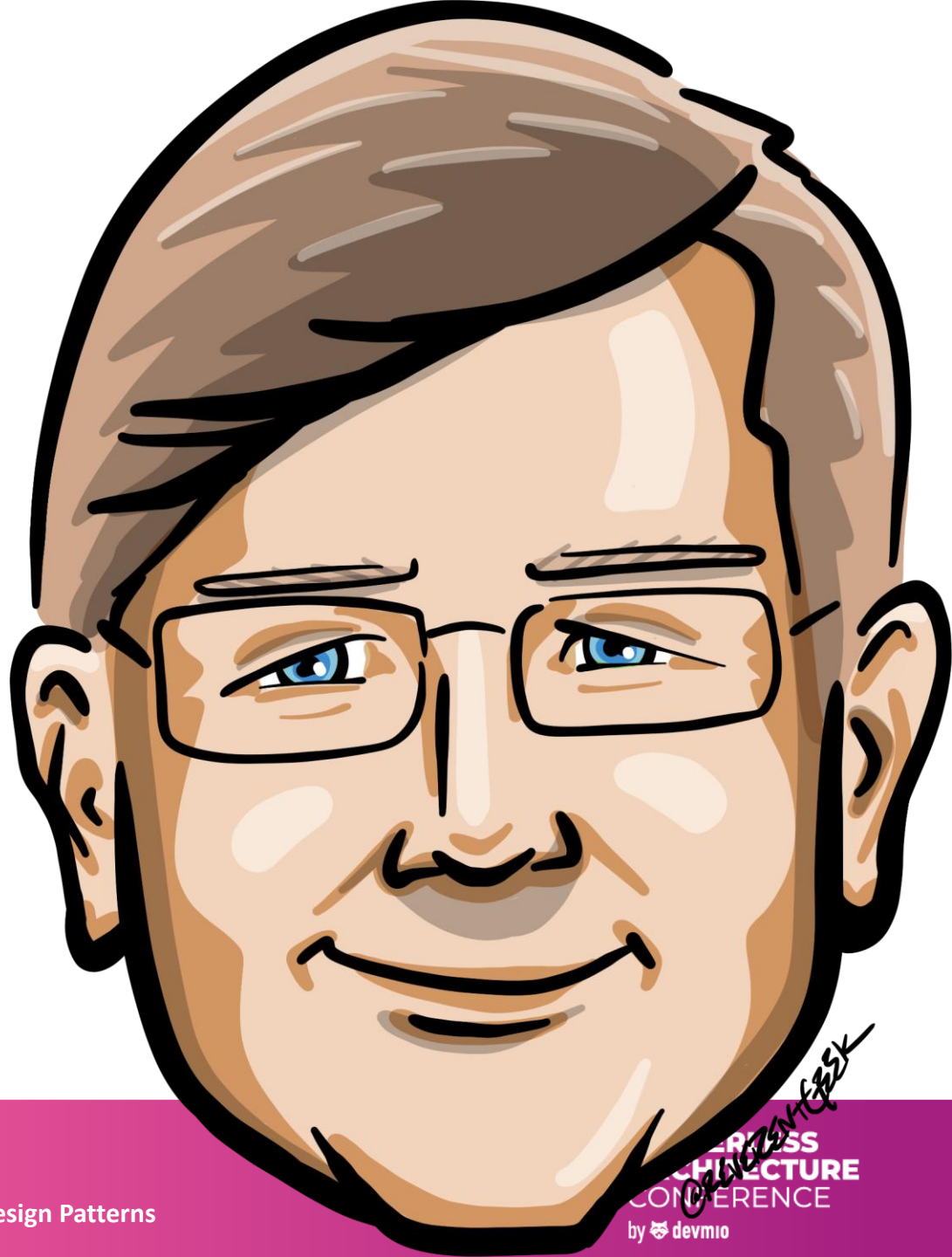
Who is Chad Green?

JASPER GO!
ENGINES & TRANSMISSIONS



Who is Chad Green?

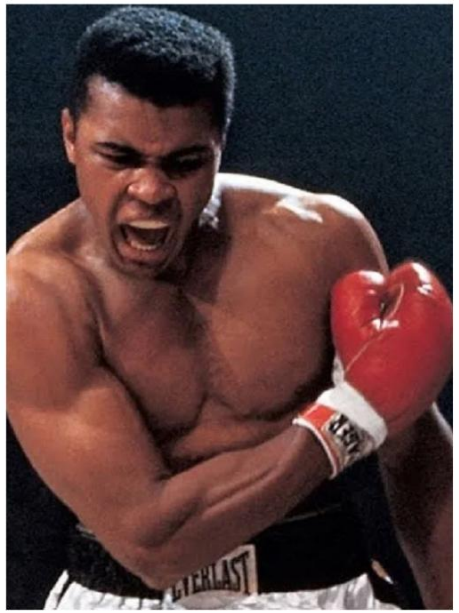
JASPER GO!
ENGINES & TRANSMISSIONS

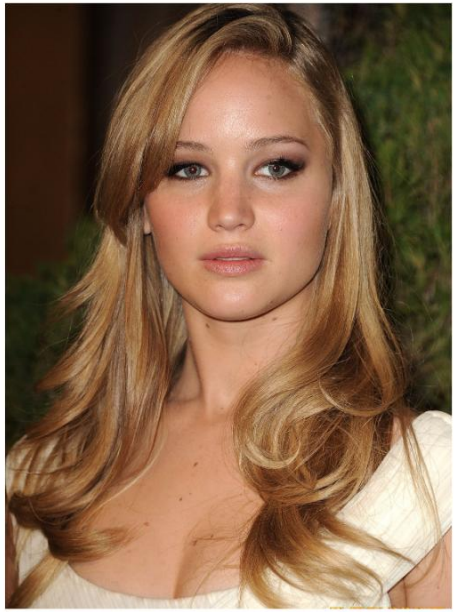


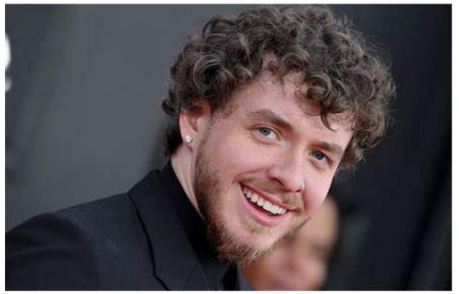
My hometown

KENTUCKY
Bourbon
TRAIL

UNITED STATES
Louisville
Ottawa
Gulf of Maine
Gulf of Mexico
Nassau
Havana
Mexico City
Sargasso Sea







Grew up in software development

Lots of Programming Languages

BASIC

Microsoft BASIC
GW-Basic
QuickBASIC
Visual Basic
VB.NET

C

C
C++
C#
Java

Niche

JavaScript
Pascal
Python
PHP
COBOL

Software development paradigms

Console Applications
Desktop Applications
Distributed Desktop Applications
Web Applications
Distributed Web Applications

COM
DCMO
.NET
Entity Framework
MVC
Service-Oriented Architecture
Microservices
Cloud-Computing

Serverless Computing

Serverless Computing

Functions-as-a-Service

Event-Driven Architecture

Microservices

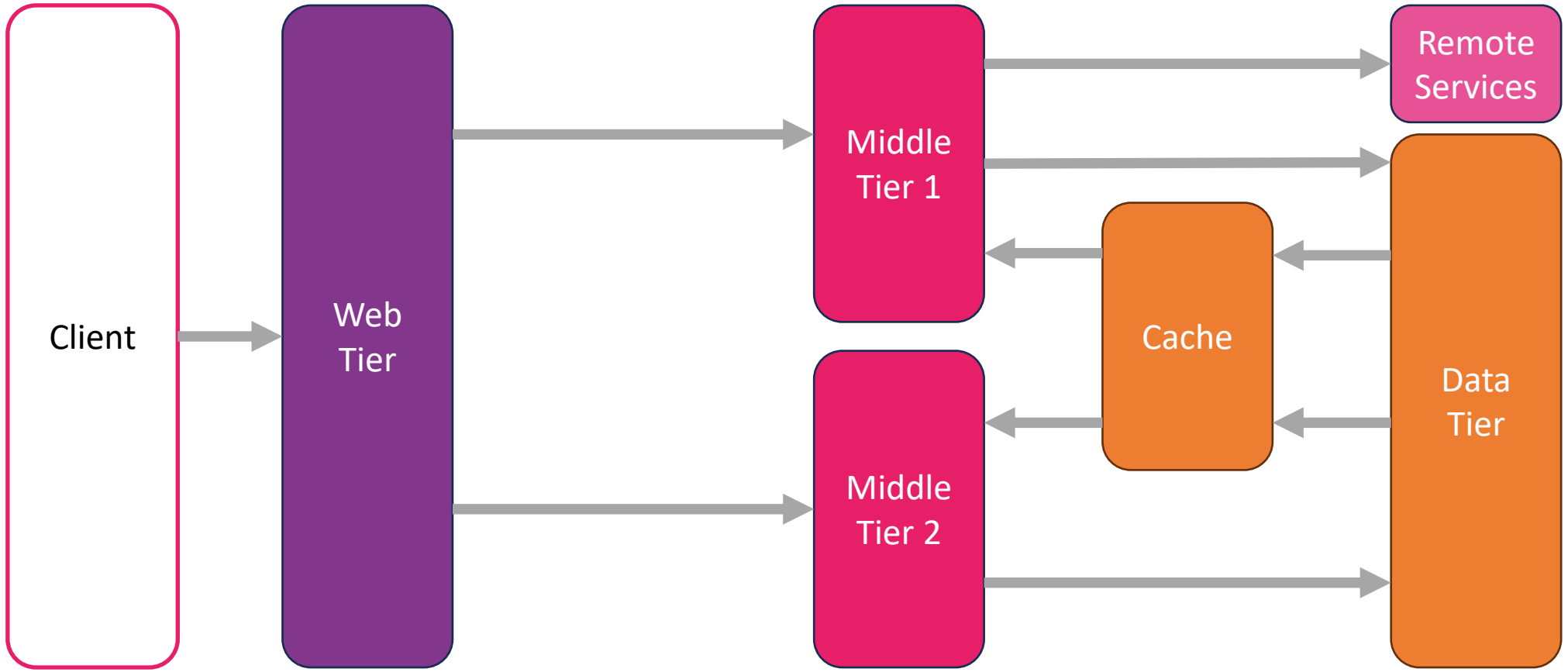
Work Experience

Disclaimer

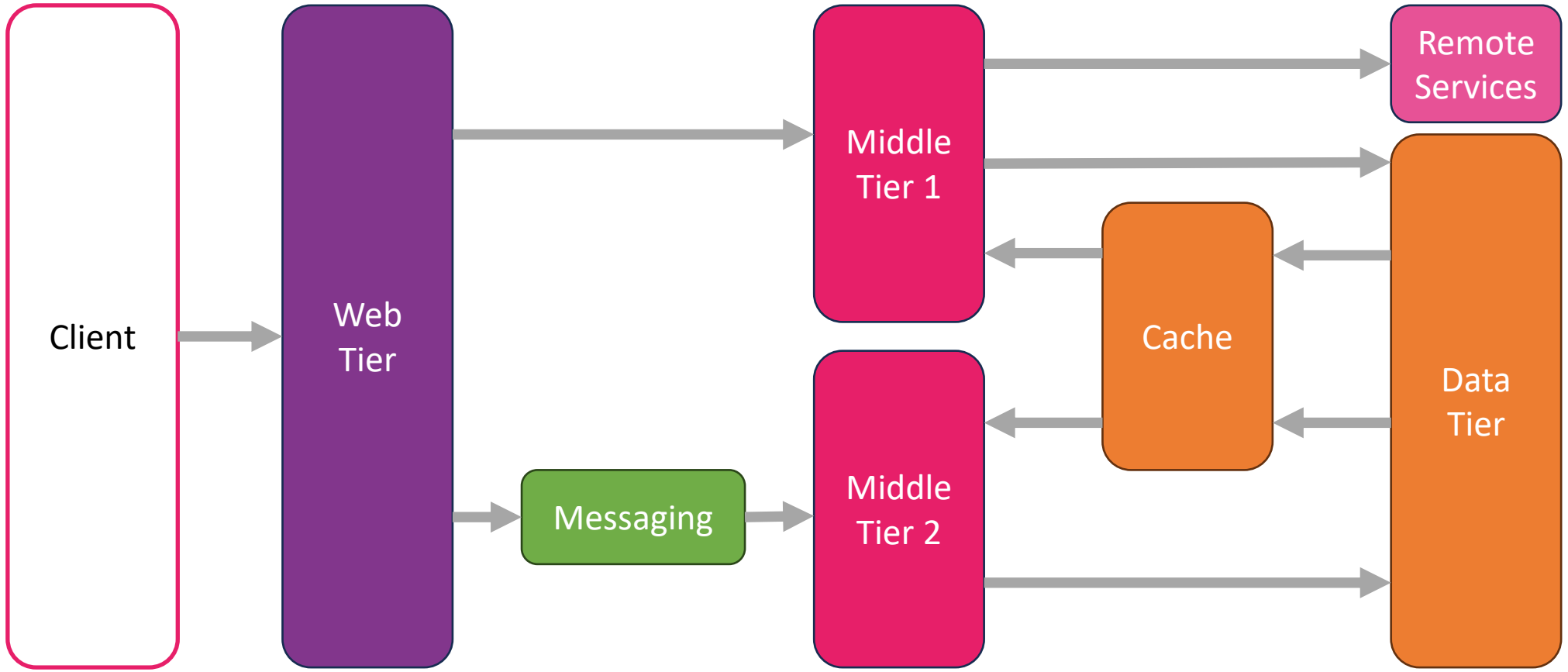
Architecture Styles

Crack the Code to Scalability

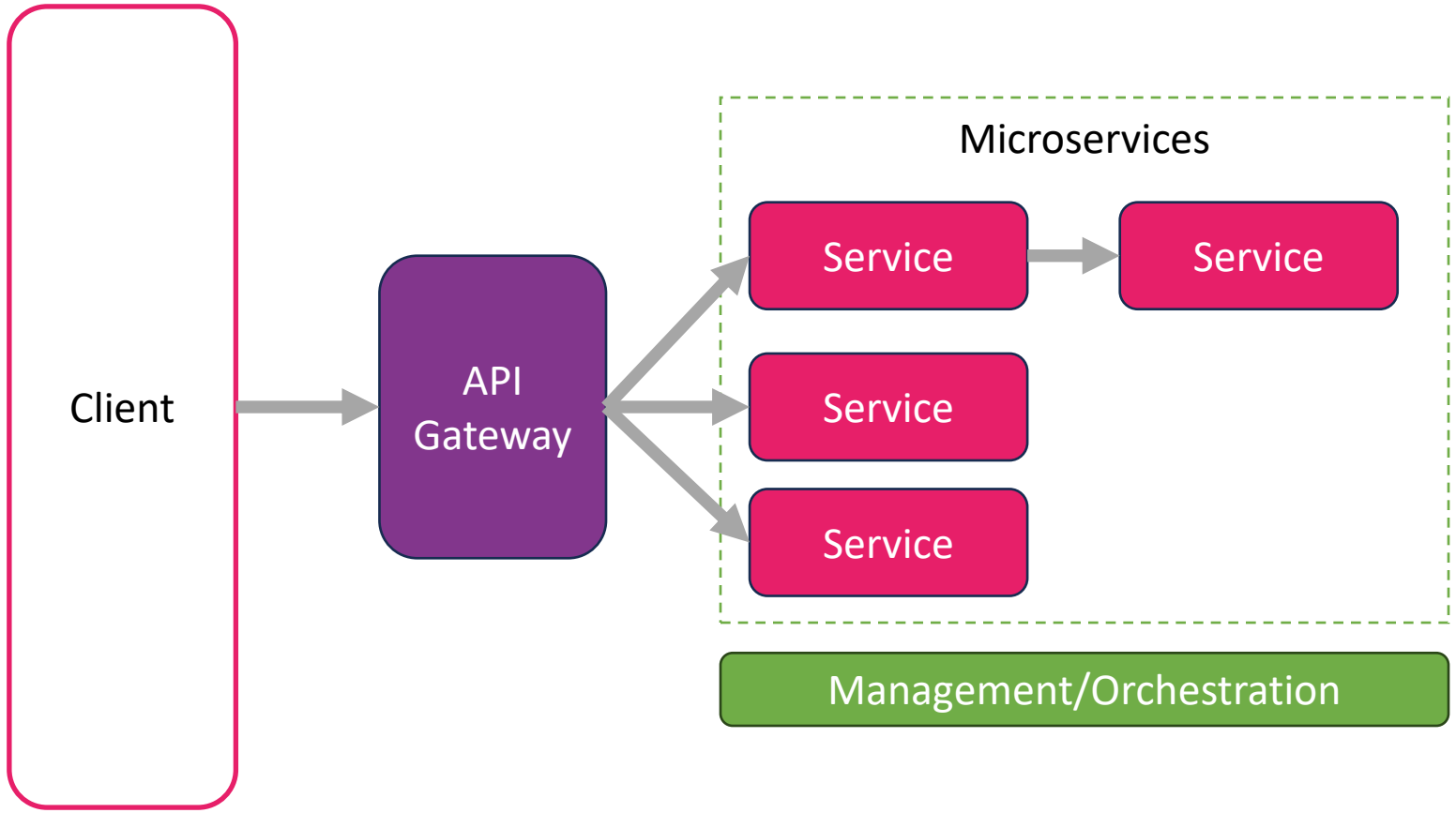
N-Tier


Client
Web Tier
Middle Tier 1
Middle Tier 2
Remote Services
Cache
Data Tier

N-Tier with Messaging



Microservices

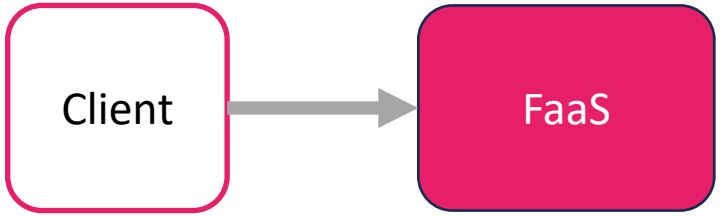


Event-Driven Architecture

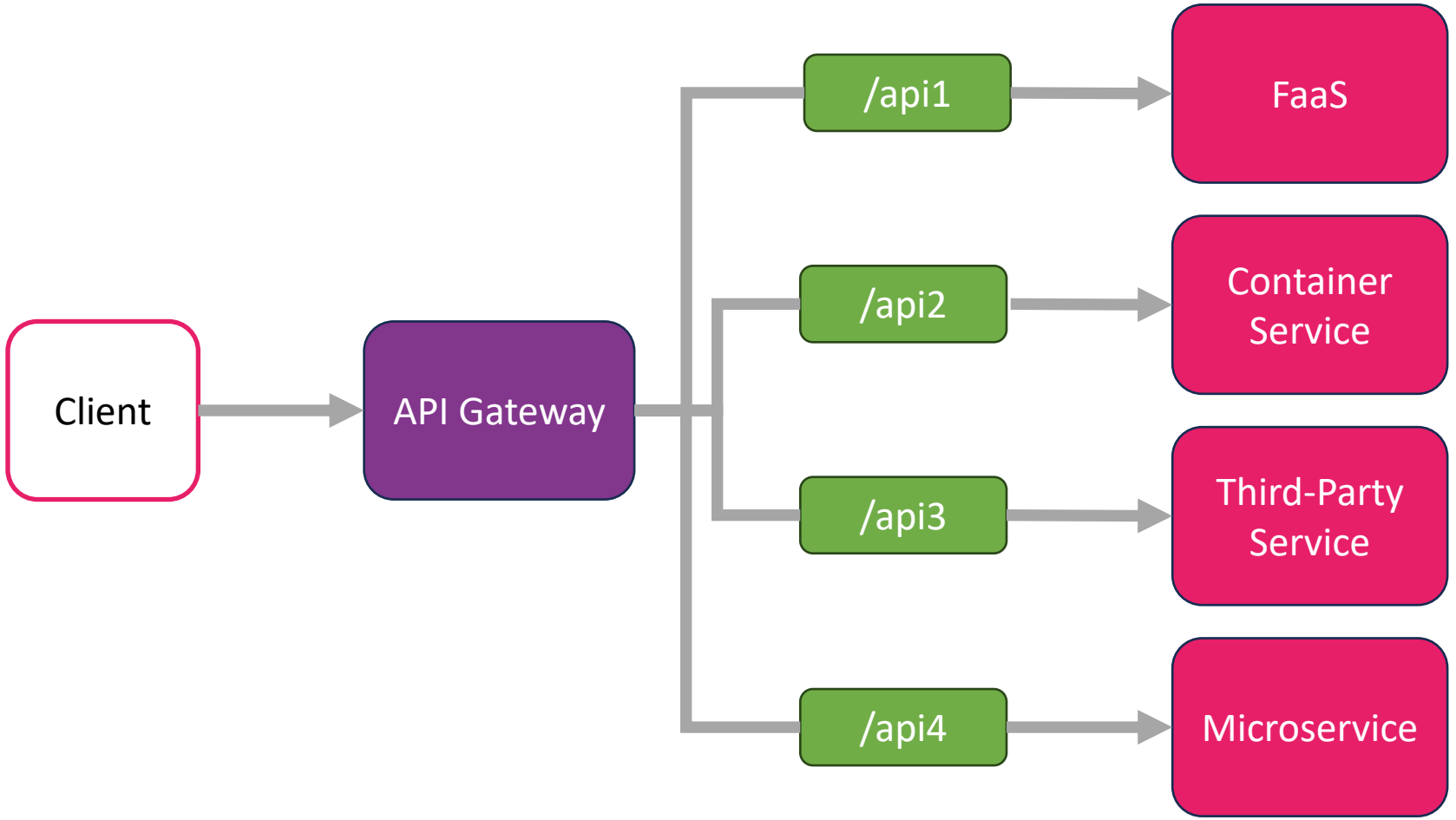
Serverless Design Patterns

Crack the Code to Scalability

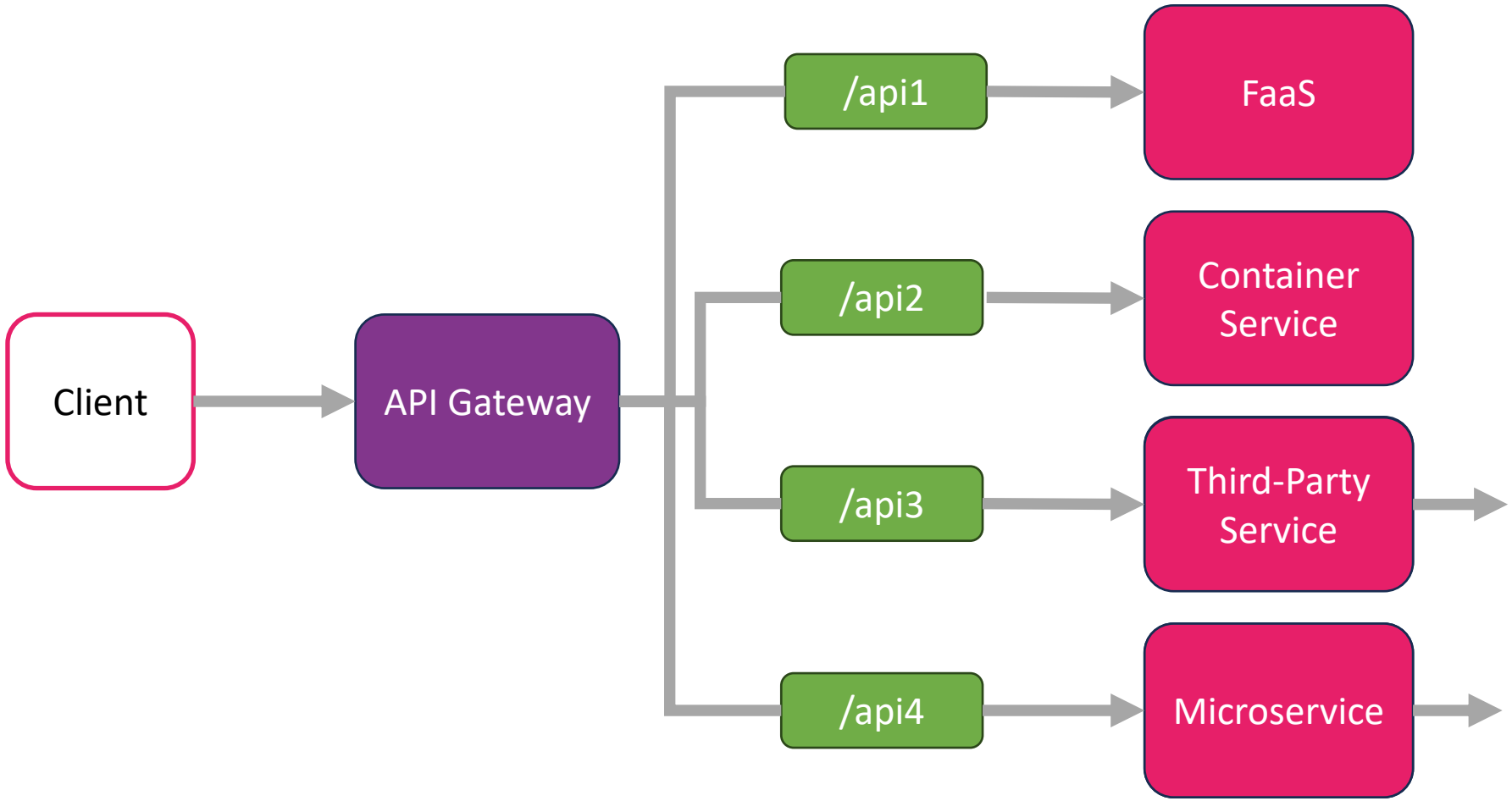
Simple Web Service



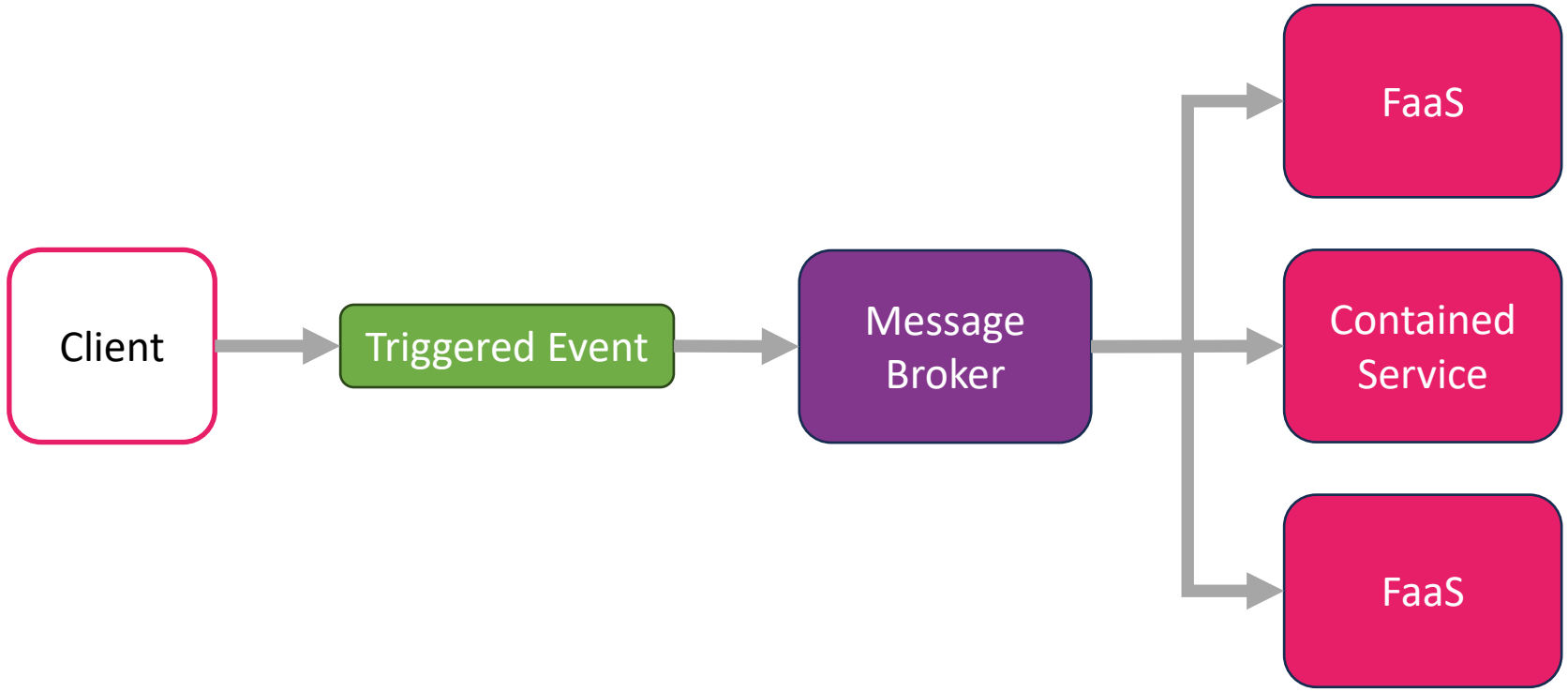
Robust API


Client
API Gateway
/api1
/api2
/api3
/api4
FaaS
Container Service
Third-Party Service
Microservice

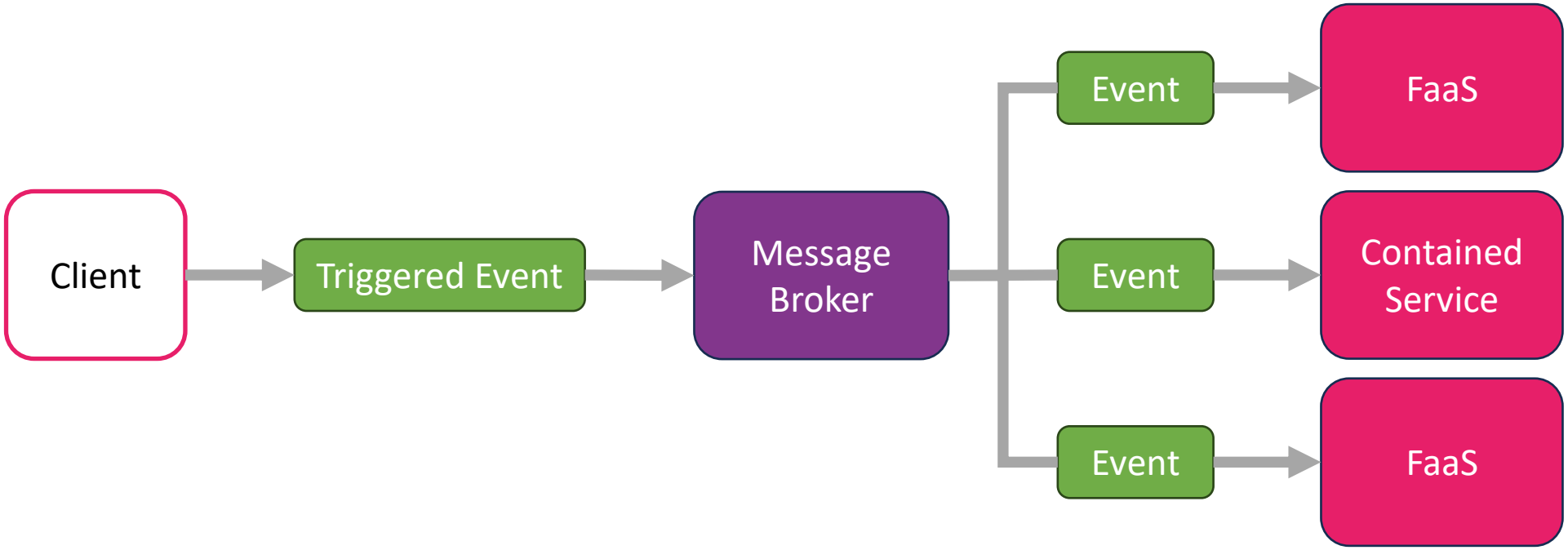
Robust API with Aggregation



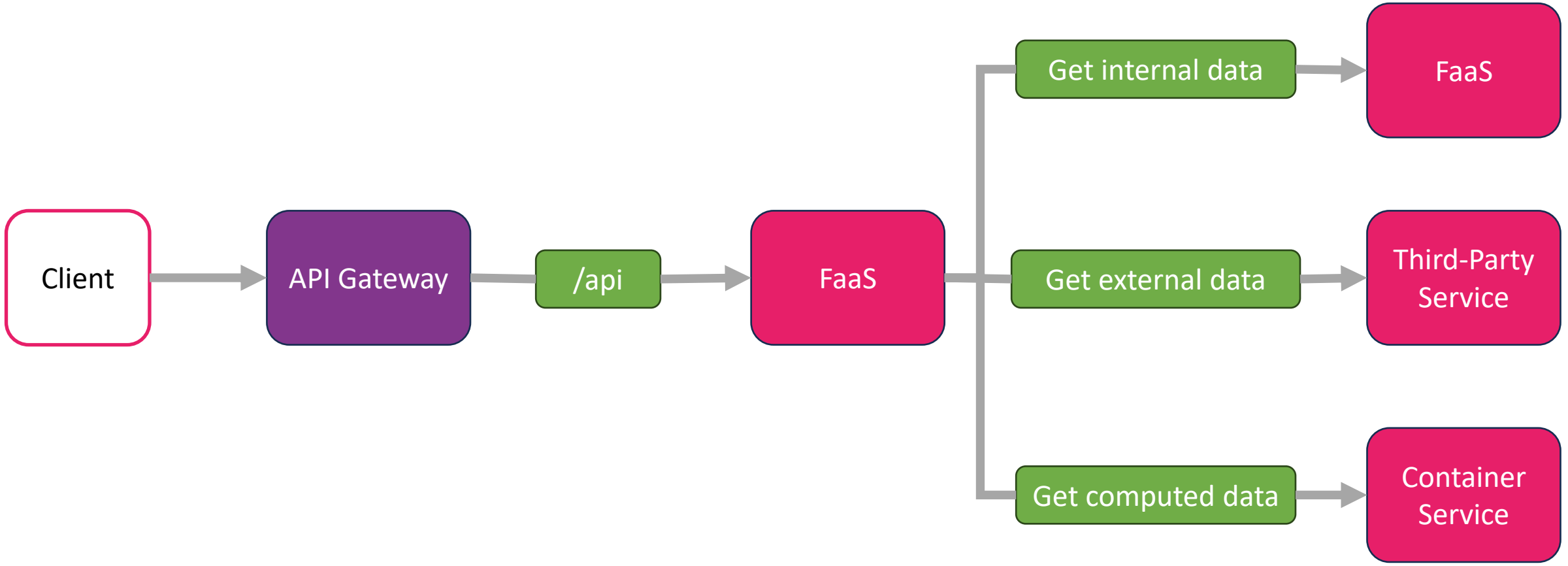
Decoupled Messaging



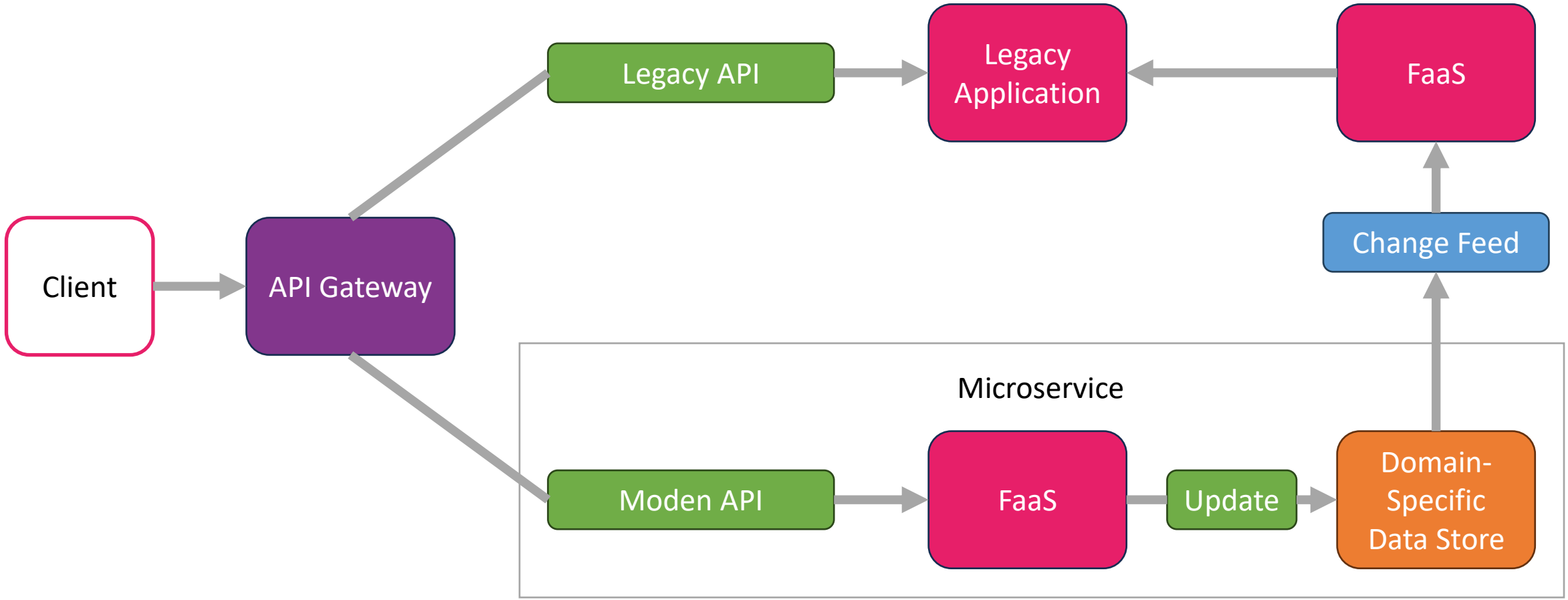
Publish/Subscribe



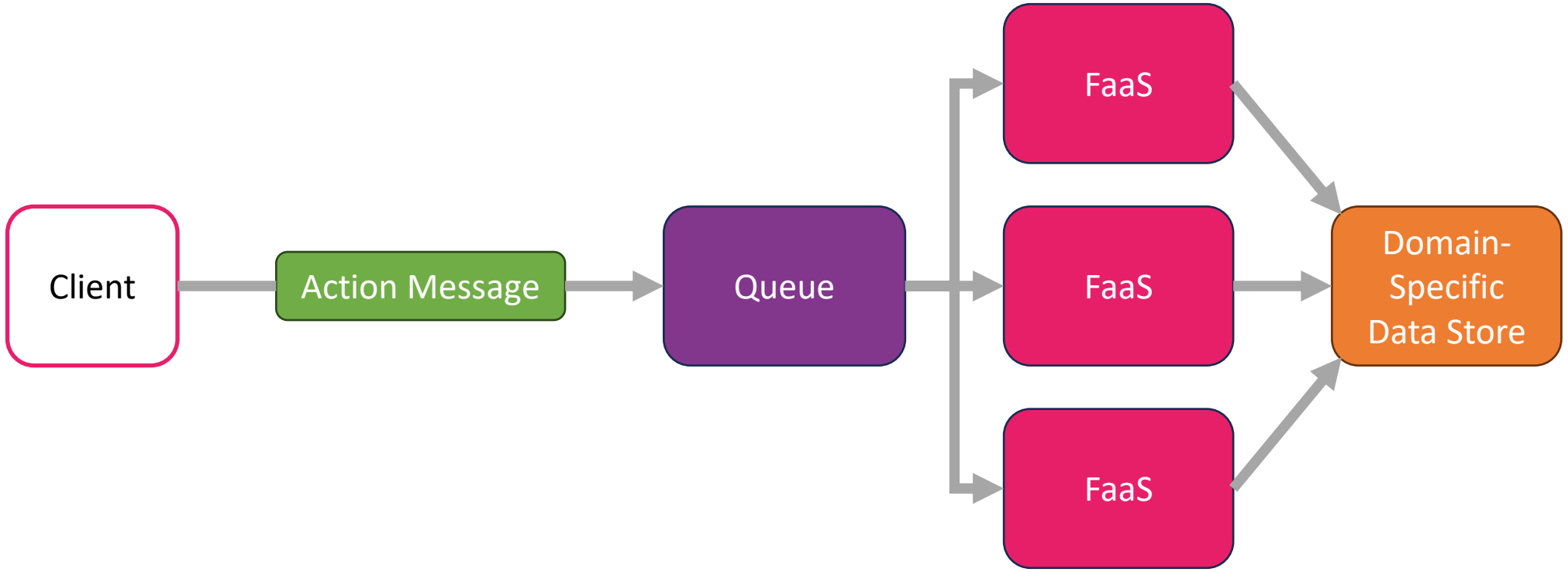
Aggregation



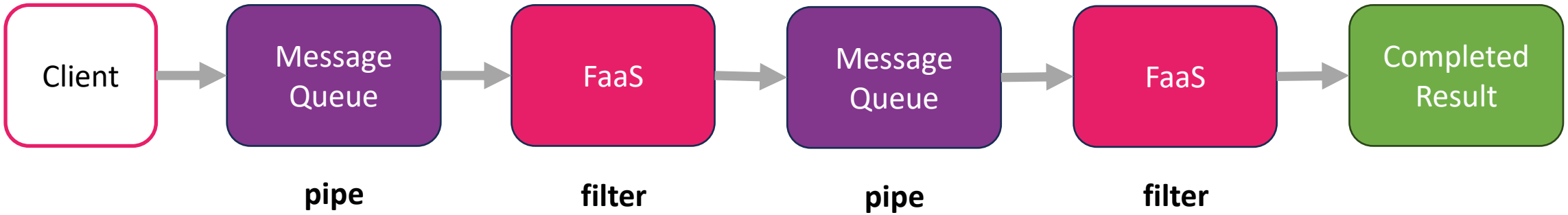
Strangler


Client
API Gateway
Legacy API
Legacy Application
FaaS
Change Feed
Microservice
Moden API
FaaS
Update
Domain-Specific Data Store

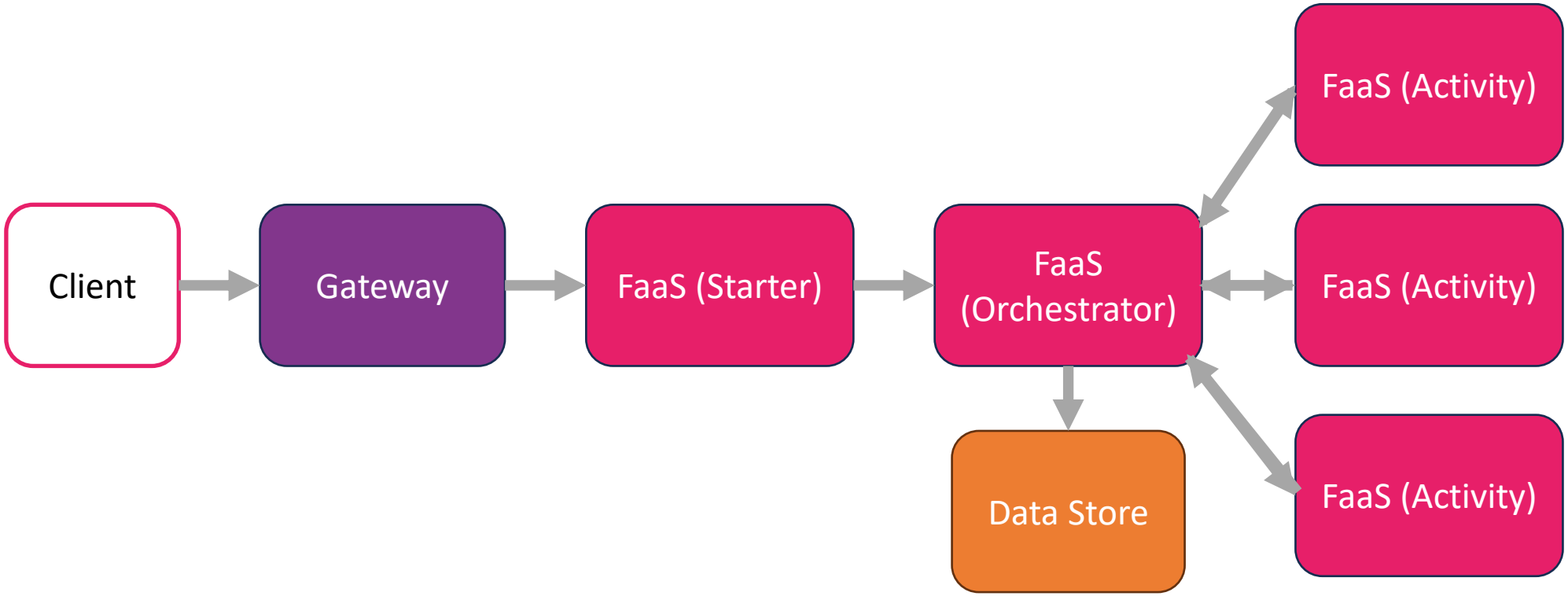
Queue-Based Load Leveling



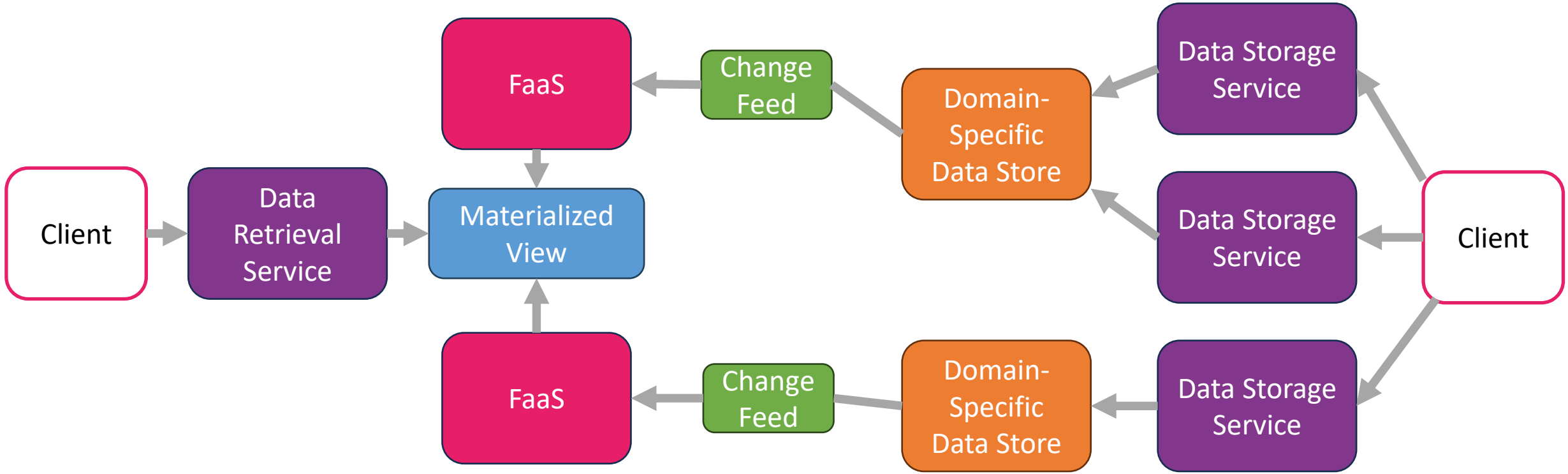
Pipes and Filters



Fan-Out/Fan-In



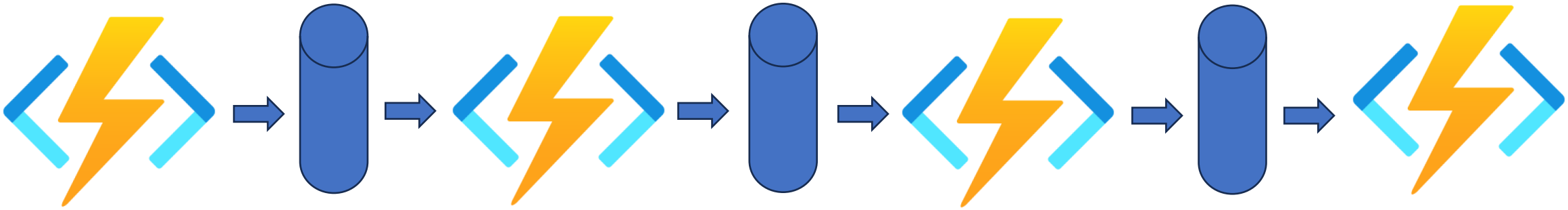
Materialized Views



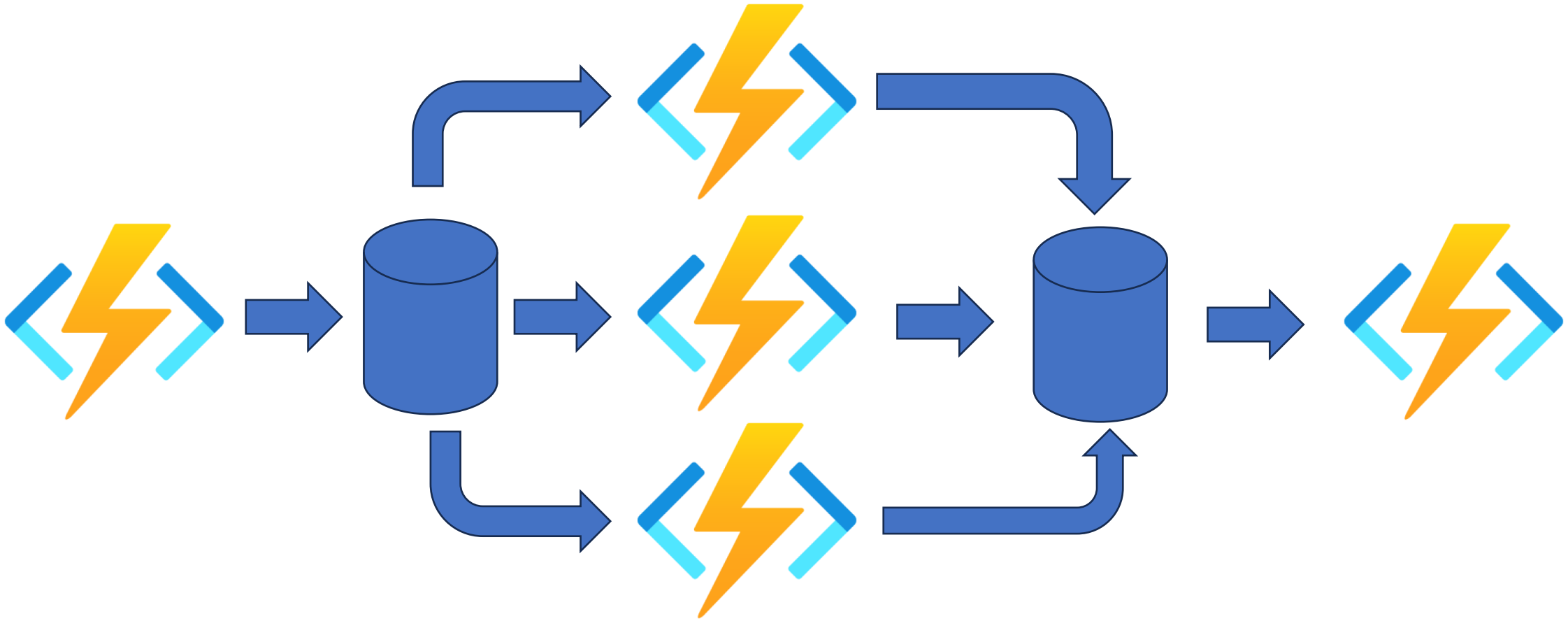
Orchestration

Crack the Code to Scalability

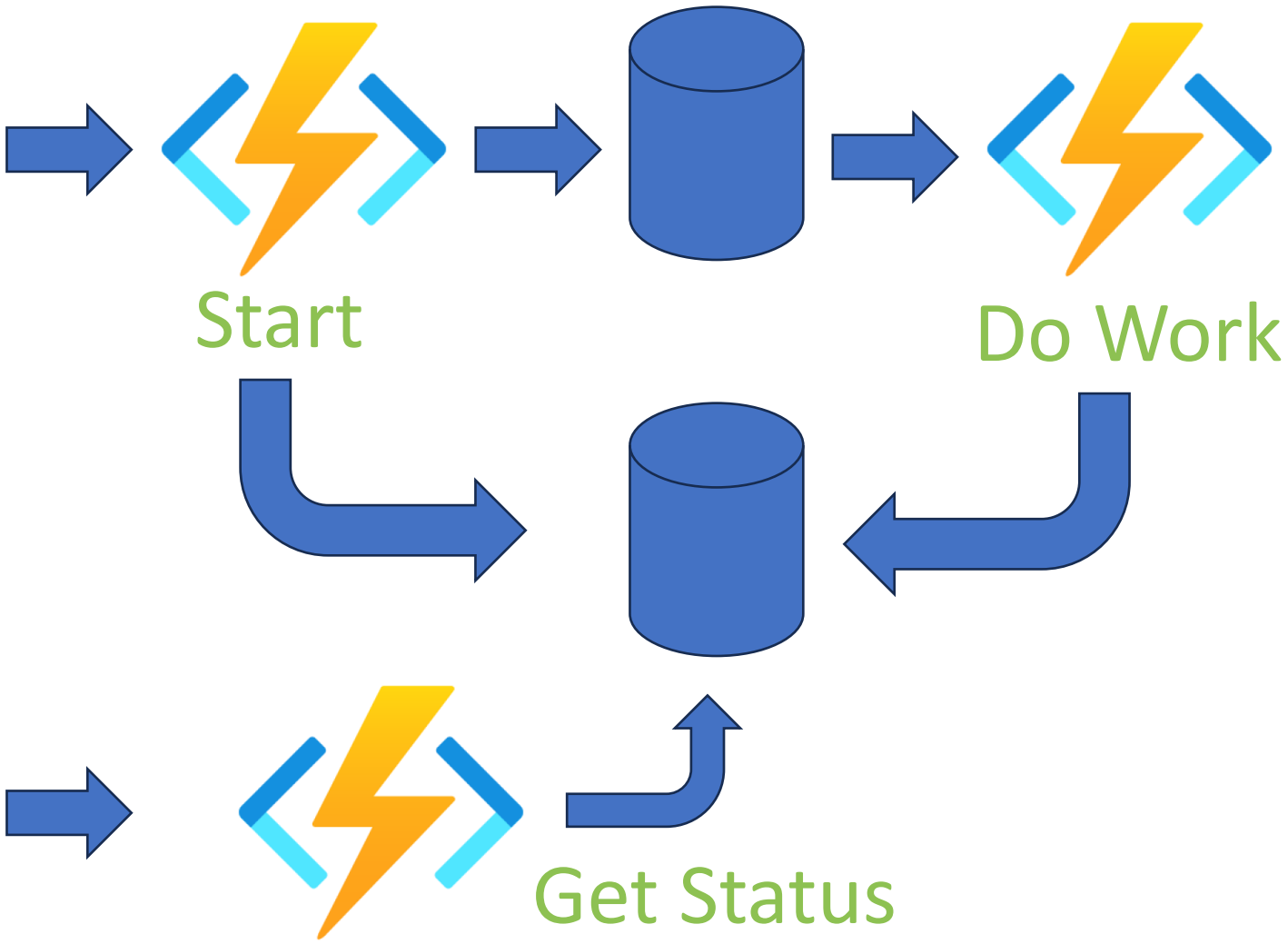
Function Chaining



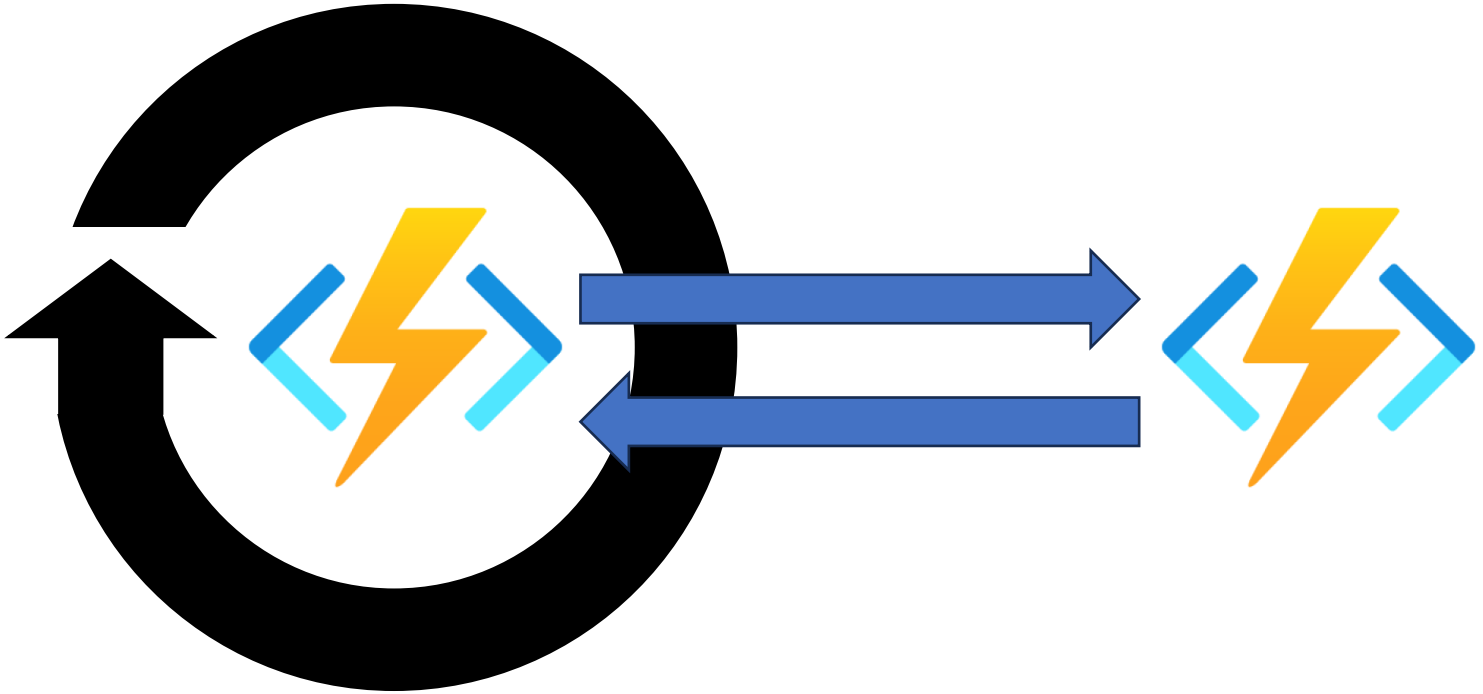
Fan Out / Fan In



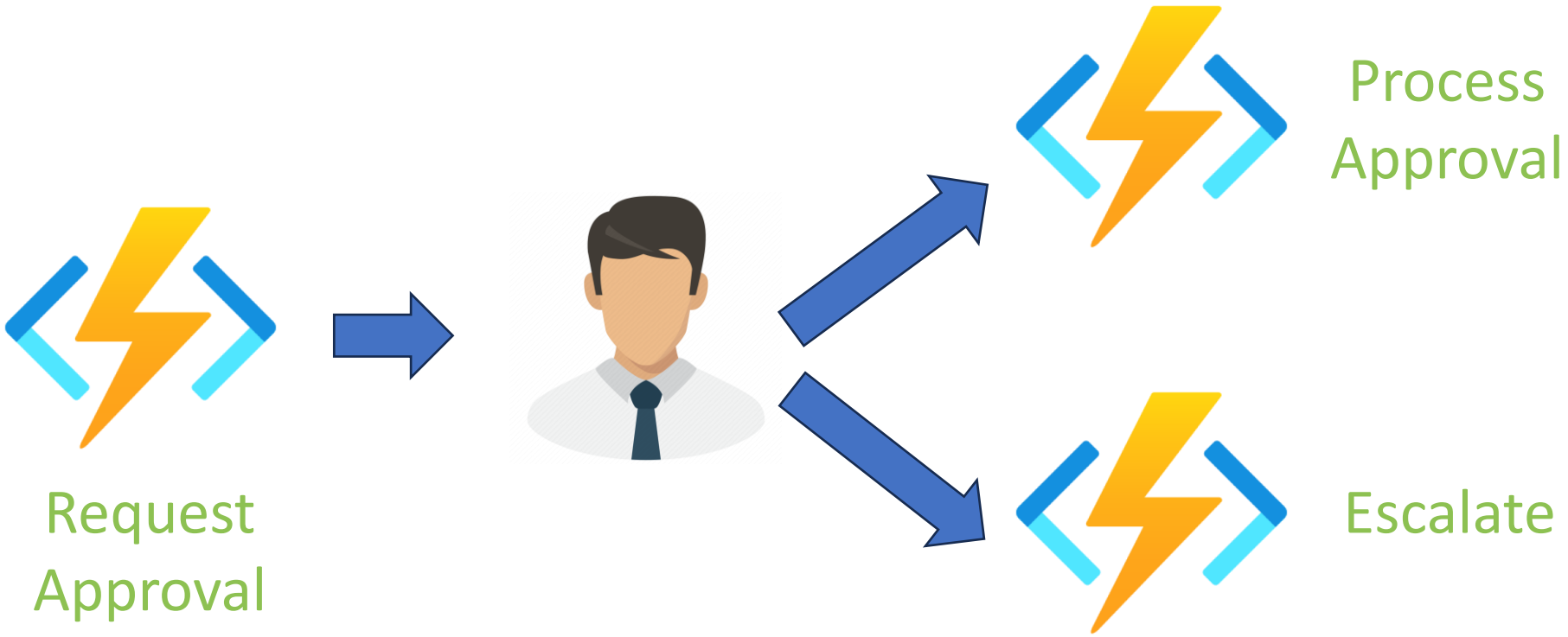
Async HTTP APIs



Monitor



Human Interaction



Serverless Best Practices

Crack the Code to Scalability

Serverless Best Practices

Stay Focused

Serverless Best Practices

Stay Focused

Handle Errors

Serverless Best Practices

Stay Focused

Handle Errors

Observe What's Going On

Serverless Best Practices

- Stay Focused
- Handle Errors
- Observe What's Going On
- Test, Test, and Test Again

Serverless Best Practices

Stay Focused

Handle Errors

Observe What's Going On

Test, Test, and Test Again

Optimize Costs

Serverless Best Practices

Stay Focused

Handle Errors

Observe What's Going On

Test, Test, and Test Again

Optimize Costs

Engage Shields

Serverless Best Practices

Stay Focused

Handle Errors

Observe What's Going On

Test, Test, and Test Again

Optimize Costs

Engage Shields

Log and Trace

Serverless Best Practices

- Stay Focused
- Handle Errors
- Observe What's Going On
- Test, Test, and Test Again
- Optimize Costs
- Engage Shields
- Log and Trace
- Versioning and Deployment

Serverless Best Practices

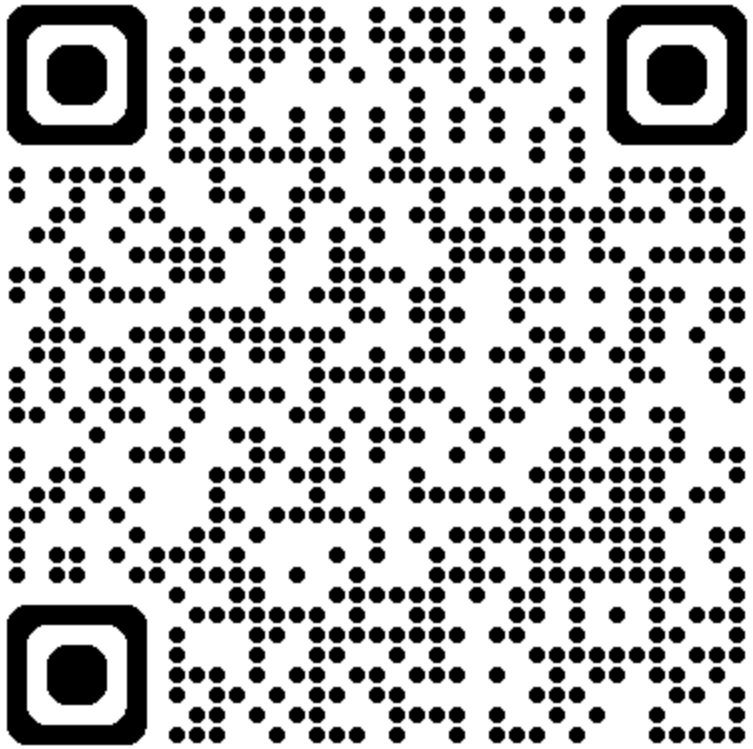
- Stay Focused
- Handle Errors
- Observe What's Going On
- Test, Test, and Test Again
- Optimize Costs
- Engage Shields
- Log and Trace
- Versioning and Deployment

Thank You

chadgreen@chadgreen.com

TaleLearnCode

ChadGreen.com

ChadGreen

ChadwickEGreen