



Microservices Develop + Test + Deploy



Somkiat Puisungnoen

Search

Somkiat | Home

Update Info 1 View Activity Log 10+ ...

Timeline About Friends 3,138 Photos More

When did you work at Opendream? X

... 22 Pending Items

Post Photo/Video Live Video Life Event

What's on your mind?

Public Post

Intro

Software Craftsmanship

Software Practitioner at สยามชานาญกิจ พ.ศ. 2556

Agile Practitioner and Technical at SPRINT3r

Somkiat Puisungnoen 15 mins · Bangkok · ...

Java and Bigdata



Module 2 : Develop + Testing

Recap Microservice

Properties of Microservice

Microservice 1.0 - 4.0

How to develop Microservice ?

How to test Microservice ?

12-factors app

Workshop



Module 3 : Deploy

How to deploy Microservice ?
Continuous Integration and Delivery
Practices of Continuous Integration
Deployment strategies
Working with containerization (Docker)
Workshop



Module 2

Develop and Testing



Service Principles

Creation

Interface

Testing

Operation

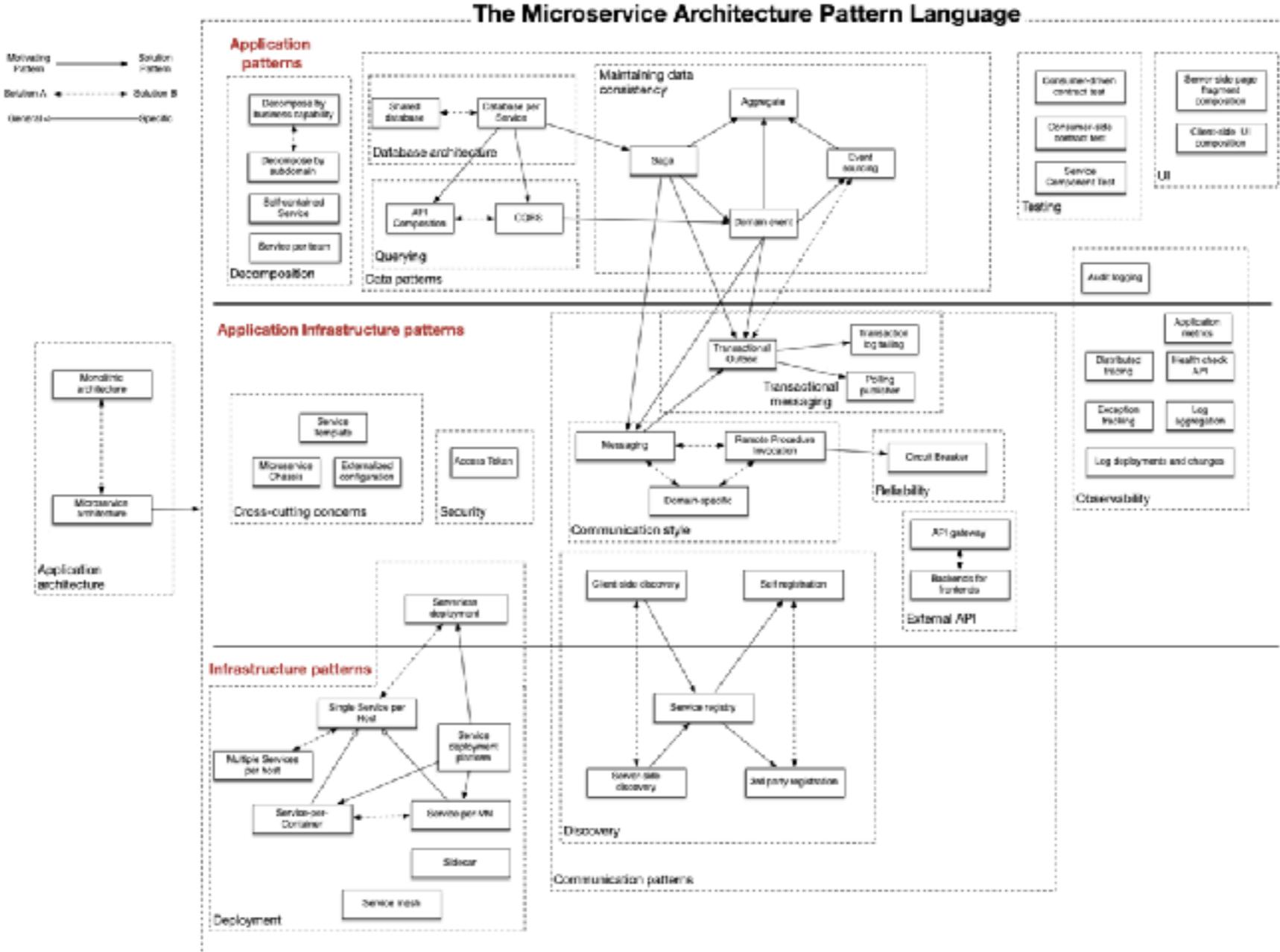
<https://github.com/Yelp/service-principles>



Microservices

© 2022 - 2023 Siam Chamnankit Company Limited. All rights reserved.

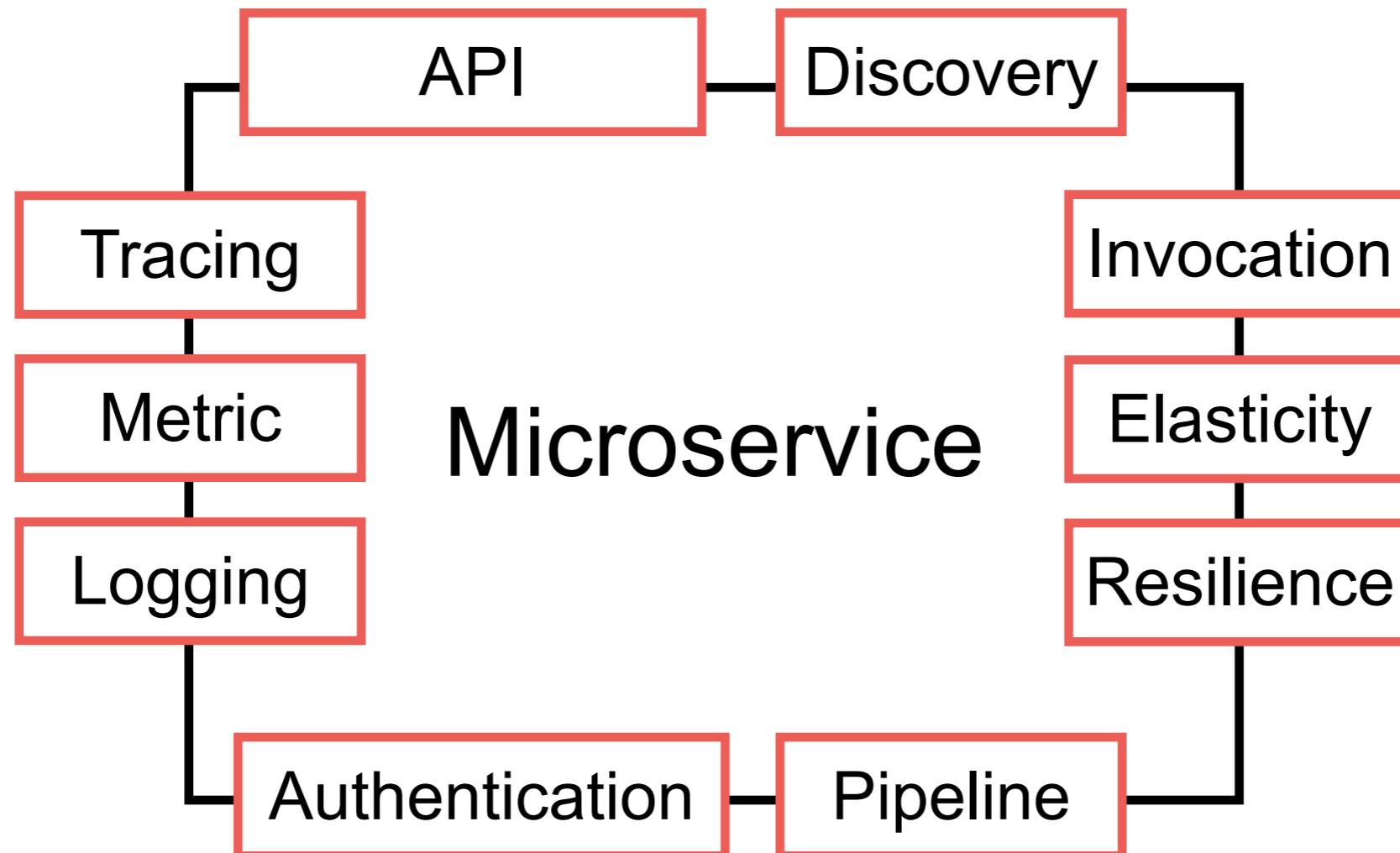
Microservice Patterns



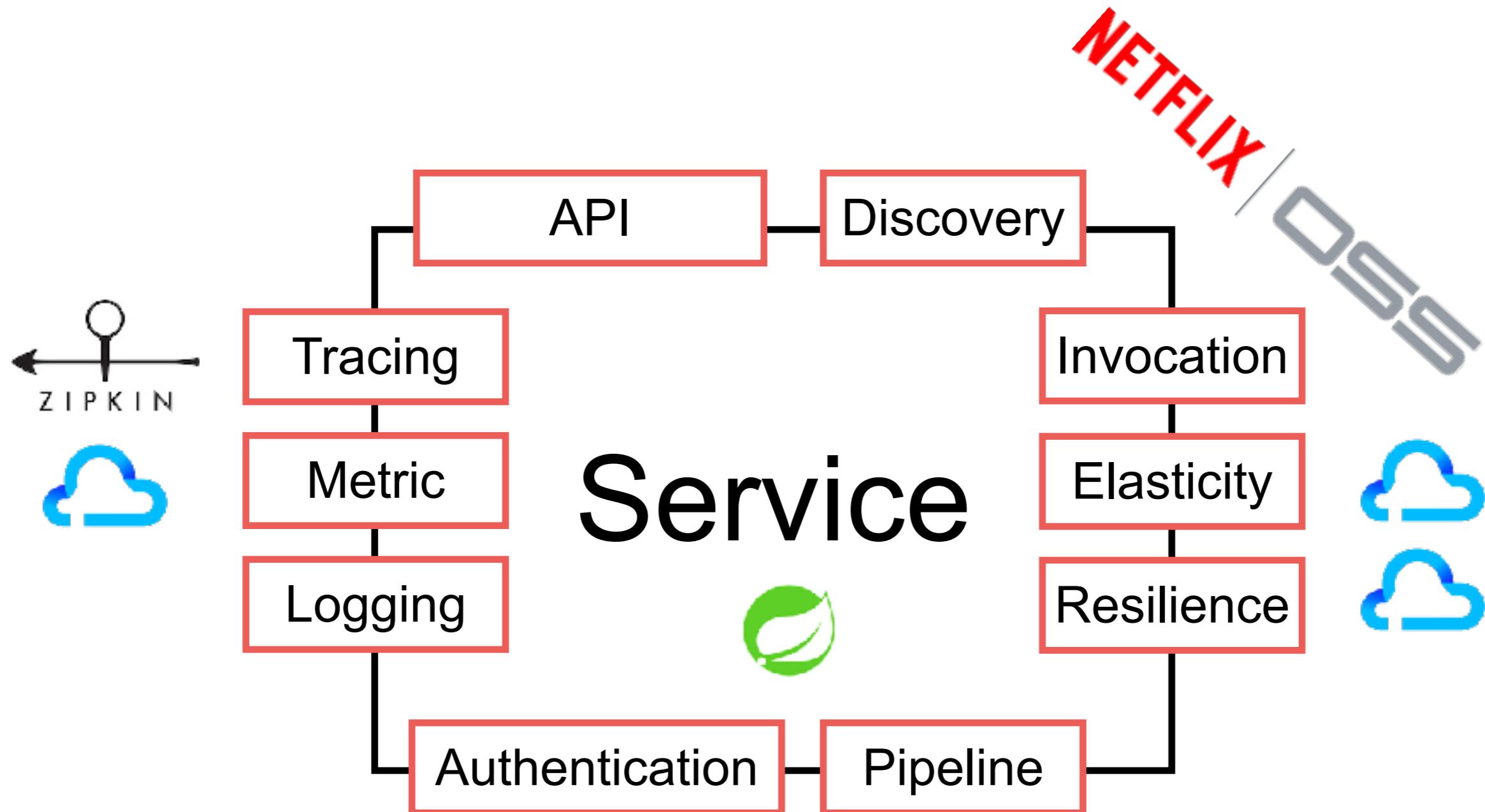
<https://microservices.io/patterns/index.html>



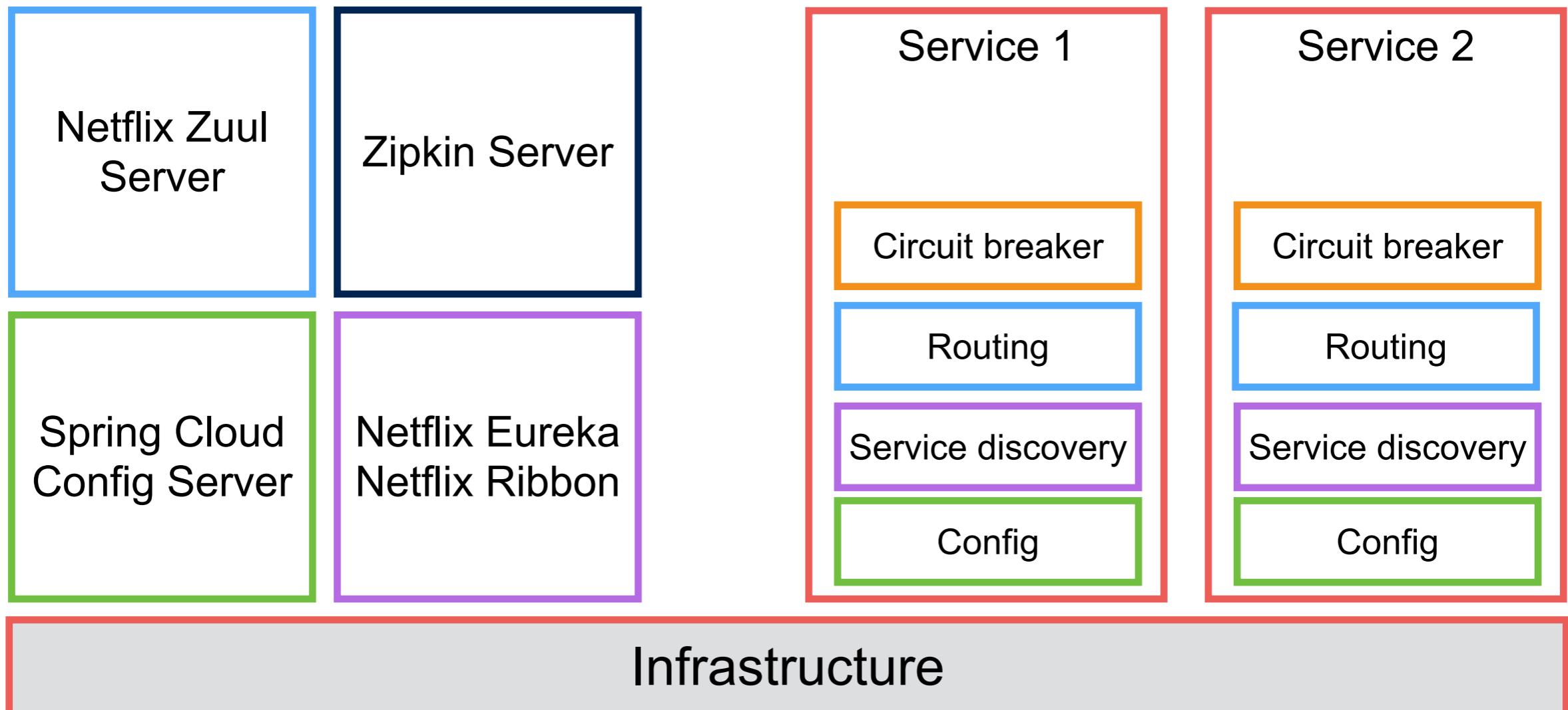
Properties of Microservice



Microservice 1.0



Tracing and Visibility



Microservice 1.0

JVM only
Add libraries to your code/service



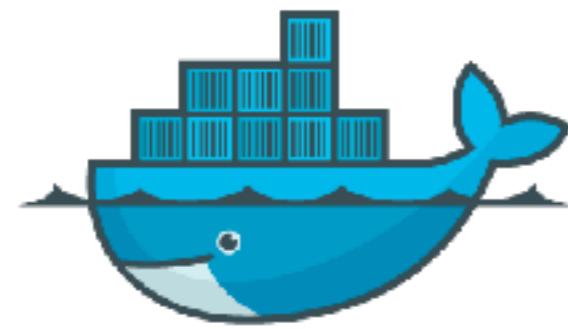
Microservice 1.5

Polyglot programming and database



Microservice 2.0

Use Container-based



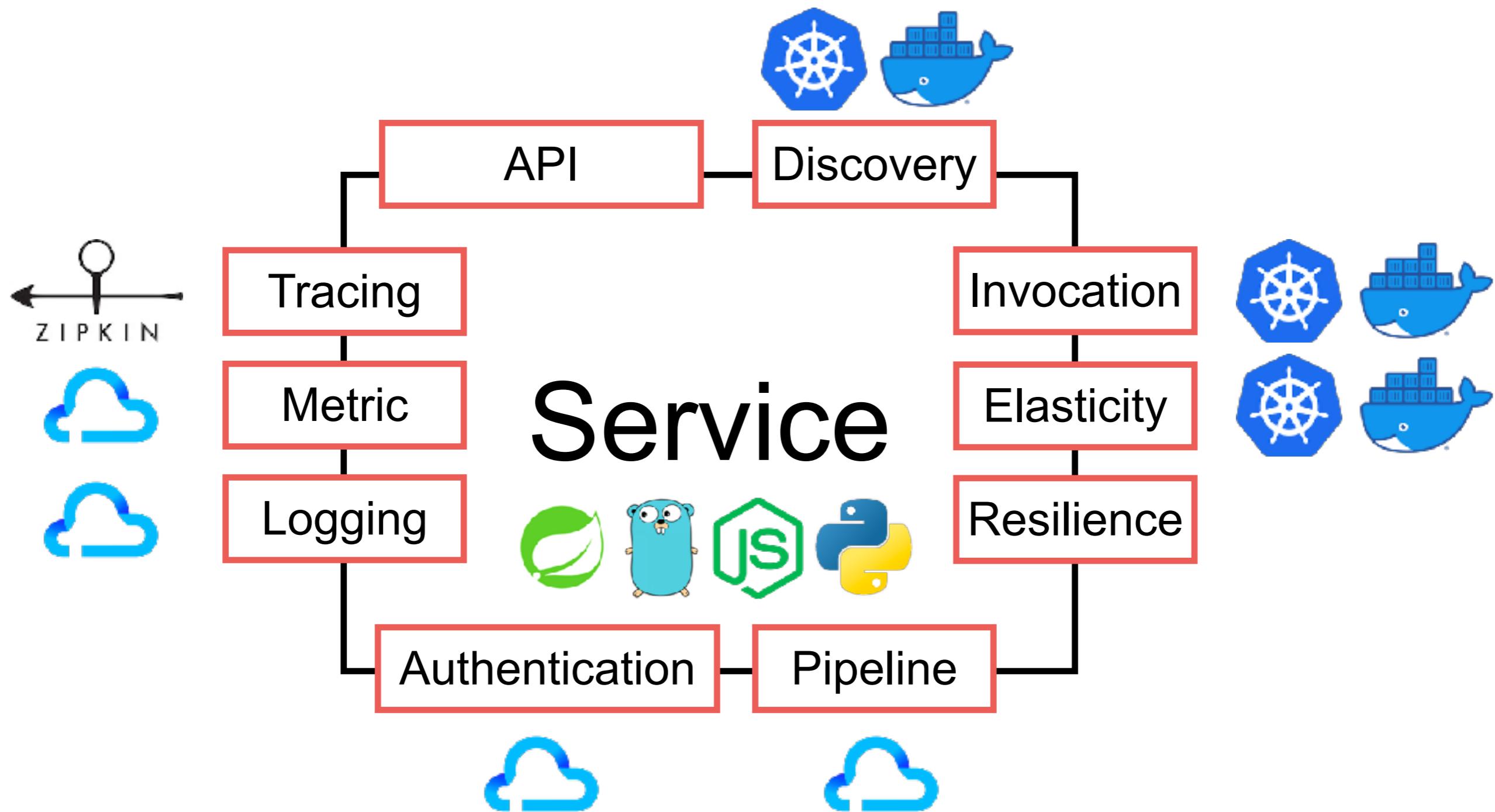
docker

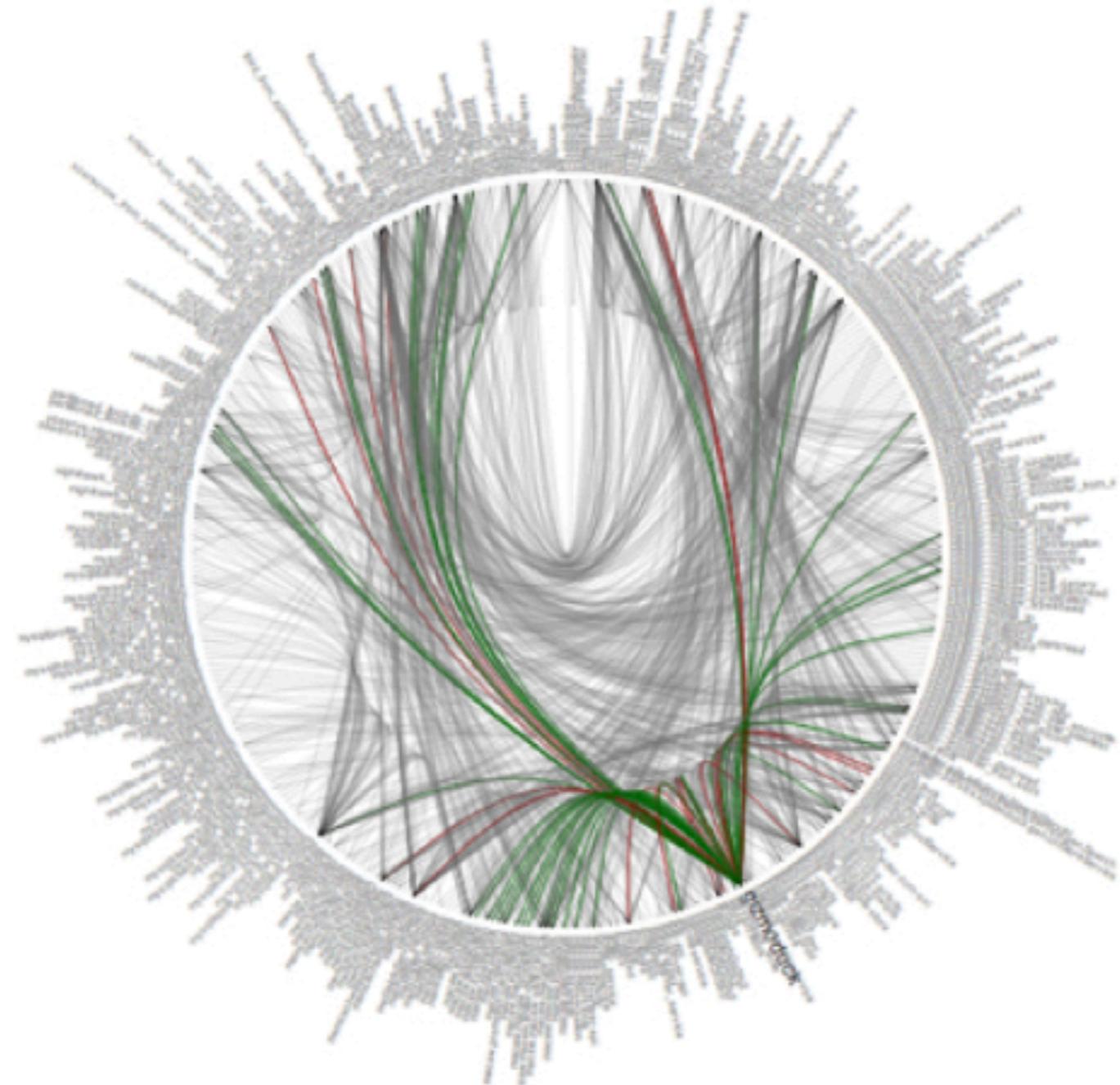


kubernetes



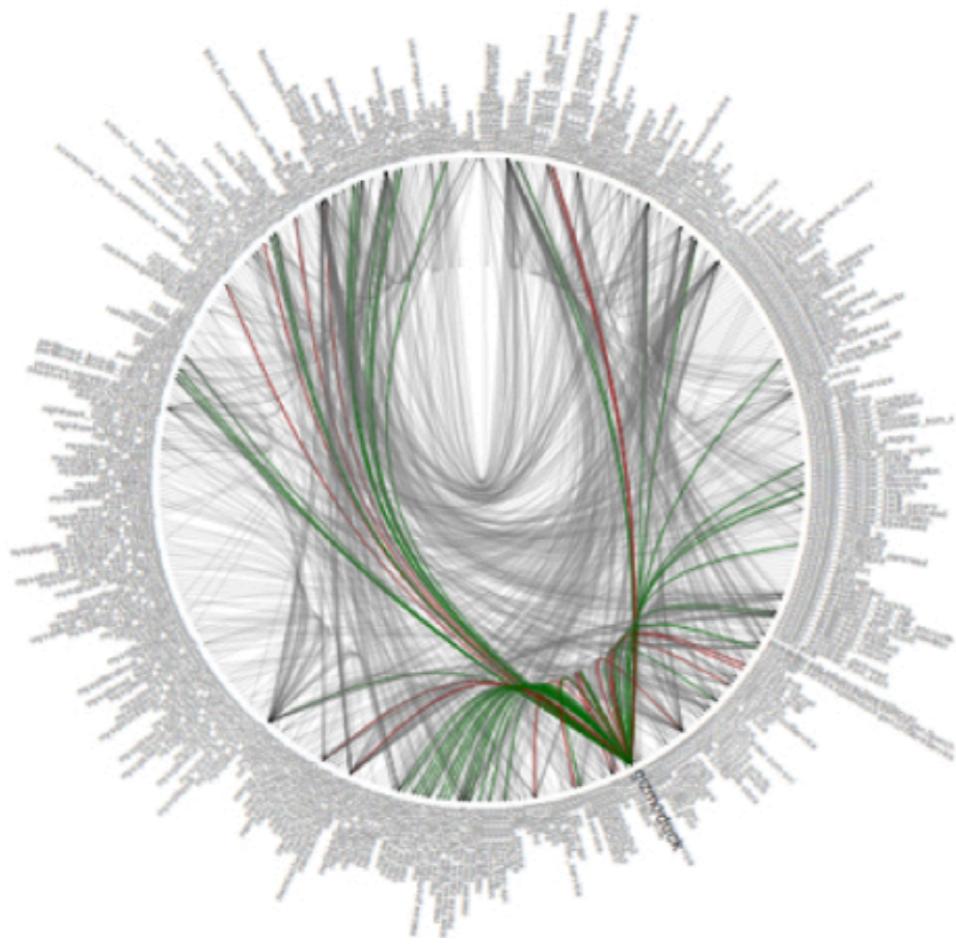
Microservice 2.0



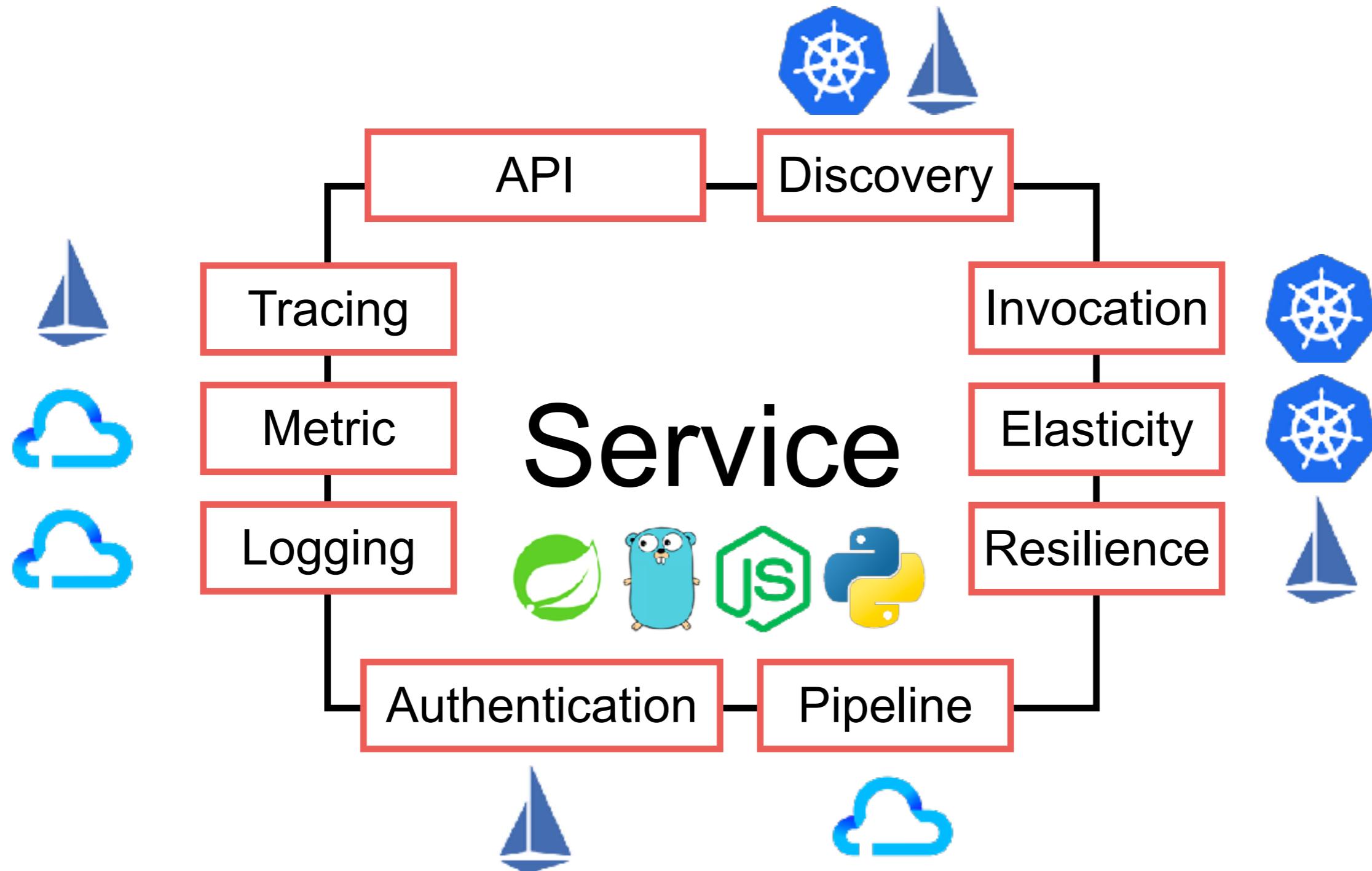


Service Mesh

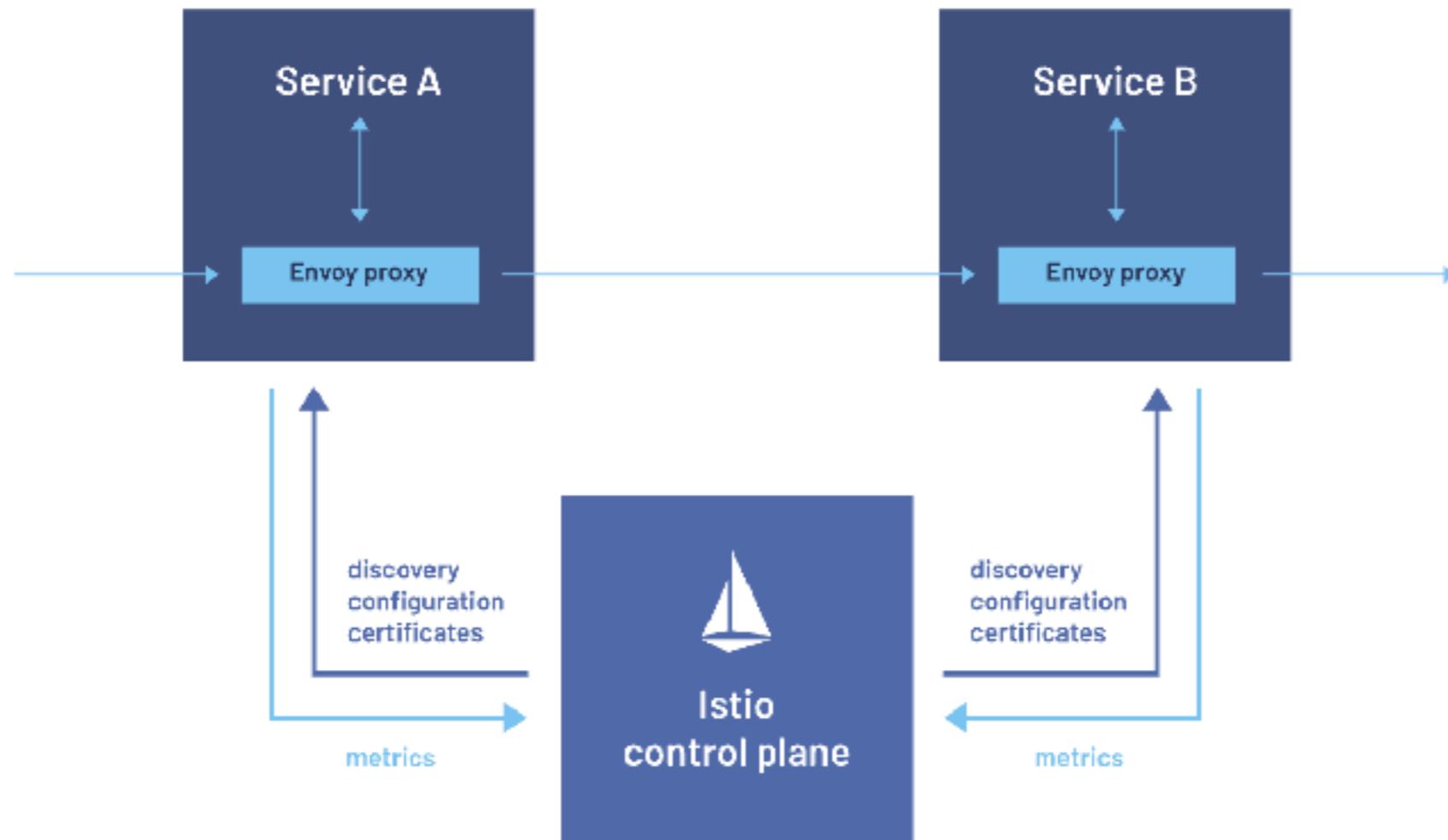




Microservice 3.0



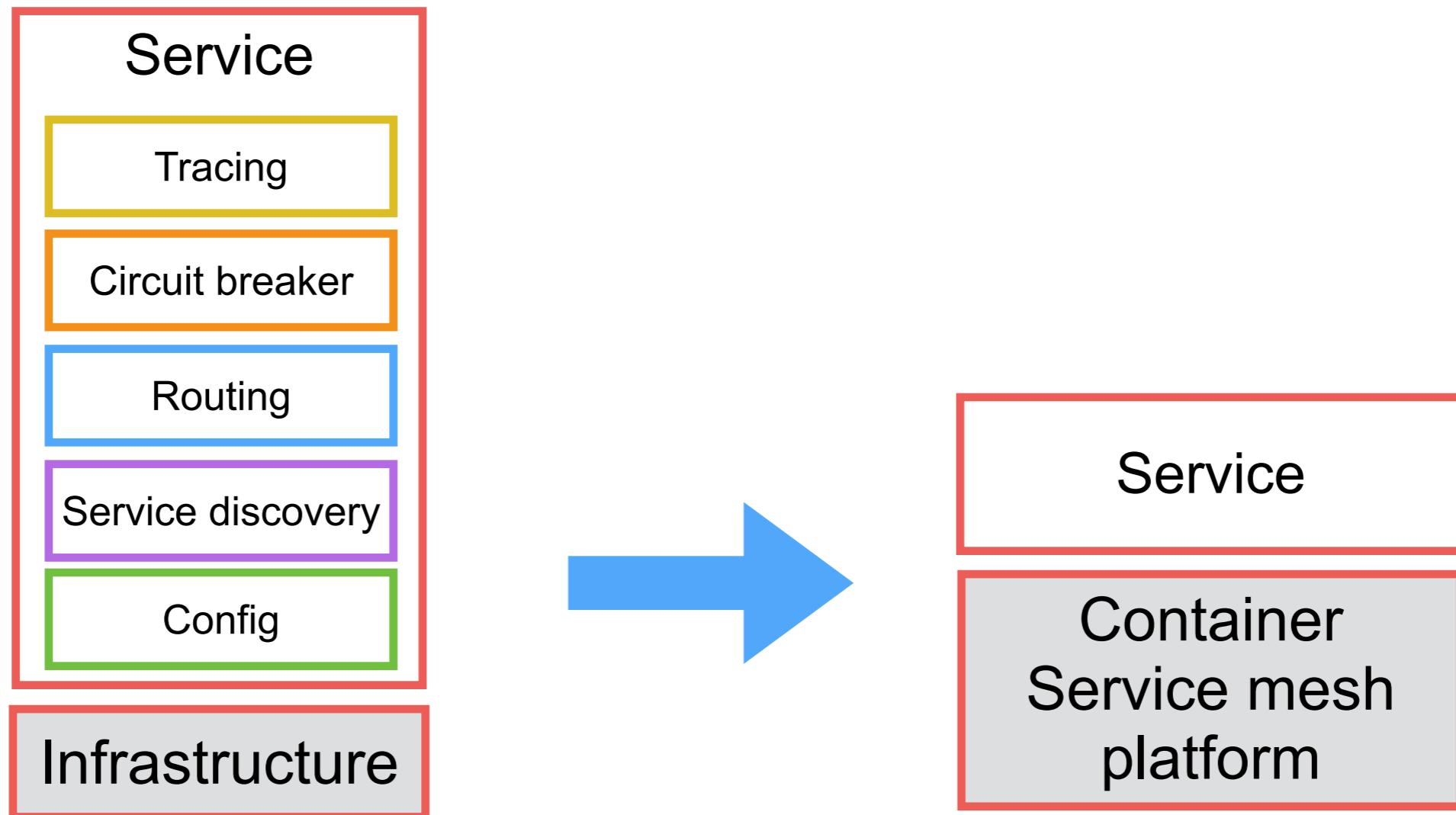
Istio



<https://istio.io/latest/about/service-mesh/>



Microservice Evolution



Microservice 4.0 == FaaS



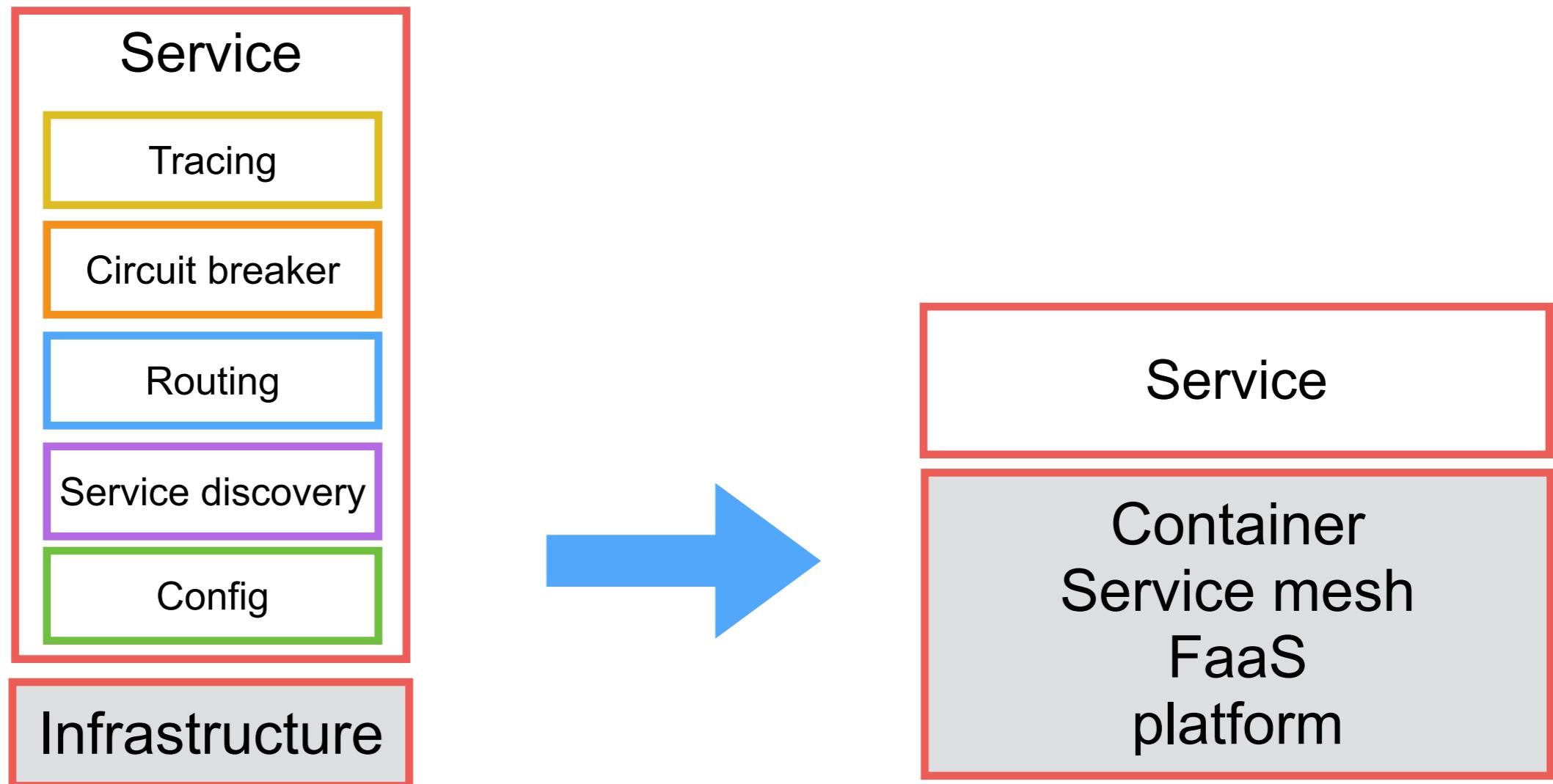
OPEN FAAS



APACHE
OpenWhisk™



Microservice Evolution

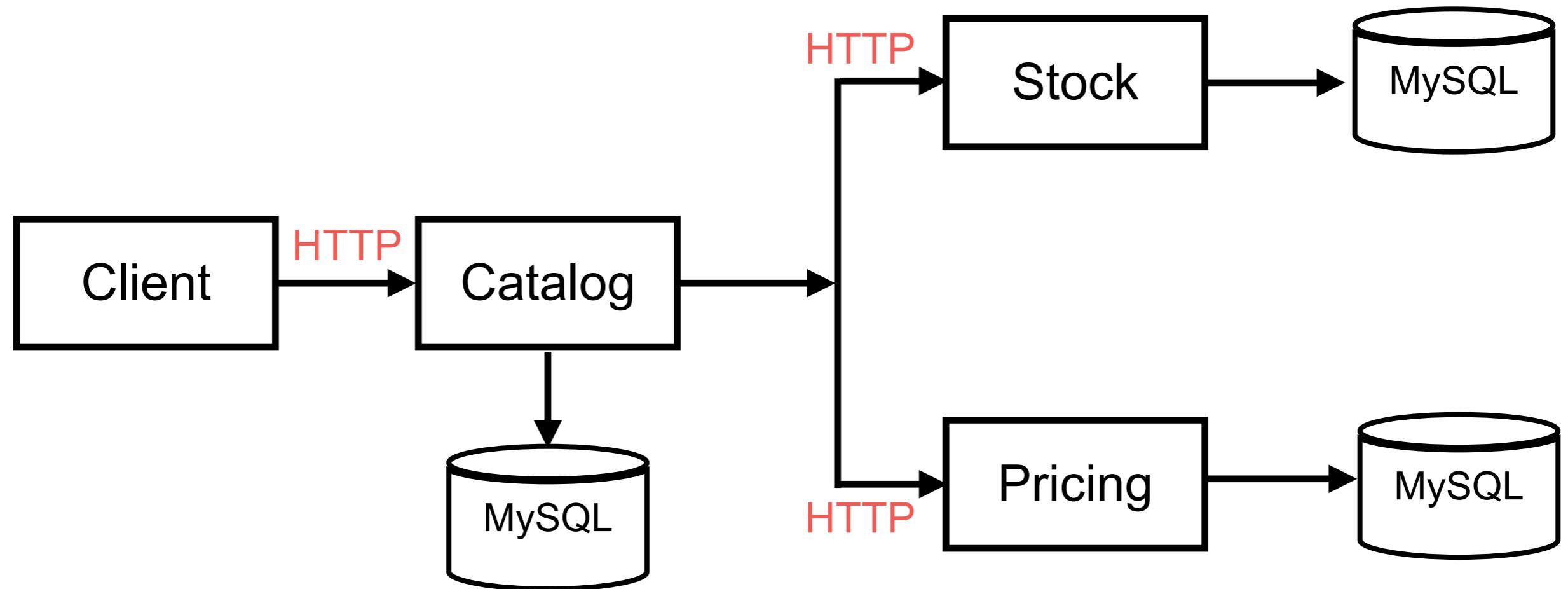


Develop Microservice Workshop

<https://github.com/up1/workshop-develop-microservices-2023>



Architecture for Synchronous

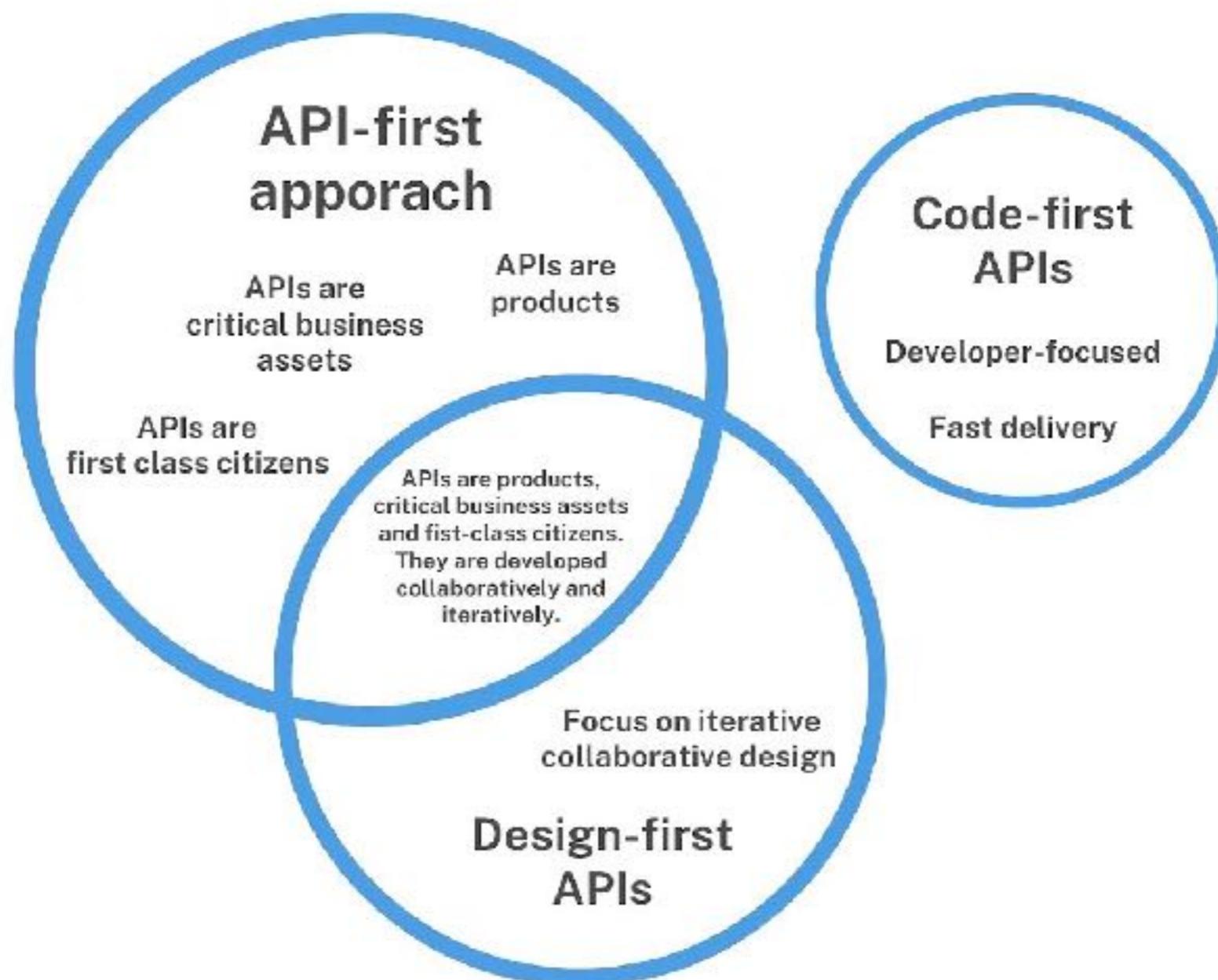


Goals of workshop

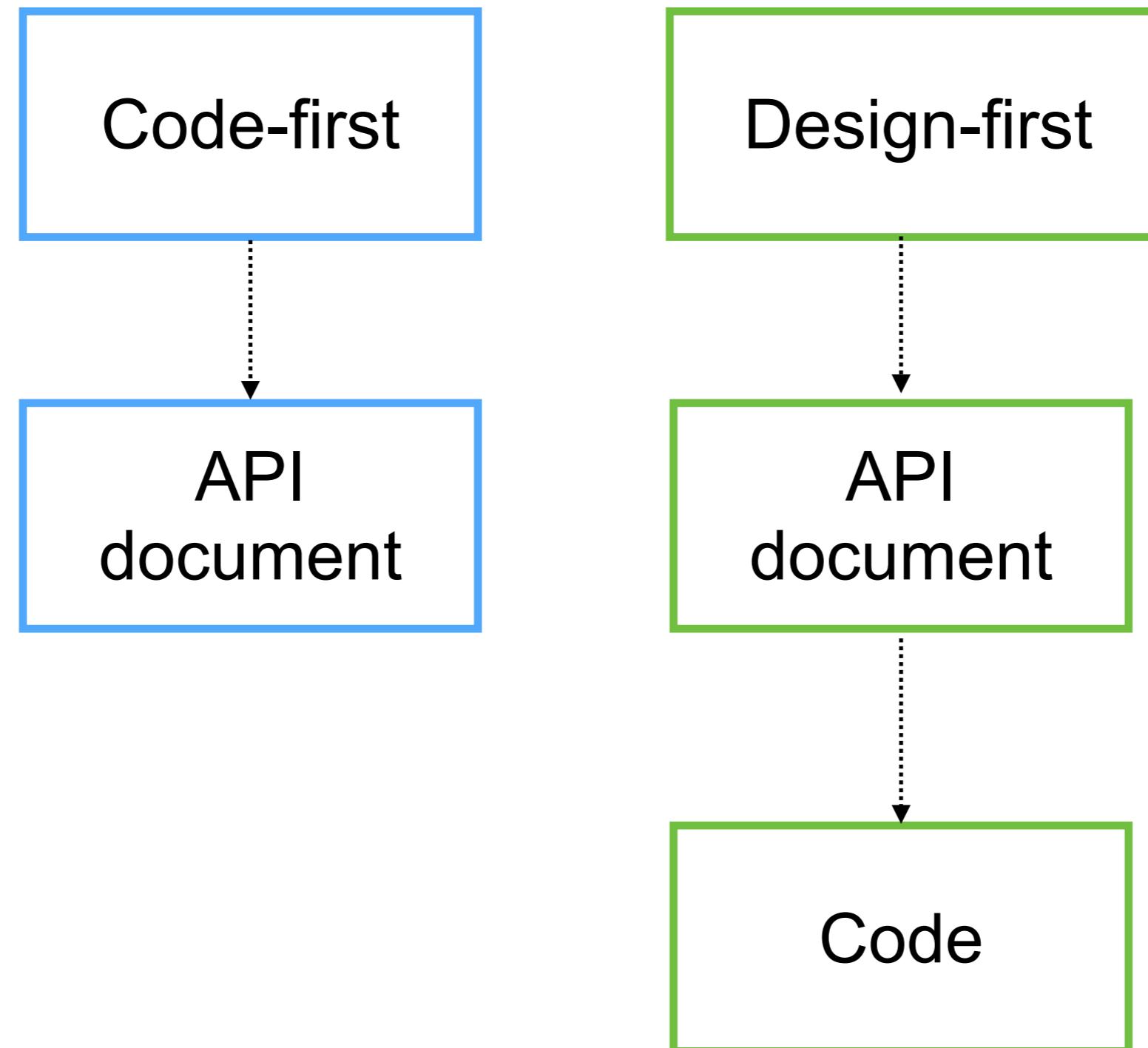
API-first design development
Circuit breaker
API gateway
Rate limit



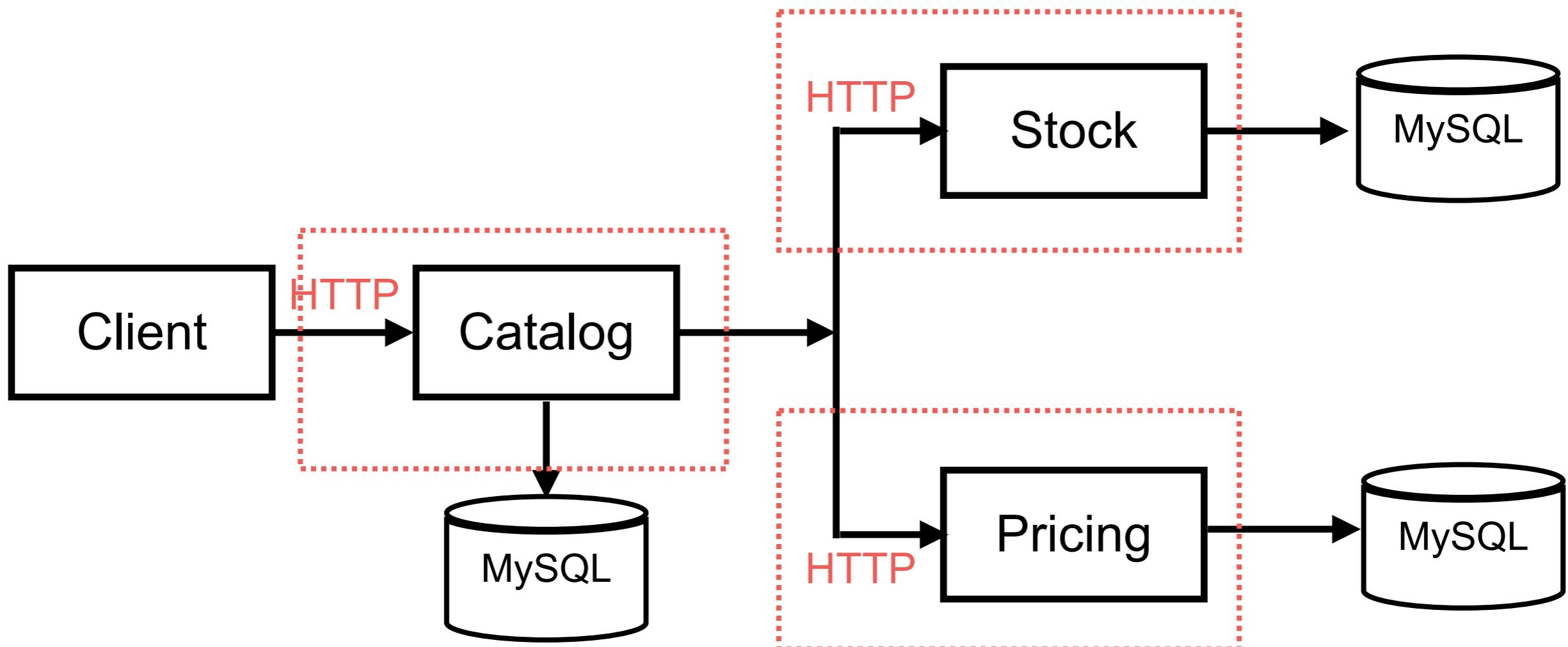
Design API Process



Design and Develop API ?

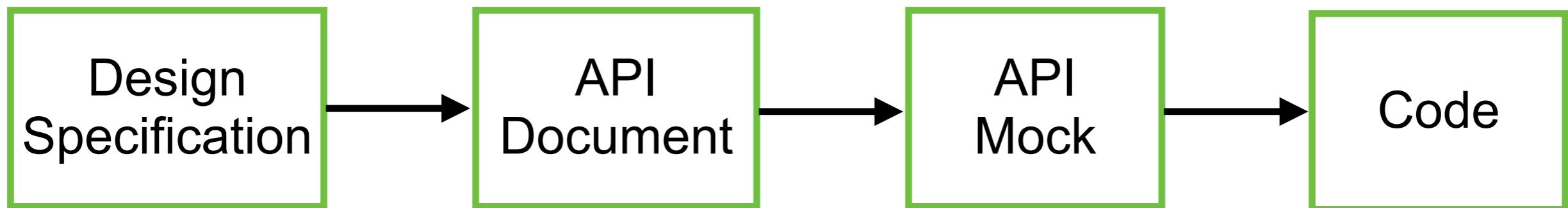


Design APIs for services



API-first design development

Swagger/OpenAPI
Postman
API Blueprint



Swagger/OpenAPI

The screenshot shows the official website for the OpenAPI Initiative. At the top, there's a navigation bar with links for About, Specification, Participate, Governance, Membership, Blog, FAQ, and What Is OpenAPI? Below the navigation is a social media icon bar with Twitter, LinkedIn, and GitHub icons, followed by a search icon.

The main banner features a dark blue background with a large green "3.1.0 RELEASE" text in the center. To the left of the release text is the message "Something great got better, get excited!". To the right is a green circular icon with a gear-like pattern. Above the banner, the text "Compatible with JSON Schema" is visible. Below the banner are two green buttons labeled "REVIEW THE SPEC" and "GETTING STARTED".

The main content area below the banner contains the OpenAPI logo (a green circular icon with a gear-like pattern next to the word "OPENAPI"). A descriptive paragraph explains what the OpenAPI Specification is and how it can be used. A green "Learn More" button is located at the bottom left of this section.

OpenAPI Initiative Home Page

The OpenAPI Specification is a specification language for HTTP APIs that provides a standardized means to define your API to others. You can quickly discover how an API works, configure infrastructure, generate client code, and create test cases for your APIs. Read more about how you can get control of your APIs now, understand the full API lifecycle and communicate with developer communities inside and outside your organization.

[Learn More](#)

<https://www.openapis.org/>



Microservices

© 2022 - 2023 Siam Chamnankit Company Limited. All rights reserved.

Postman

The screenshot shows the Postman homepage. At the top, there's a navigation bar with links for Product, Pricing, Enterprise, Resources and Support, and Explore. A search bar says "Search Postman" and a "Launch Postman" button is on the right. Below the navigation is a large yellow callout pointing to the main content area. The main content area has a heading "Build APIs together" with a sub-section "Over 25 million developers use Postman. Get started by signing up or downloading the desktop app." It includes an input field for "jsmith@example.com" and a "Sign Up for Free" button. Below this is a "Download the desktop app" section with icons for Windows, Mac, and Linux. To the right of the callout is a detailed view of the Postman interface showing a workspace with a database collection, requests, and responses. A sidebar on the left shows collections like "Helen MR" and "Database". At the bottom right is a cartoon illustration of three people launching a rocket.

Build APIs together

Over 25 million developers use Postman. Get started by signing up or downloading the desktop app.

jsmith@example.com

Sign Up for Free

Download the desktop app

Windows Mac Linux

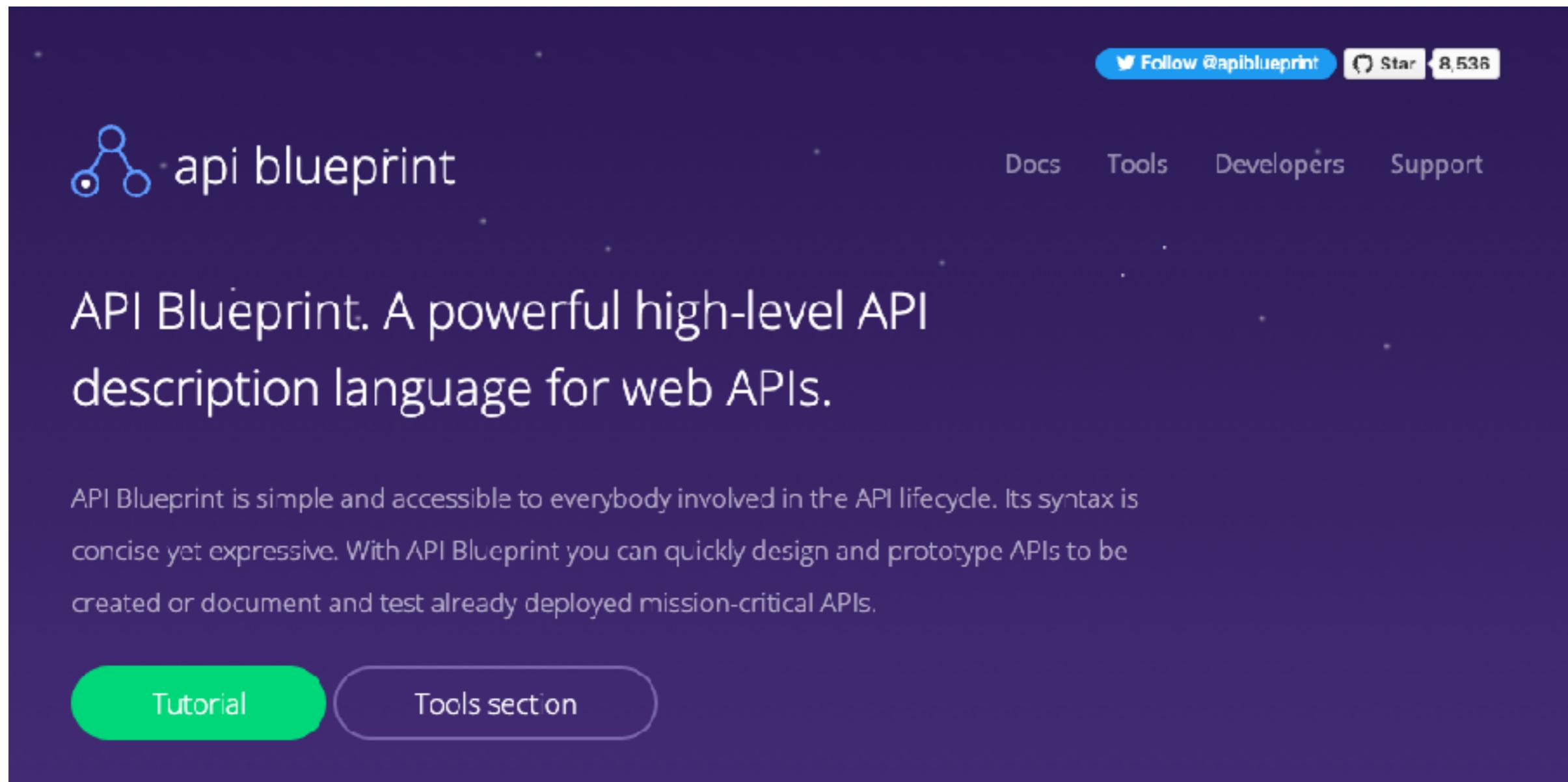
What is Postman?

https://www.postman.com/home

<https://www.postman.com/home>



API Blueprint



The screenshot shows the official website for API Blueprint. At the top right, there are social media links for Twitter and GitHub, followed by a star icon indicating 8,536 stars. Below the header, the logo 'api blueprint' is displayed next to a blue network icon. A navigation bar includes links for 'Docs', 'Tools', 'Developers', and 'Support'. The main content area features a large heading: 'API Blueprint. A powerful high-level API description language for web APIs.' Below this, a paragraph explains the purpose of API Blueprint: 'API Blueprint is simple and accessible to everybody involved in the API lifecycle. Its syntax is concise yet expressive. With API Blueprint you can quickly design and prototype APIs to be created or document and test already deployed mission-critical APIs.' At the bottom of the main section, two buttons are visible: a green 'Tutorial' button and a white 'Tools section' button.

Follow @apiblueprint ⚡ Star 8,536

api blueprint

Docs Tools Developers Support

API Blueprint. A powerful high-level API description language for web APIs.

API Blueprint is simple and accessible to everybody involved in the API lifecycle. Its syntax is concise yet expressive. With API Blueprint you can quickly design and prototype APIs to be created or document and test already deployed mission-critical APIs.

[Tutorial](#) [Tools section](#)

<https://apiblueprint.org/>



Design-first API development Workshop



SwaggerTM
Supported by SMARTBEAR



POSTMAN



api blueprint

<https://github.com/up1/workshop-api-first>



Microservices

© 2022 - 2023 Siam Chamnankit Company Limited. All rights reserved.

Plan for failure !!



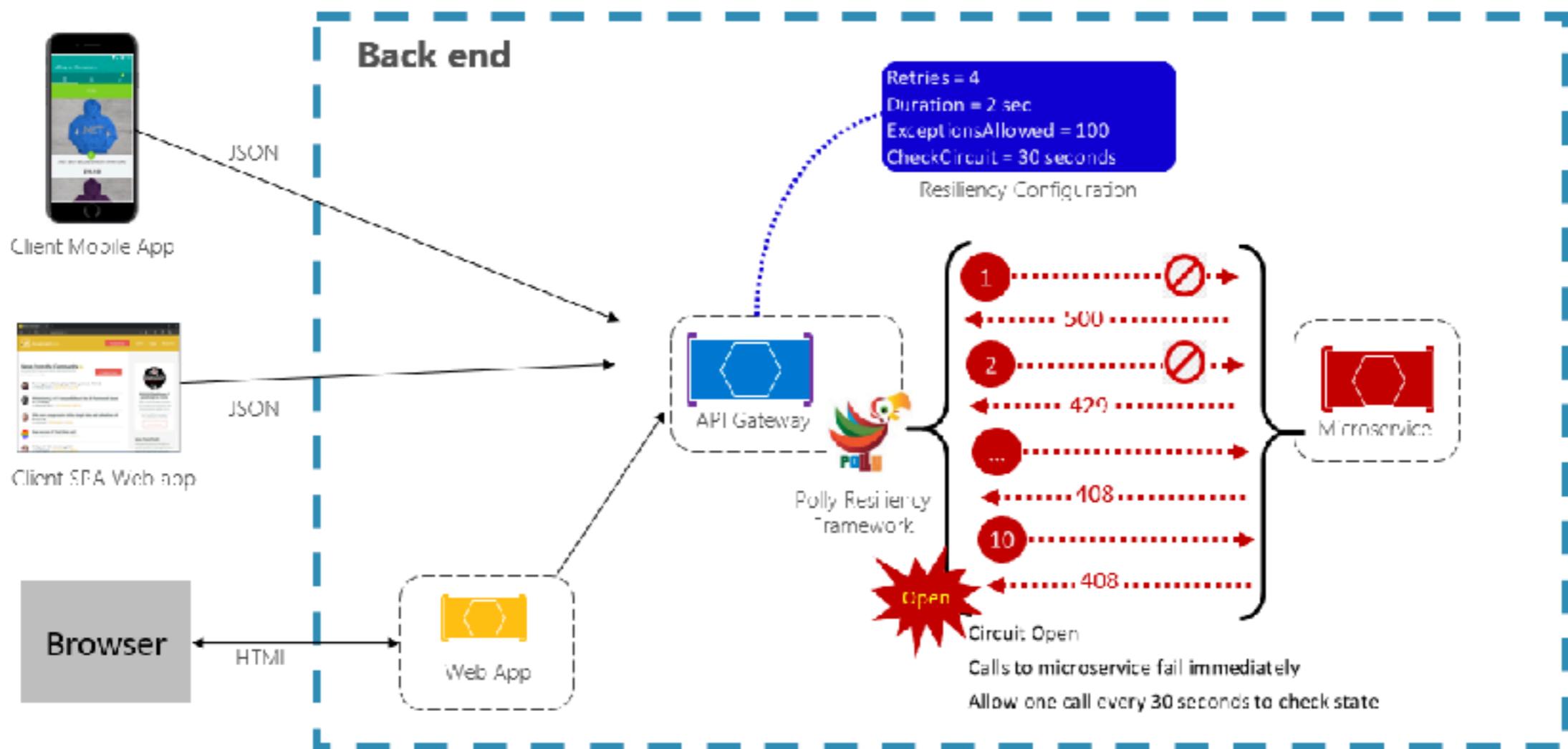
Resilience Patterns

Policy	Description
Timeout	Places limit on the duration for which a caller can wait for a response
Retry	Configures retry operations on designated operations
Cache	Stores responses automatically
Fallback	Defines structured behavior upon a failure
Circuit breaker	Blocks requested operations for a predefined period when faults exceed a configured threshold

<https://learn.microsoft.com/en-us/dotnet/architecture/cloud-native/application-resiliency-patterns>



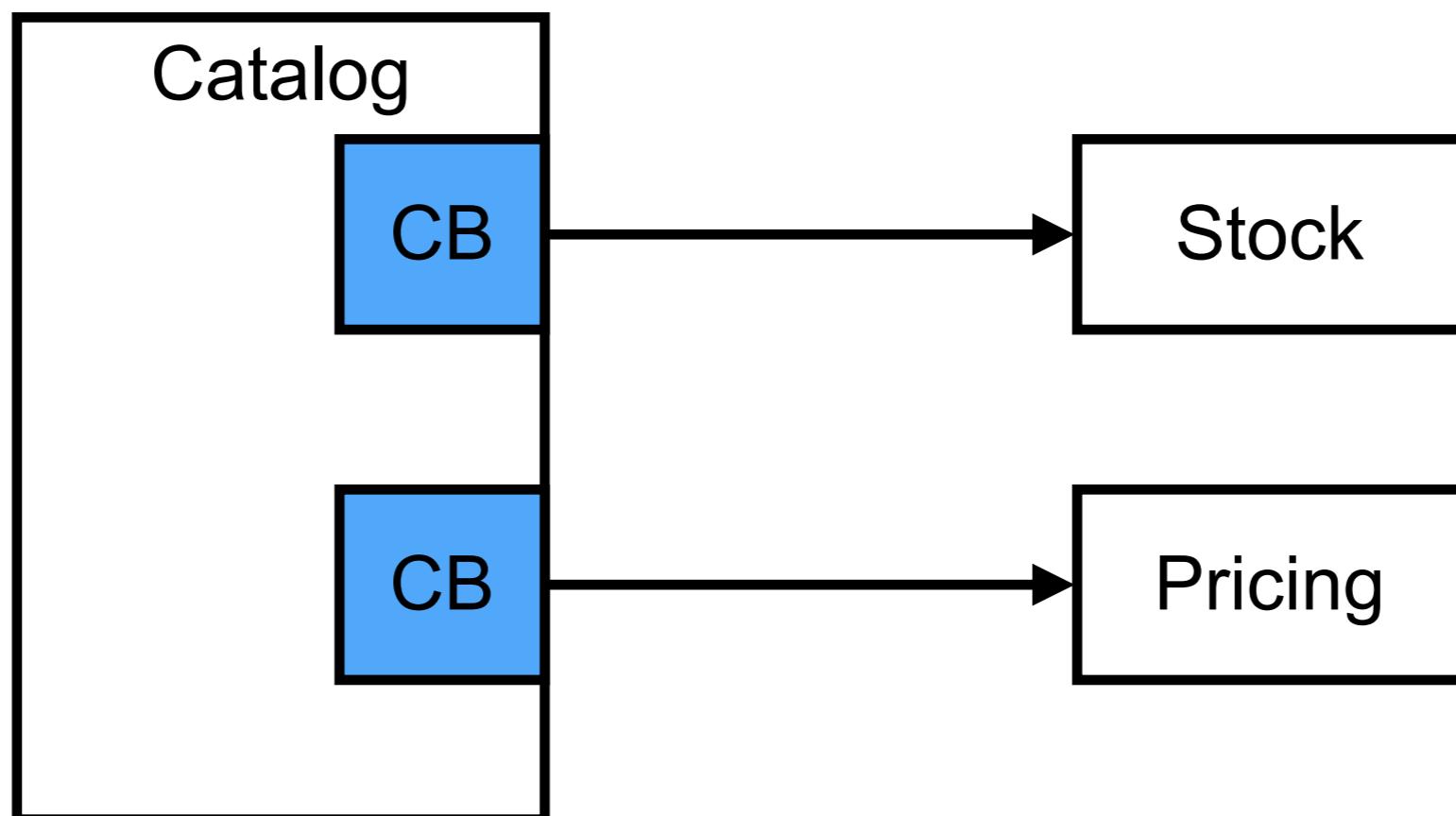
Circuit Breaker



<https://learn.microsoft.com/en-us/dotnet/architecture/cloud-native/application-resiliency-patterns>

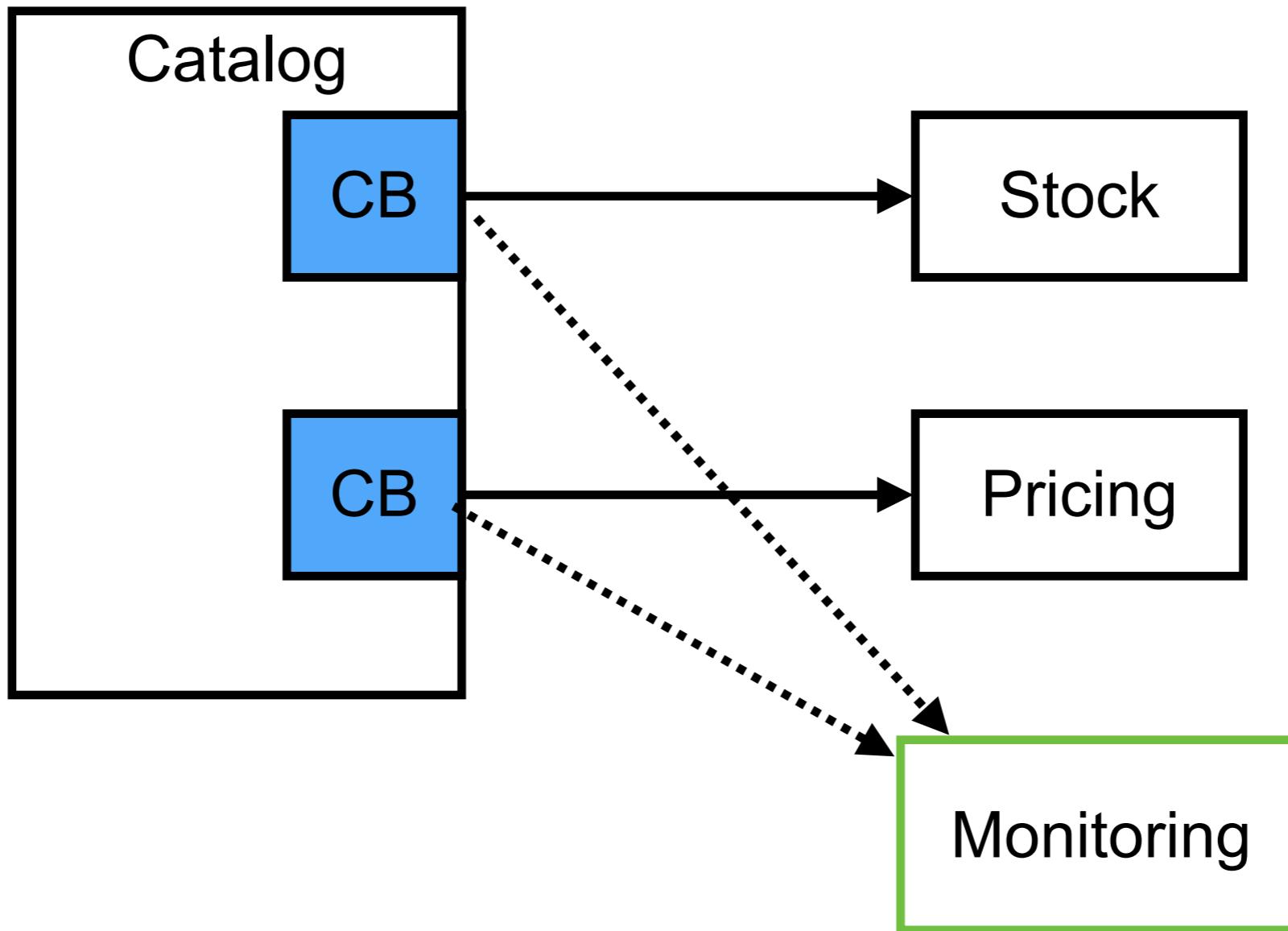


Circuit Breaker



Circuit Breaker

Integrate with alert/monitoring system



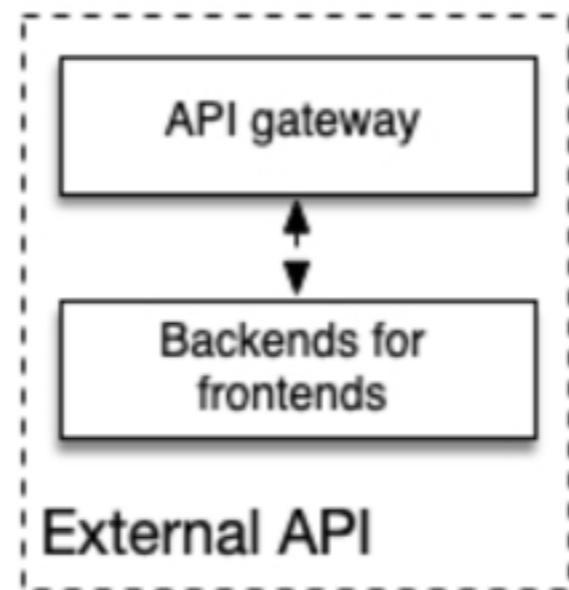
Circuit Breaker Workshop



<https://github.com/up1/workshop-develop-microservices-2023>



External APIs



<https://microservices.io/patterns/index.html>

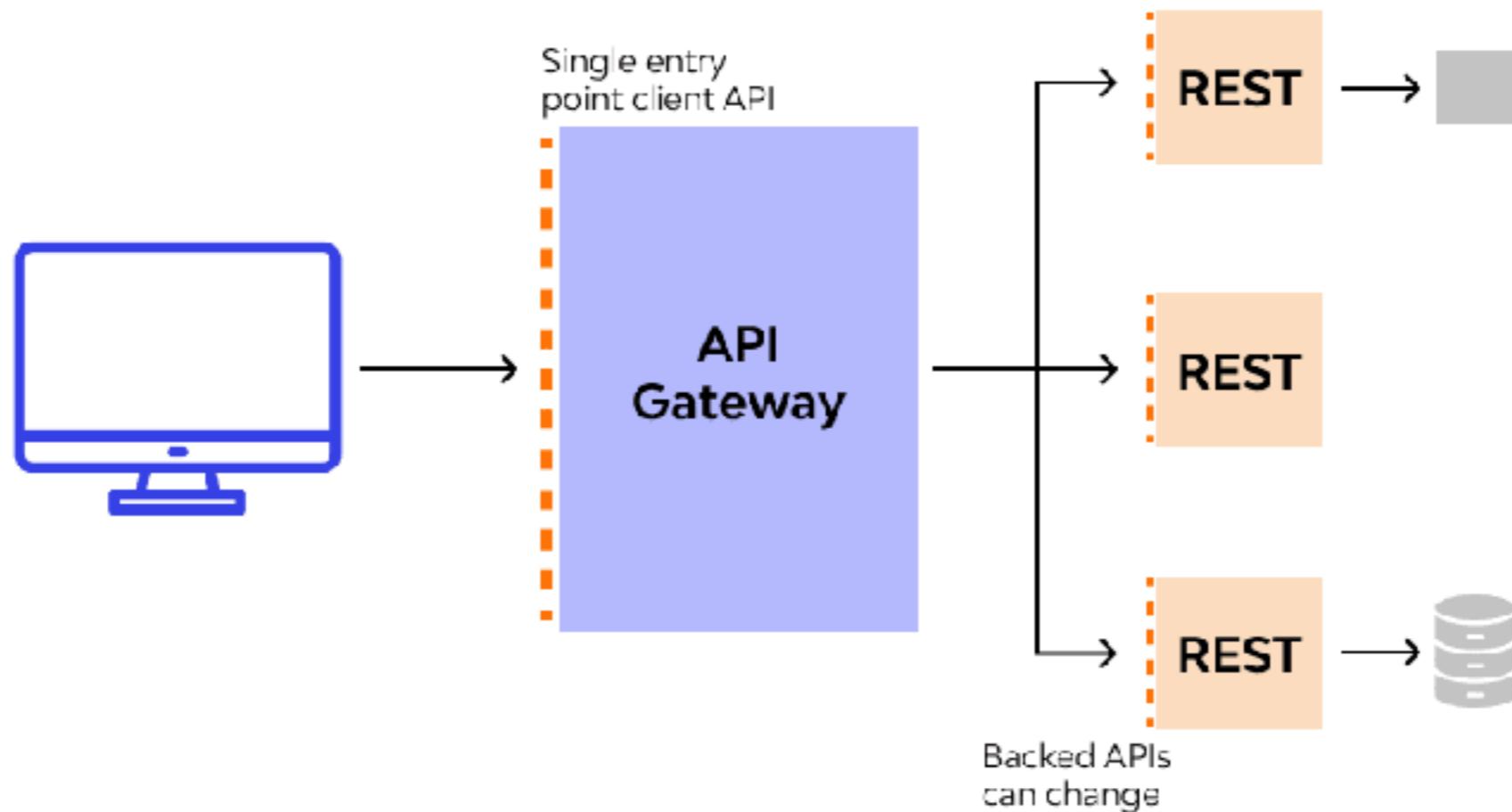


API Gateway



API Gateway

Single entry point for all clients



API Gateway

Routing to services

Circuit breaker

Rate limit

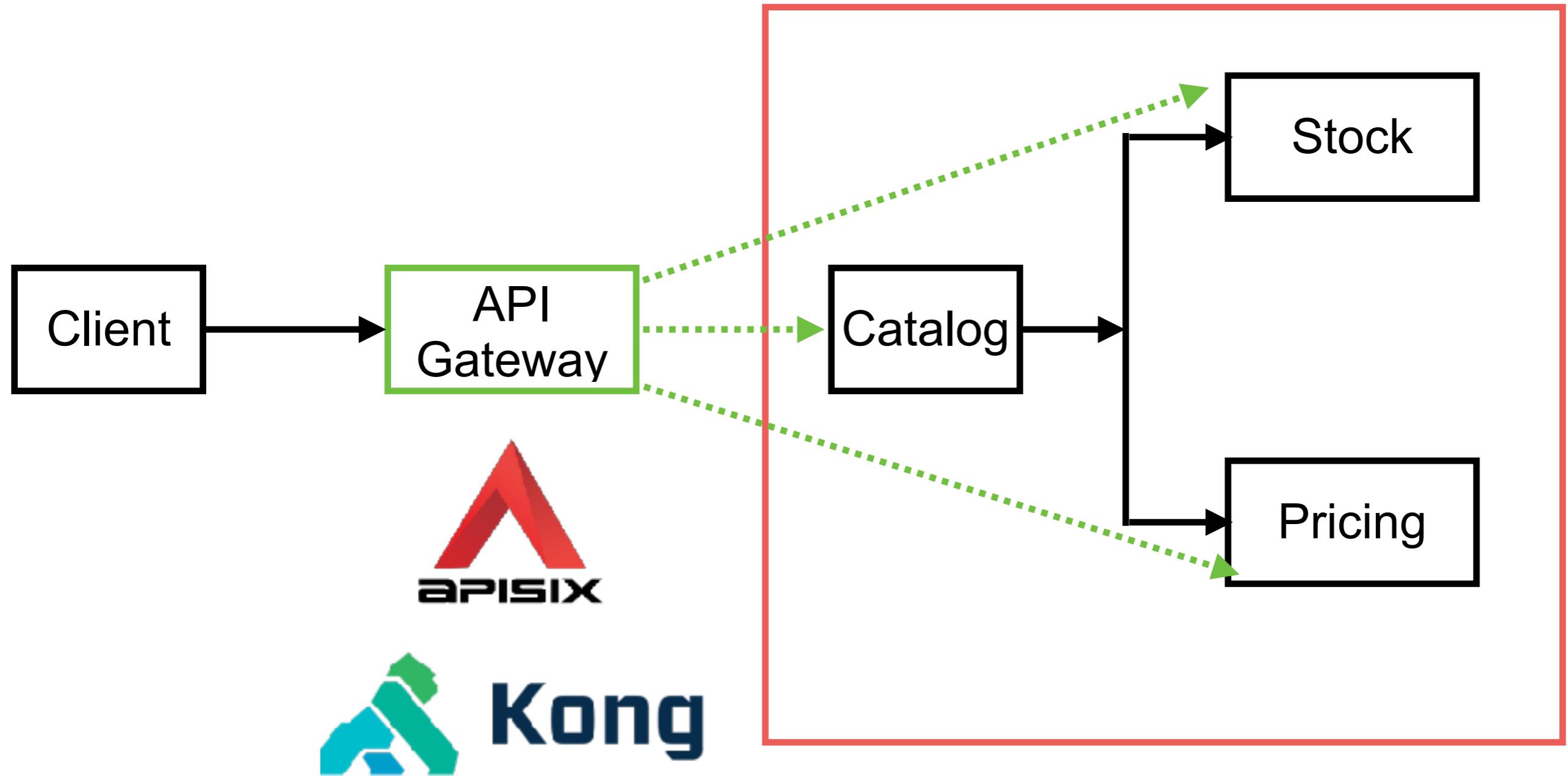
Authentication/Authorization

Logging

Tracing



Workshop with API Gateway



<https://github.com/up1/workshop-develop-microservices-2023>

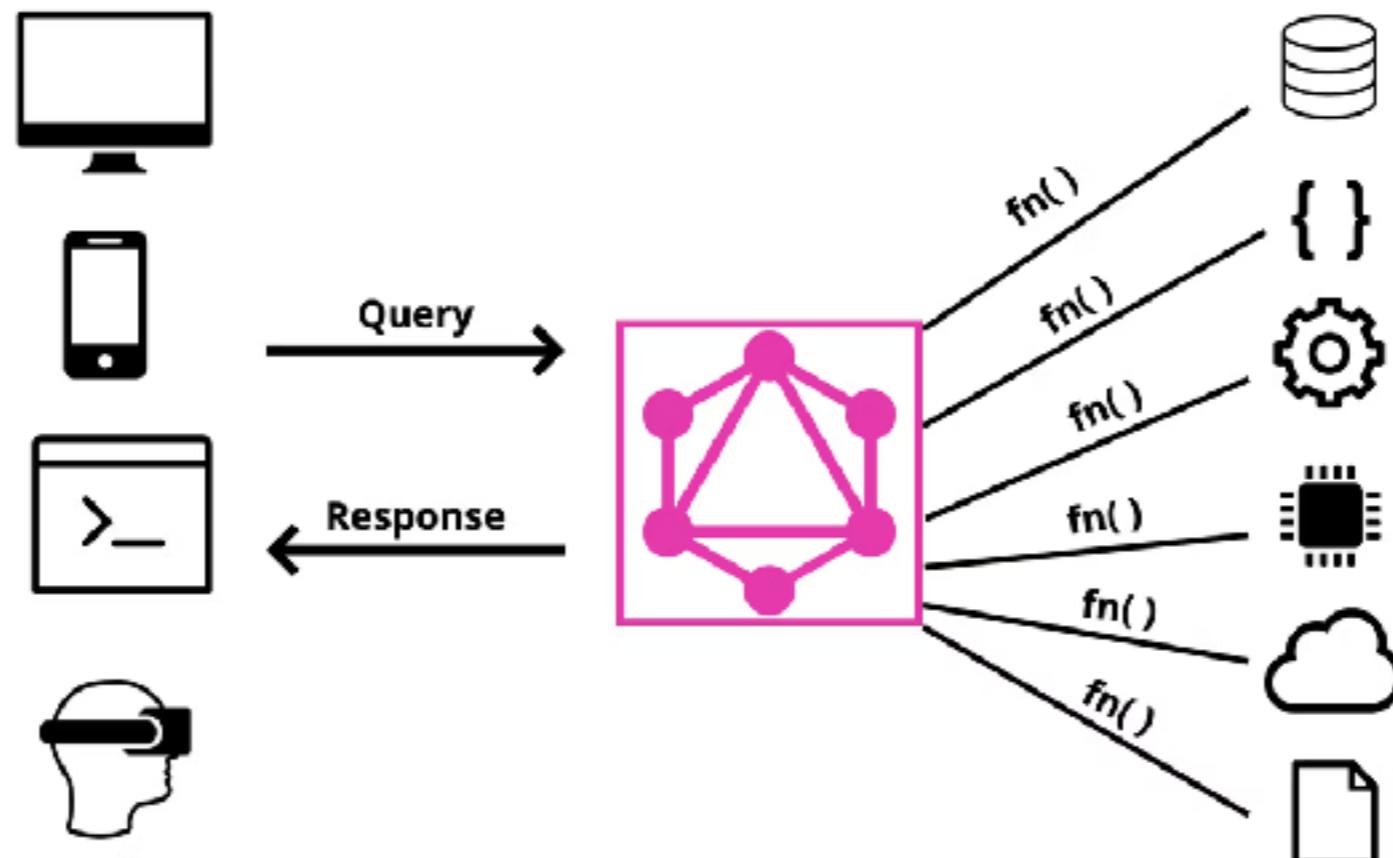


Working with GraphQL



GraphQL

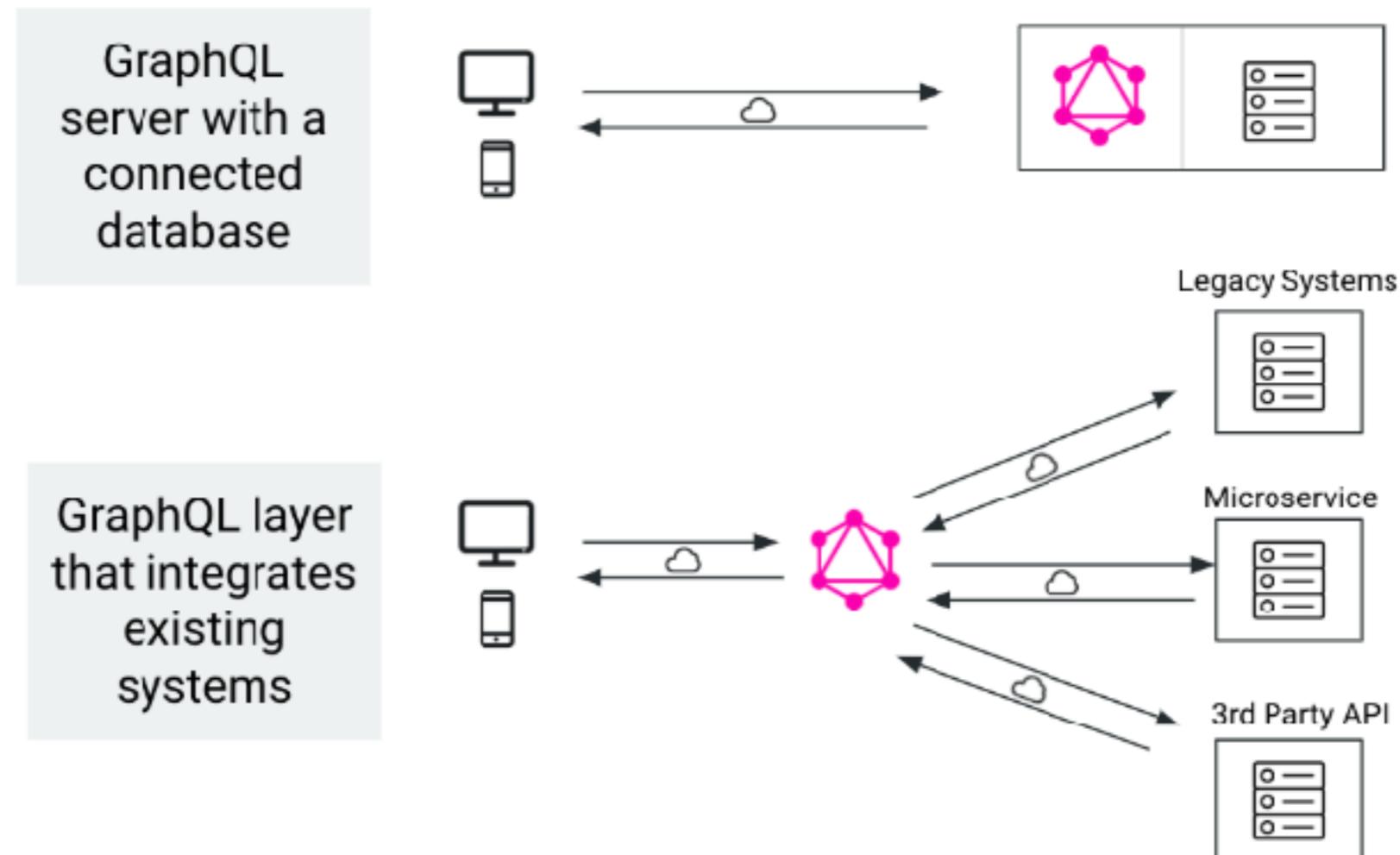
Query language for APIs and runtime
Aggregate data from your existing data



<https://graphql.org/>



GraphQL Use Cases



<https://www.rapid7.com/blog/post/2022/11/14/graphql-security-the-next-evolution-in-api-protection/>



GraphQL Advantages

Faster than REST

Reduce number of requests from client

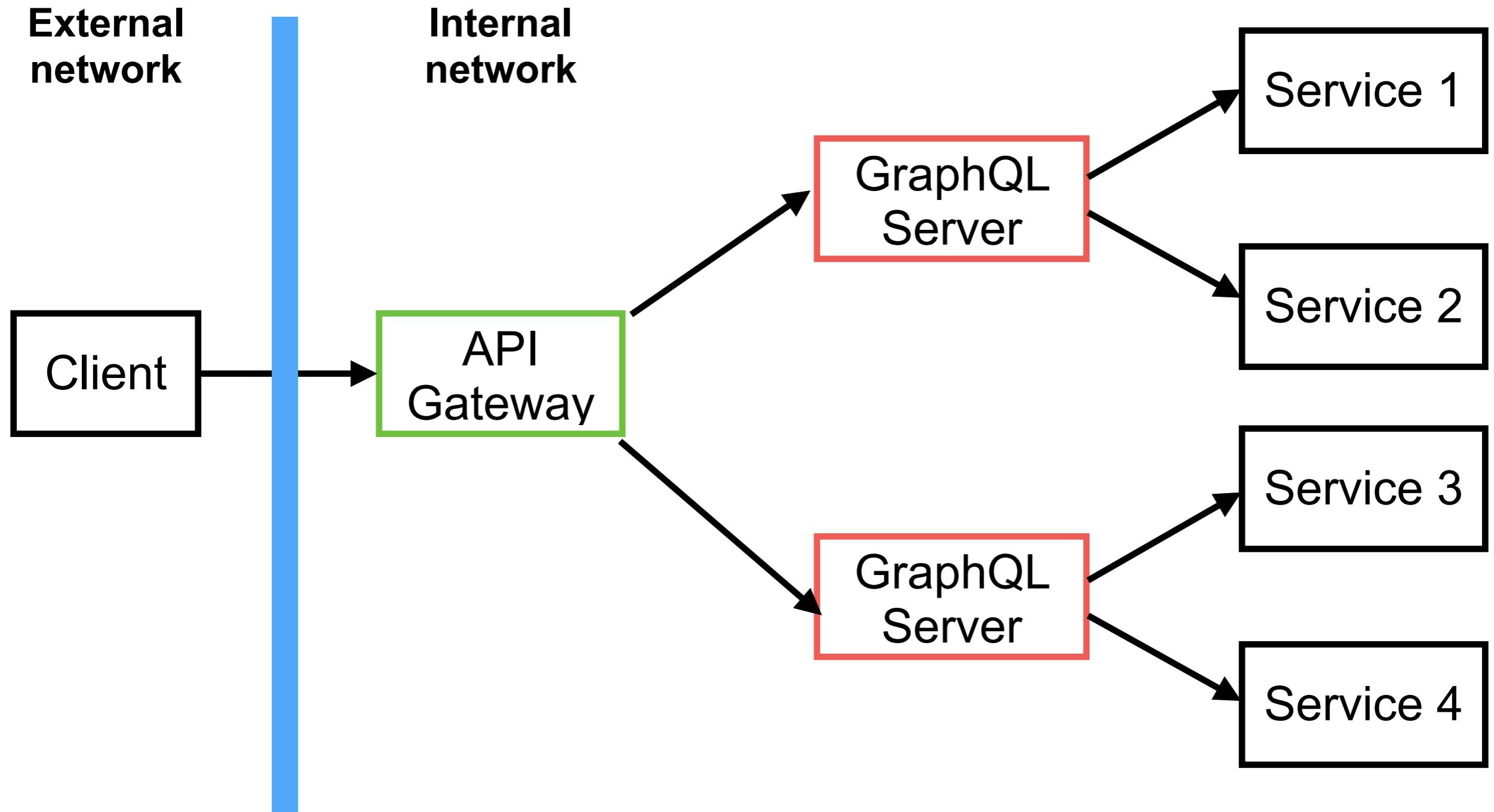
Reduce size of data for each client

Suite for complex systems

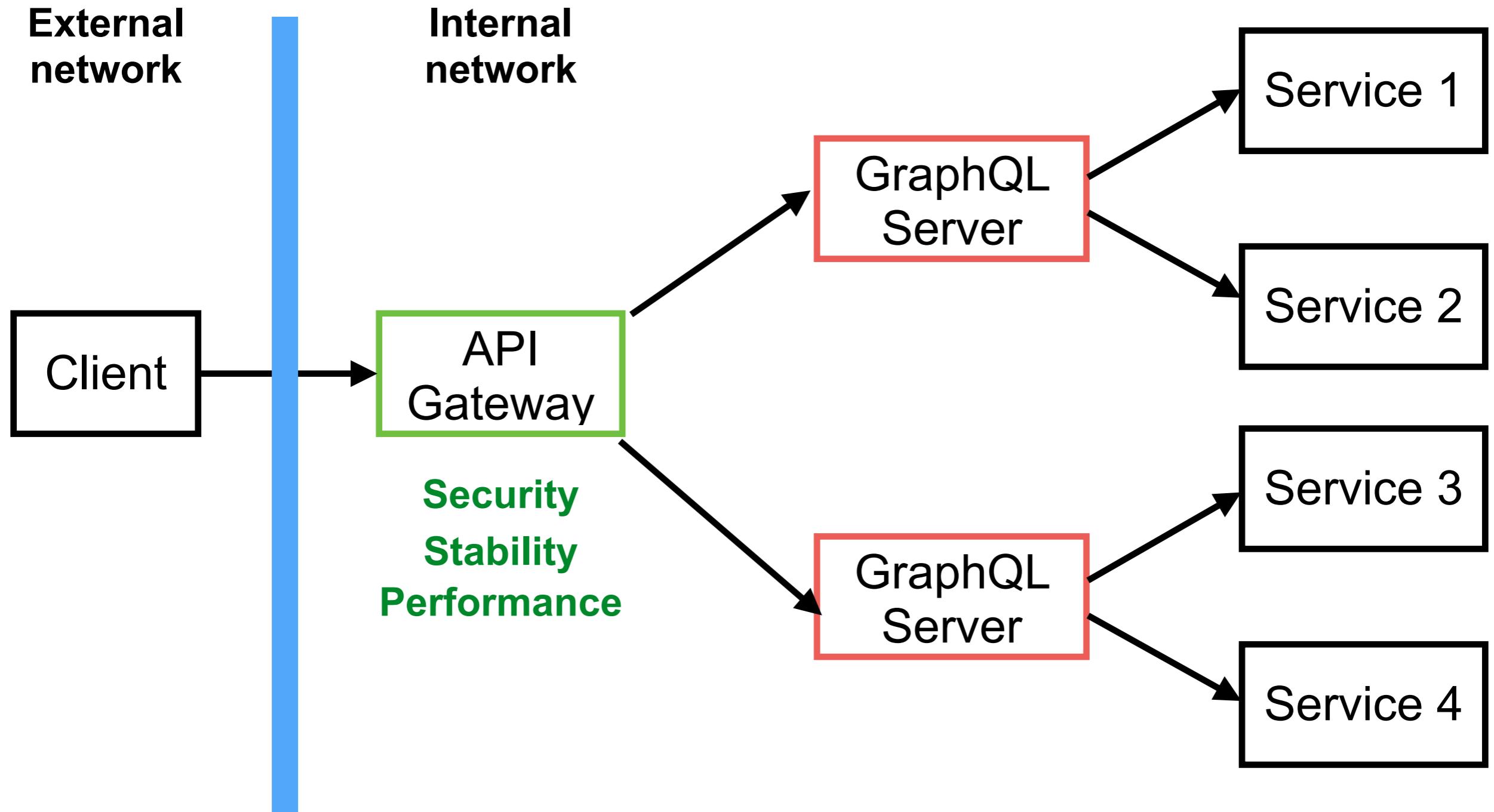
Define data shape and platform



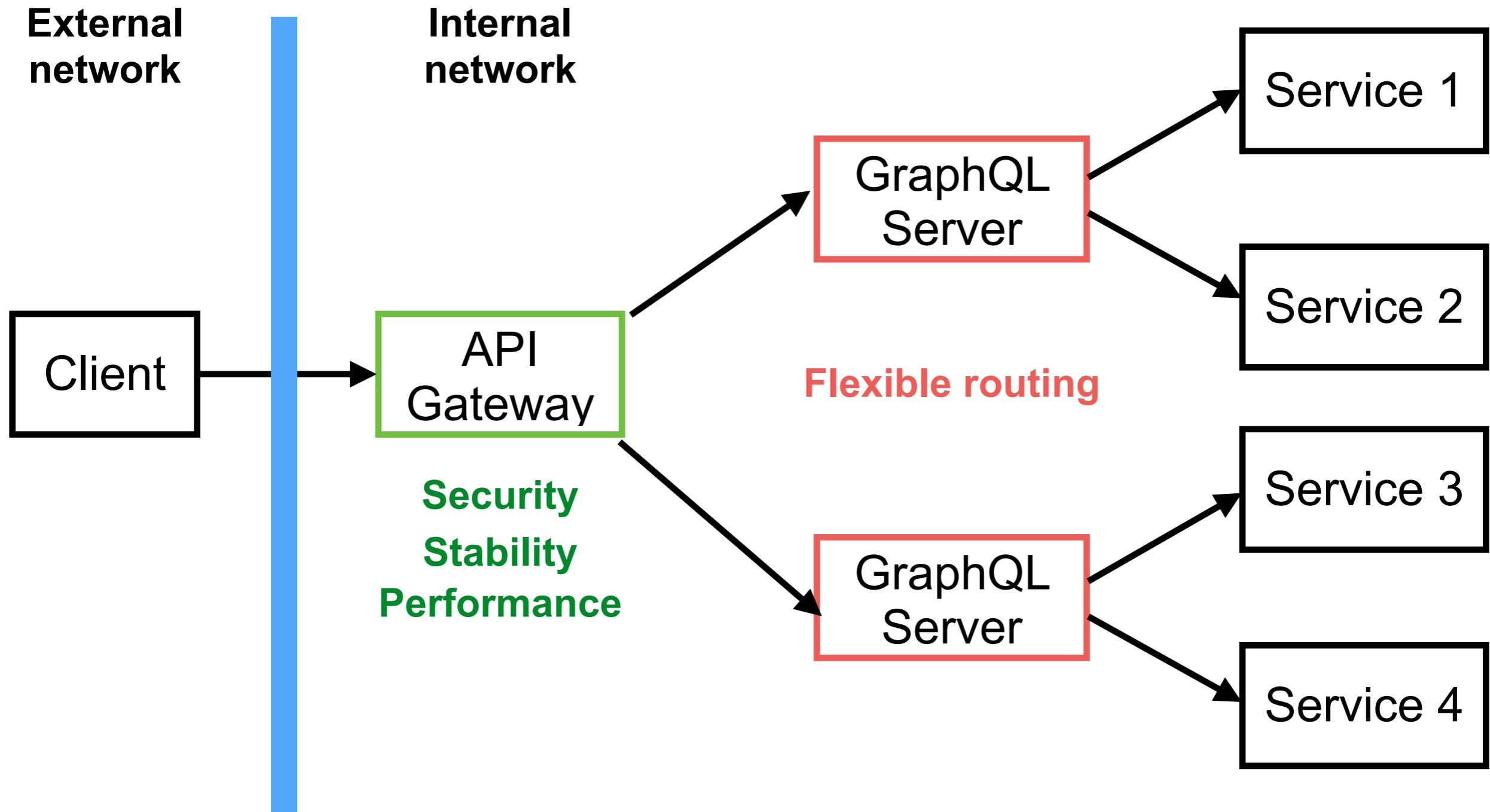
API gateway and GraphQL



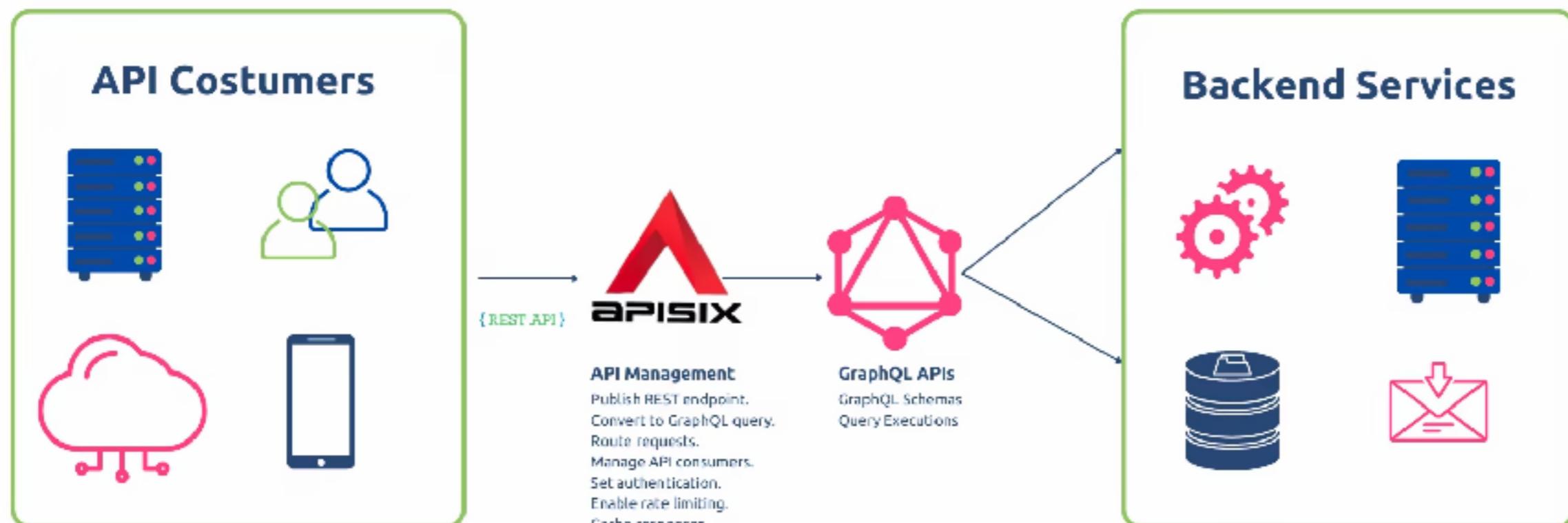
API gateway and GraphQL



API gateway and GraphQL



API gateway and GraphQL



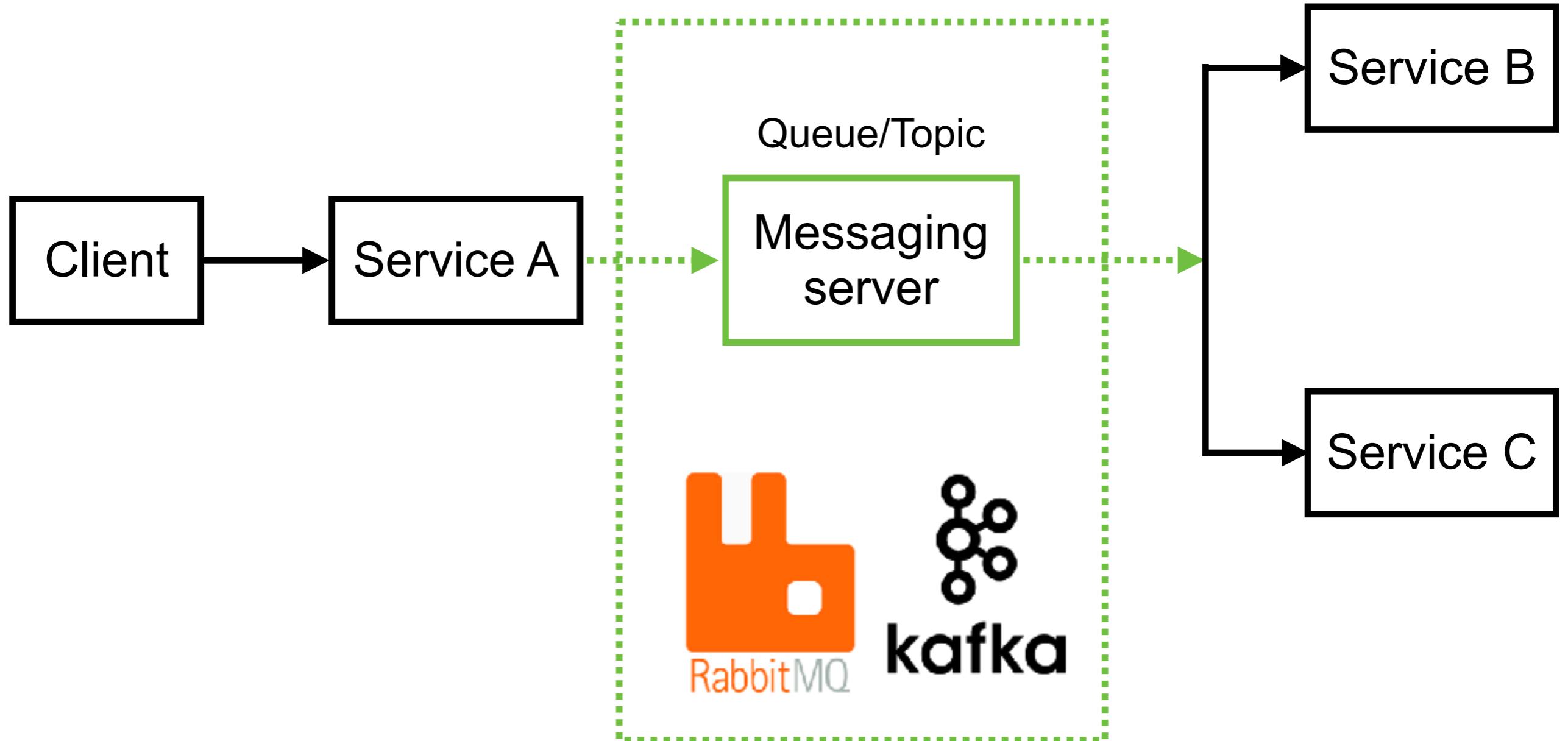
<https://api7.ai/blog/manage-graphql-api-with-api-gateway>



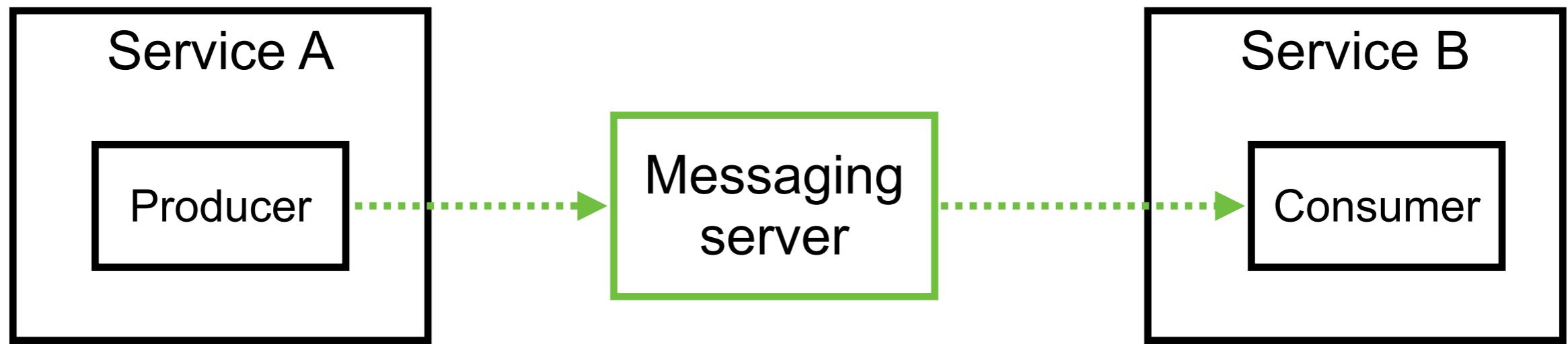
Working with Asynchronous



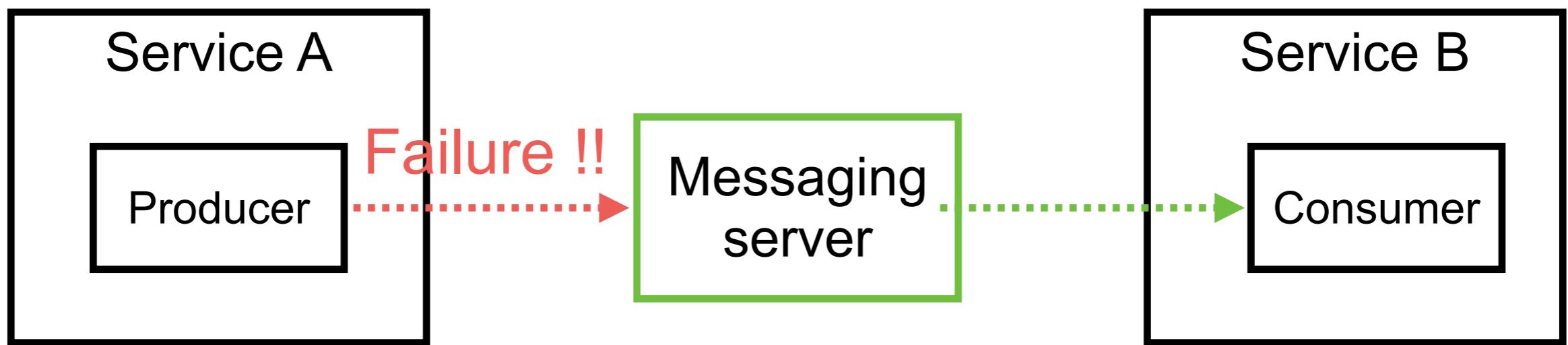
Fire and Forget pattern



Fire and Forget pattern

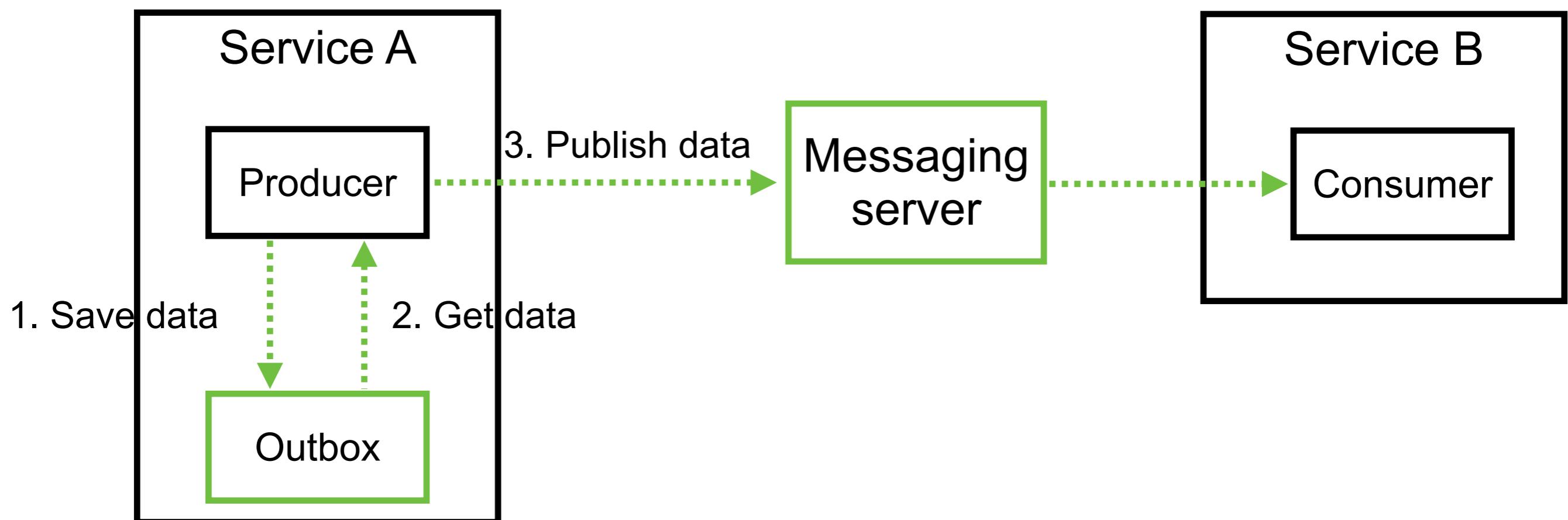


Failure while publishing message



Outbox pattern

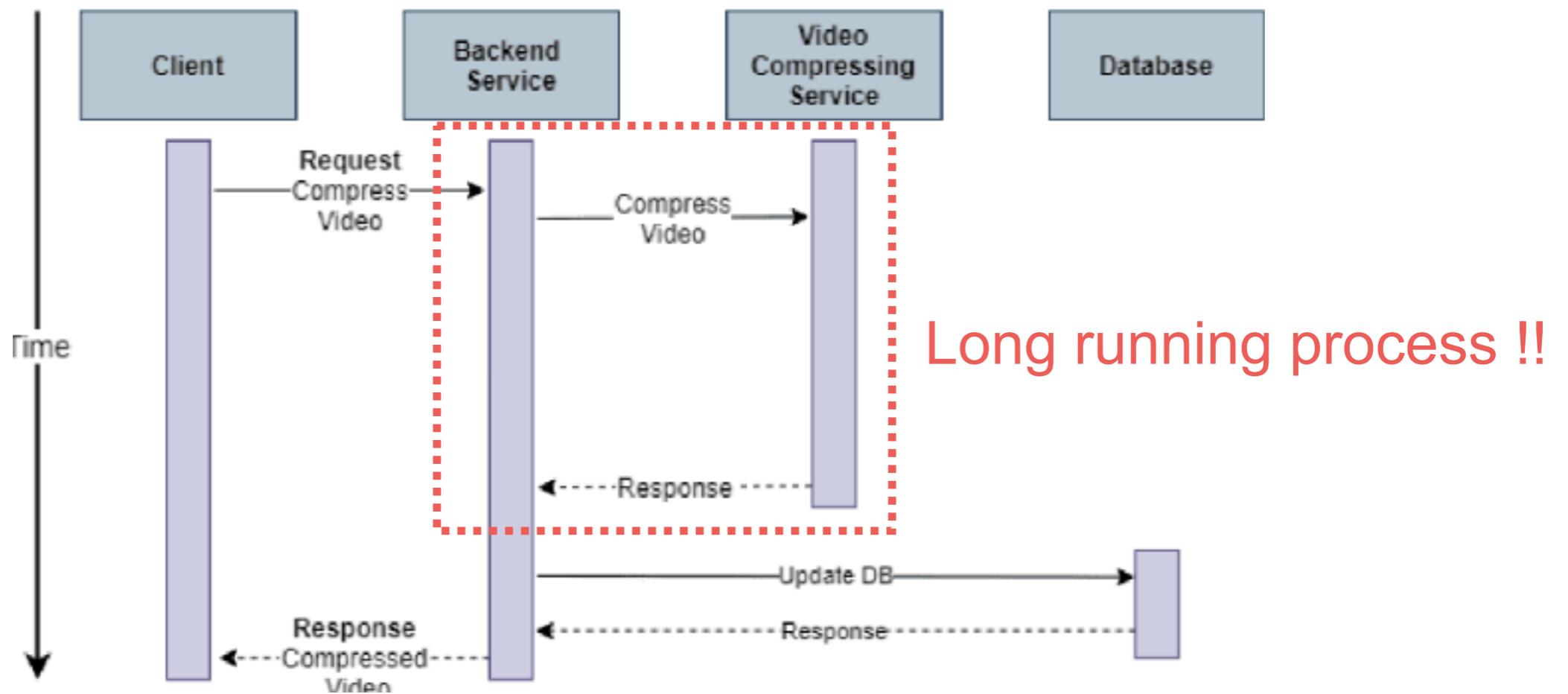
Solution to provide reliability for publish message



Async request and response

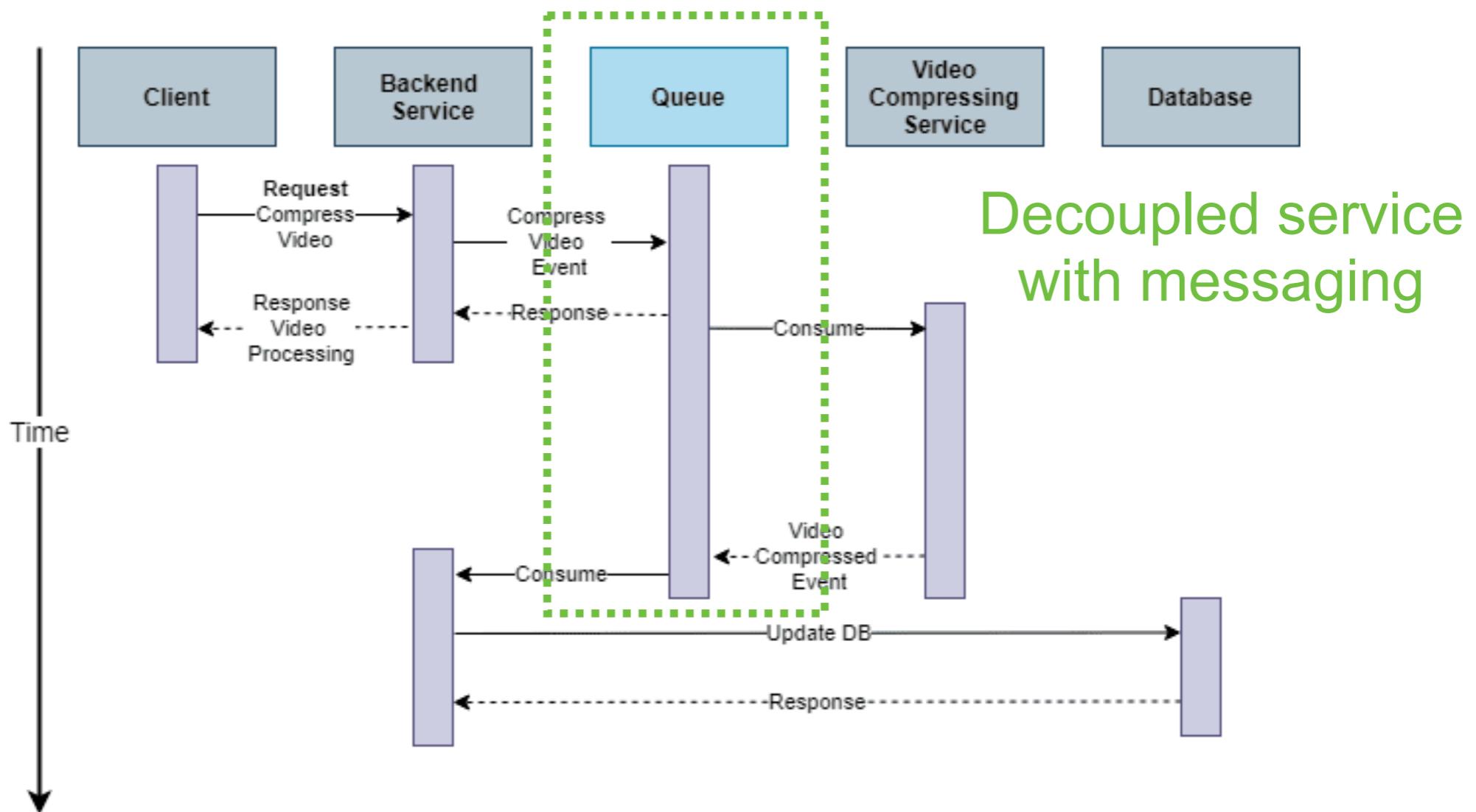
Decoupled backend service from frontend client

Reduce response time !!

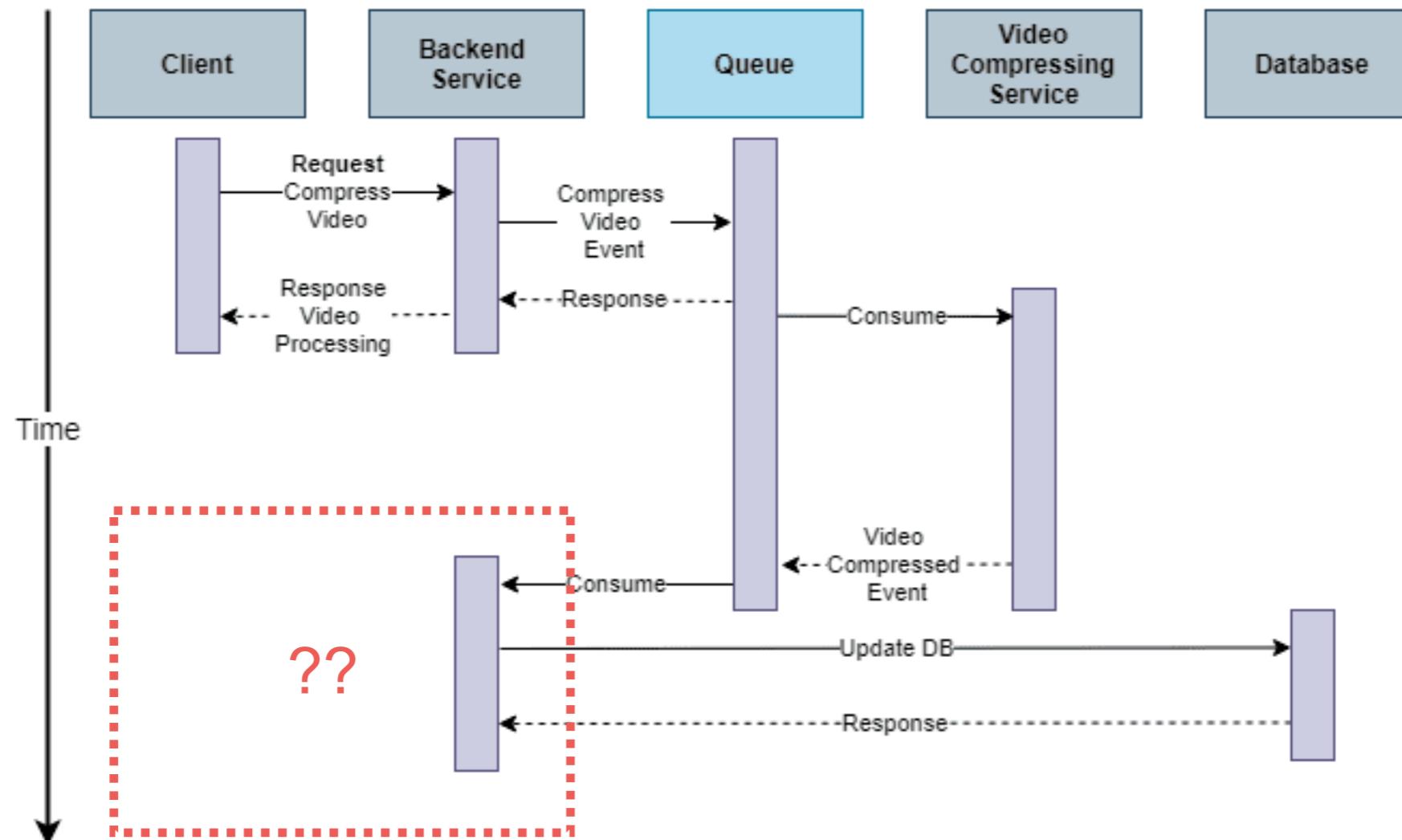


Async request and response

Decoupled backend service from frontend client
Reduce response time !!



How to send response to client ?



How to send response to client ?

HTTP polling

WebSocket

WebHook or Callback

SSE (Sever Send Event)



Asynchronous communication workshop



<https://github.com/up1/workshop-develop-microservices-2023>



Important Quality Services



Important Quality Services

Observability
Configurability
Security



How to find an issue ?

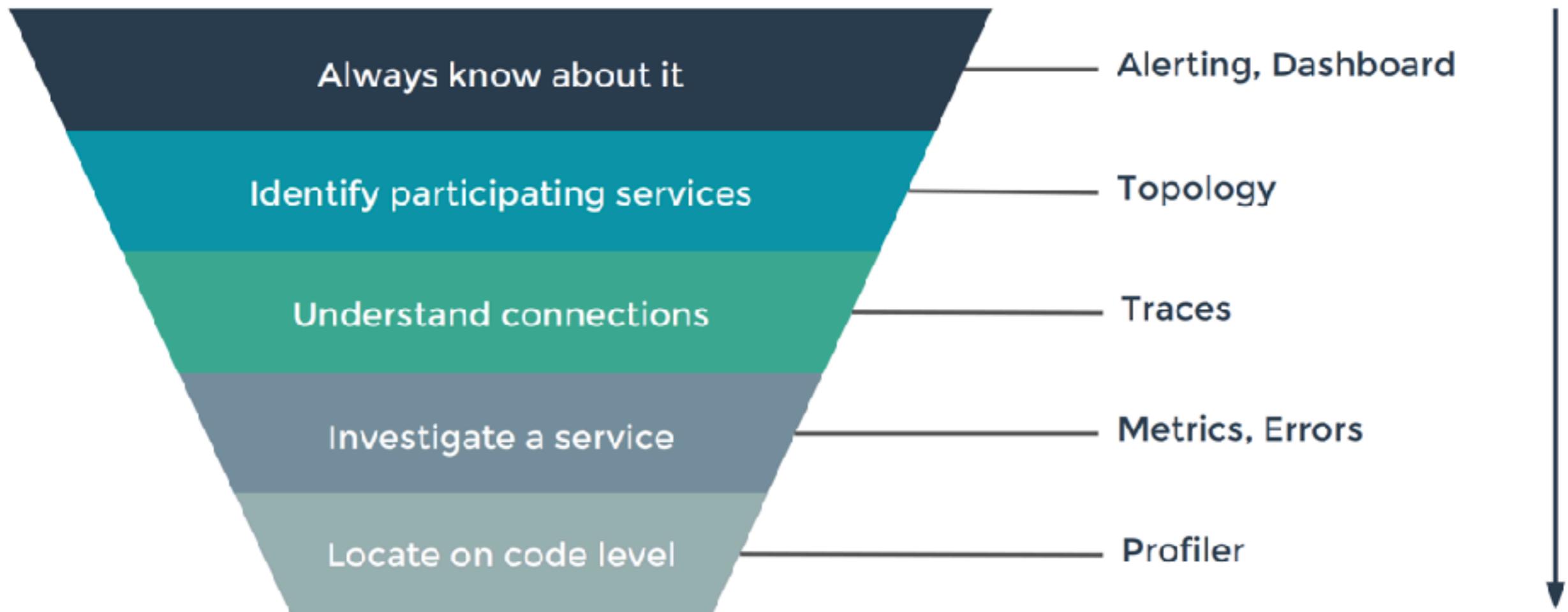
Reactive

Vs.

Proactive



How to find an issue ?



Observable services



Observability vs Monitoring

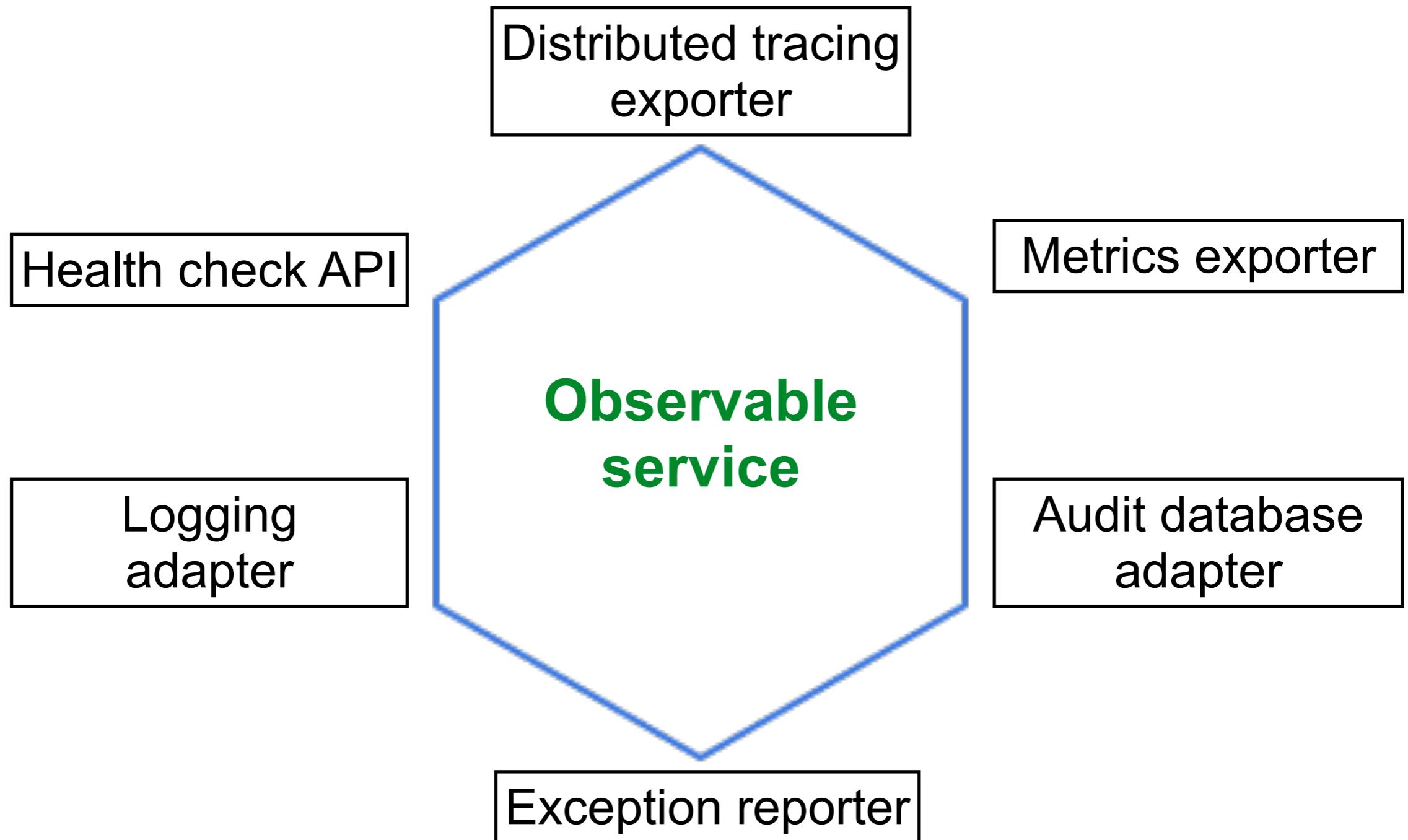


Design observable services

- Health check API
- Log aggregation
- Distributed tracing
- Exception tracking
- Application metrics
- Audit logging

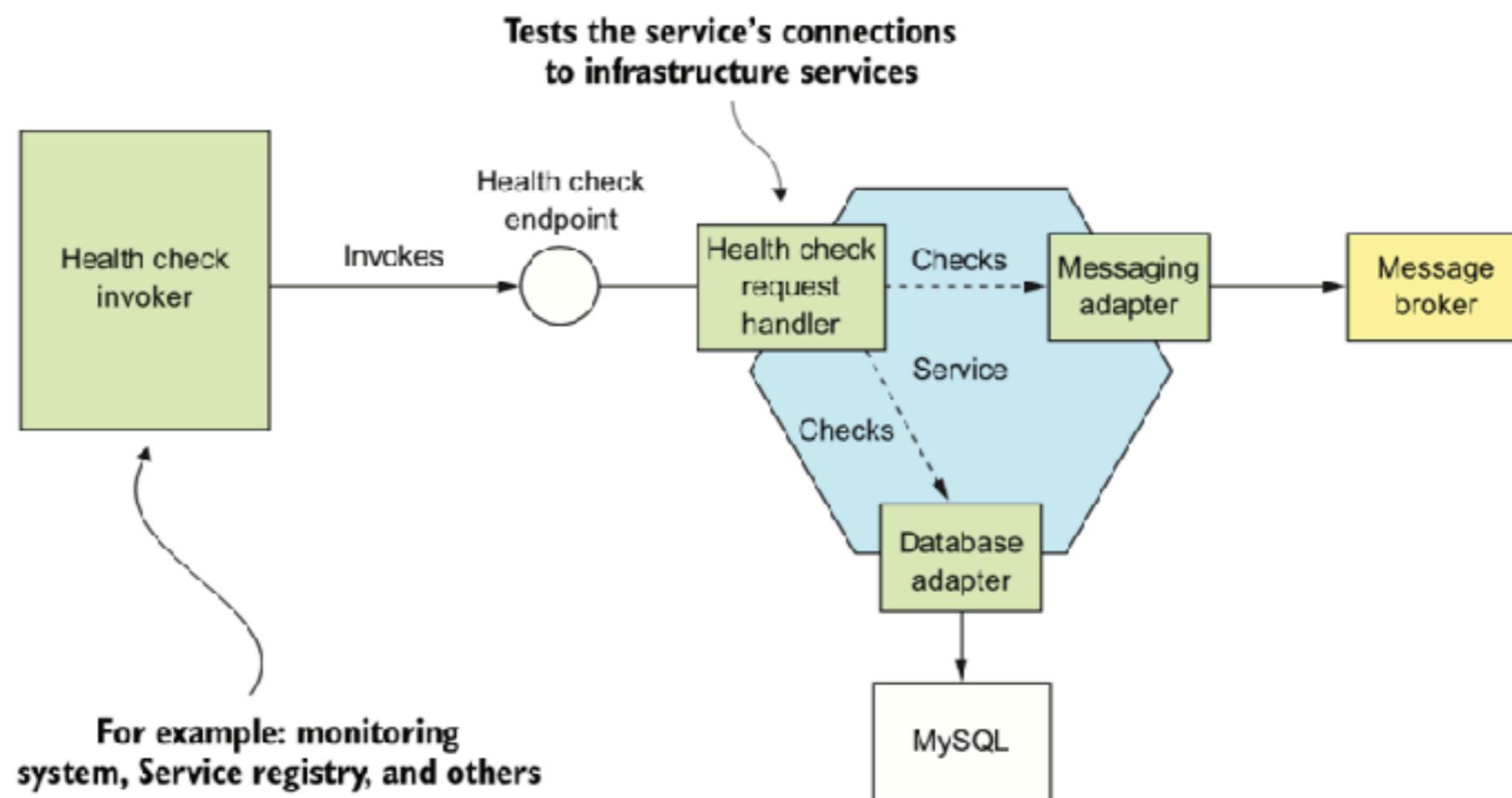


Observable services



Health check API

Expose an endpoint that return the health of service



Monitor your services

Service catalog

Uptime

Service discovery/registry



Service Catalog

The screenshot shows the Backstage Service Catalog interface. At the top, there's a header with a user greeting "Bună ziua, guest!", the title "Backstage Service Catalog", and a timestamp "07:56". Below the header, there are tabs for "SERVICES", "WEBSITES", "LIBRARIES", "DOCUMENTATION", and "OTHER". On the left, a sidebar contains icons for navigation and search, along with sections for "PERSONAL" (Owned: 3, Starred: 0) and "SPOTIFY" (All: 6). The main content area is titled "Owned (3)" and displays a table with three rows:

NAME	OWNER	LIFECYCLE	DESCRIPTION	ACTIONS
playback-order	guest	production	Playback Order	
searcher	guest	production	Searcher	
shuffle-api	guest	production	Shuffle API	

At the bottom right of the main content area, there are buttons for "CREATE COMPONENT" and "SUPPORT".

<https://backstage.io/>



Kuma Uptime

The screenshot shows the Uptime Kuma web interface. On the left, a sidebar lists monitors: Check Port (100%, green), Example.com (100%, green), Facebook (0%, red), Google (100%, green), Inbox by Gmail (0%, red), LouisLam.net (100%, green), MySQL (100%, green), and Ping (100%, green). The main area displays the status for **LouisLam.net** at <https://louislam.net>. It includes a green "Up" button, a checkmark icon, and a message "Check every 60 seconds.". Below this are metrics: Response (Current: 271 ms, 24-hour: 138 ms), Avg. Response (24-hour: 138 ms), Uptime (24-hour: 100%, 30-day: 100%), and Cert Exp. (2023-05-21). A graph at the bottom shows Response Time (ms) over time, with a significant spike around 21:10.

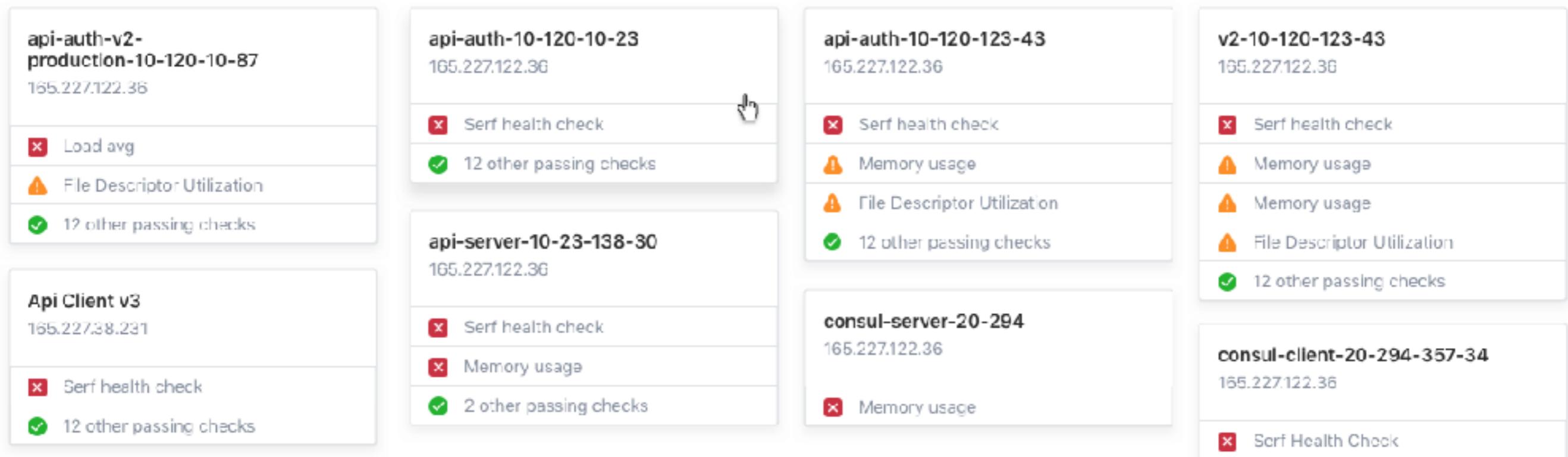
<https://github.com/louislam/uptime-kuma>



Microservices

© 2022 - 2023 Siam Chamnkit Company Limited. All rights reserved.

Service Discovery

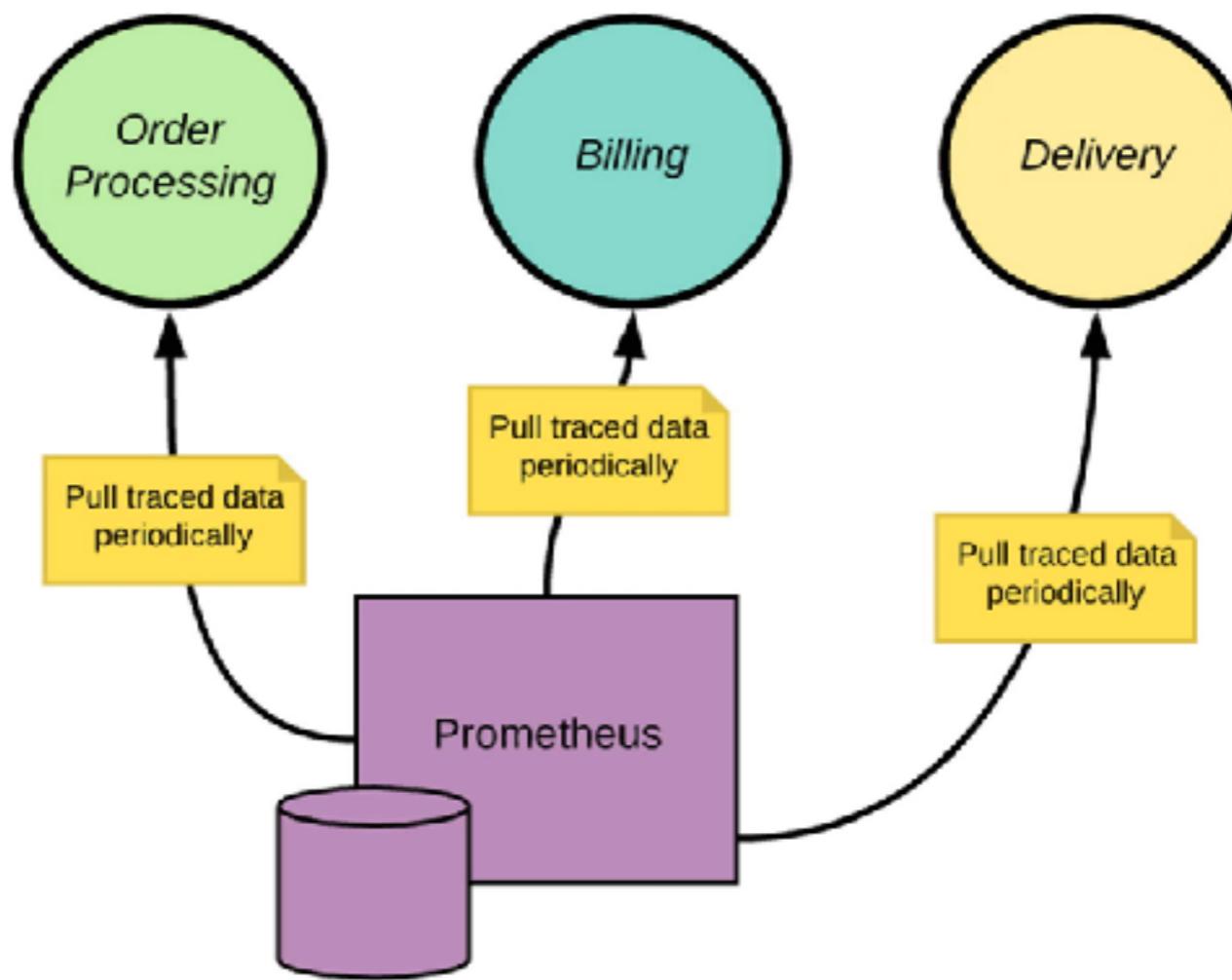


<https://www.consul.io/>



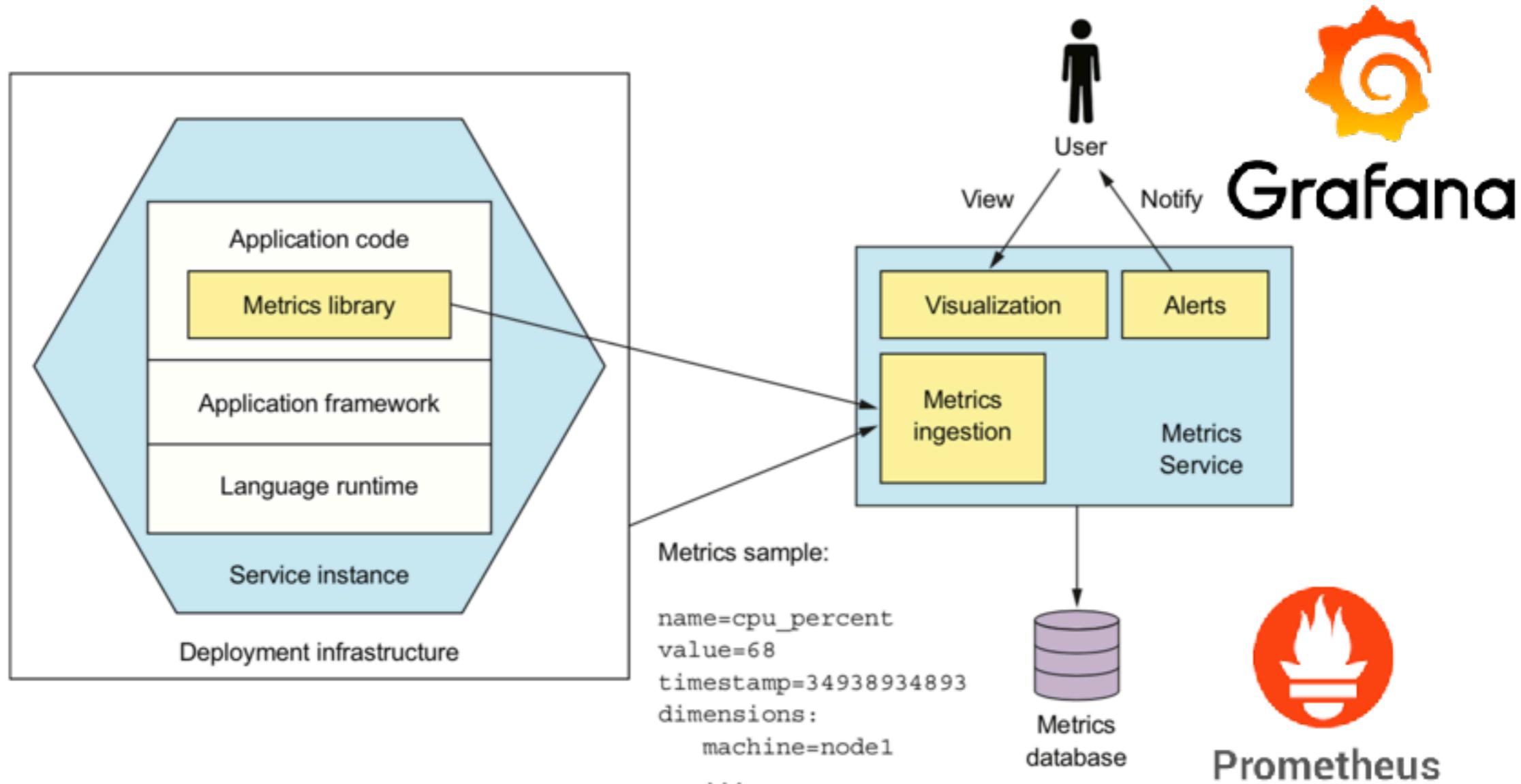
Application metrics

Services maintain metrics and expose to metric server
(counters, gauges)



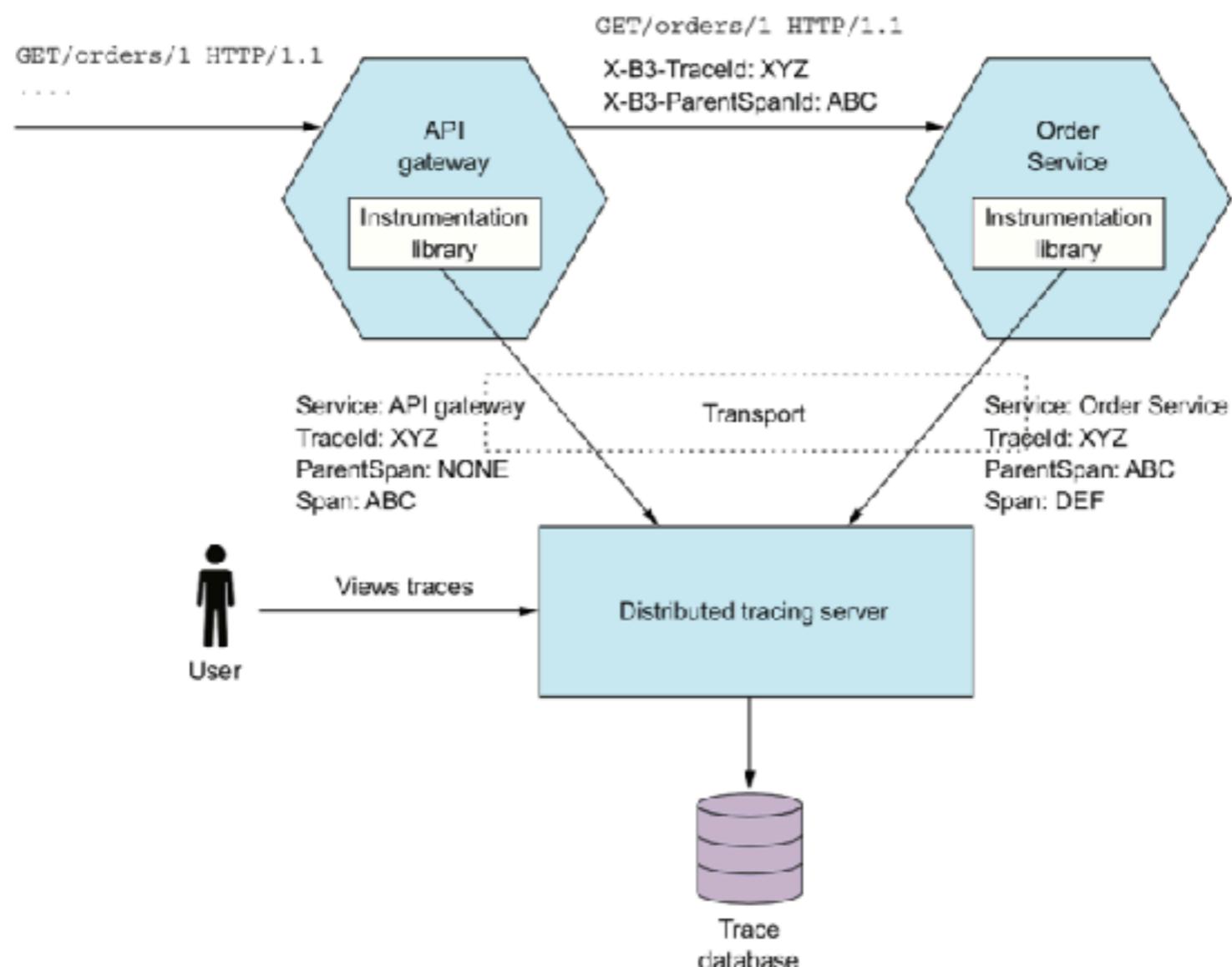
Application metrics

Services maintain metrics and expose to metric server
(counters, gauges)

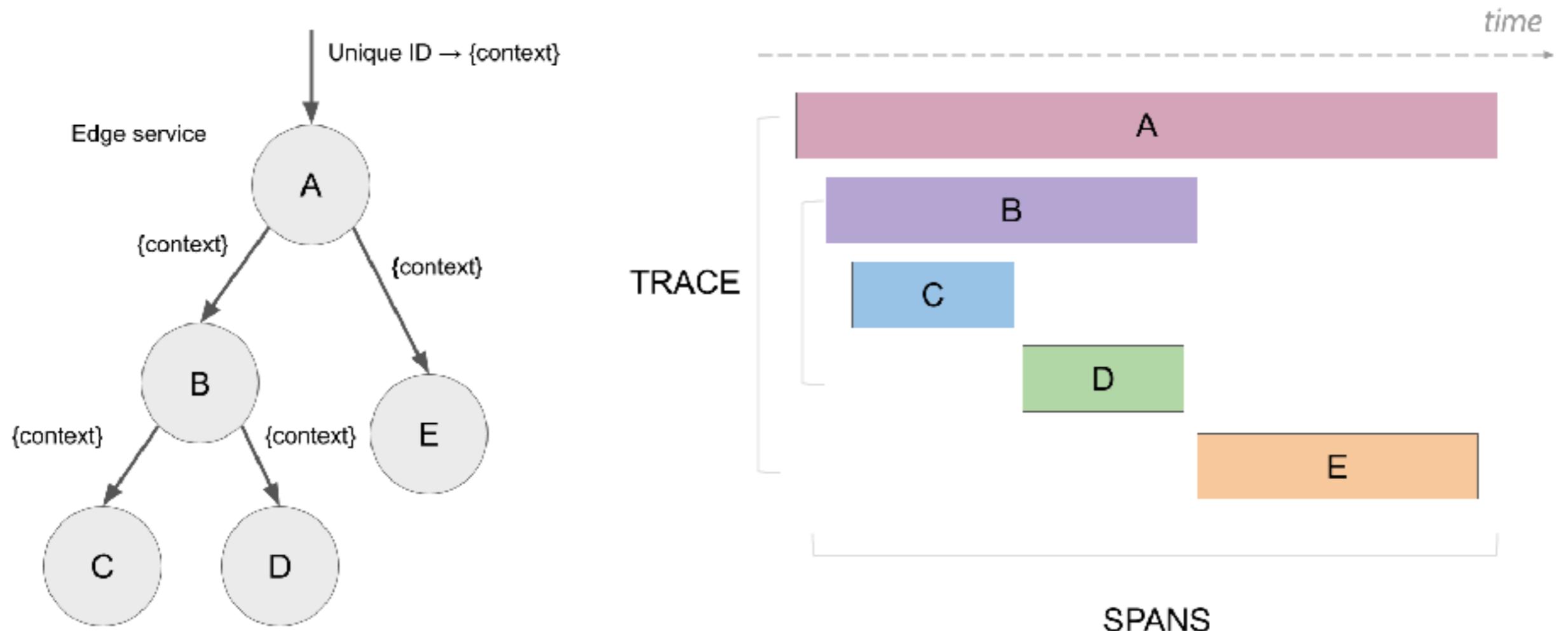


Distributed tracing

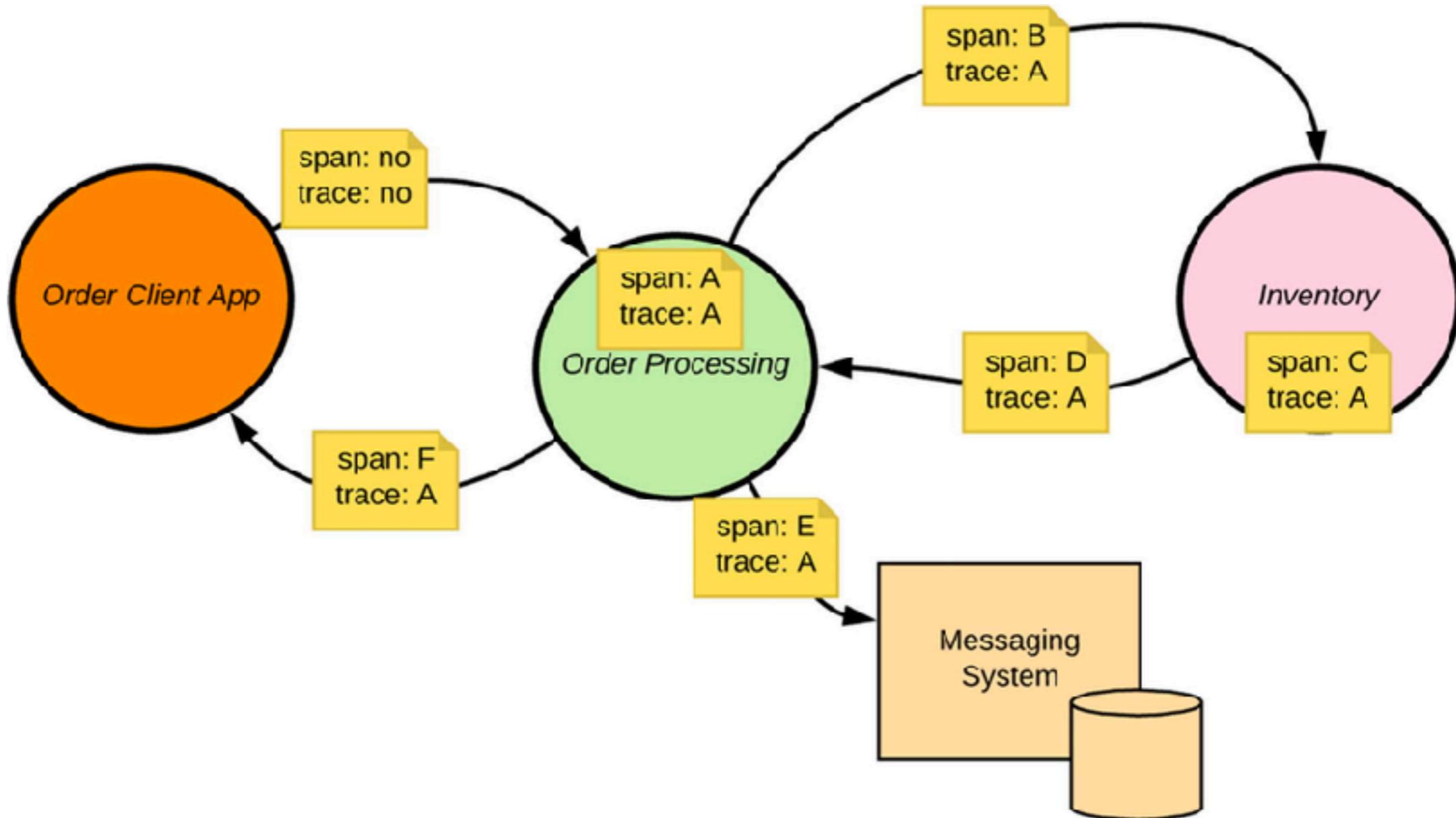
Assign each external request a **unique ID** and trace requests as flow between services



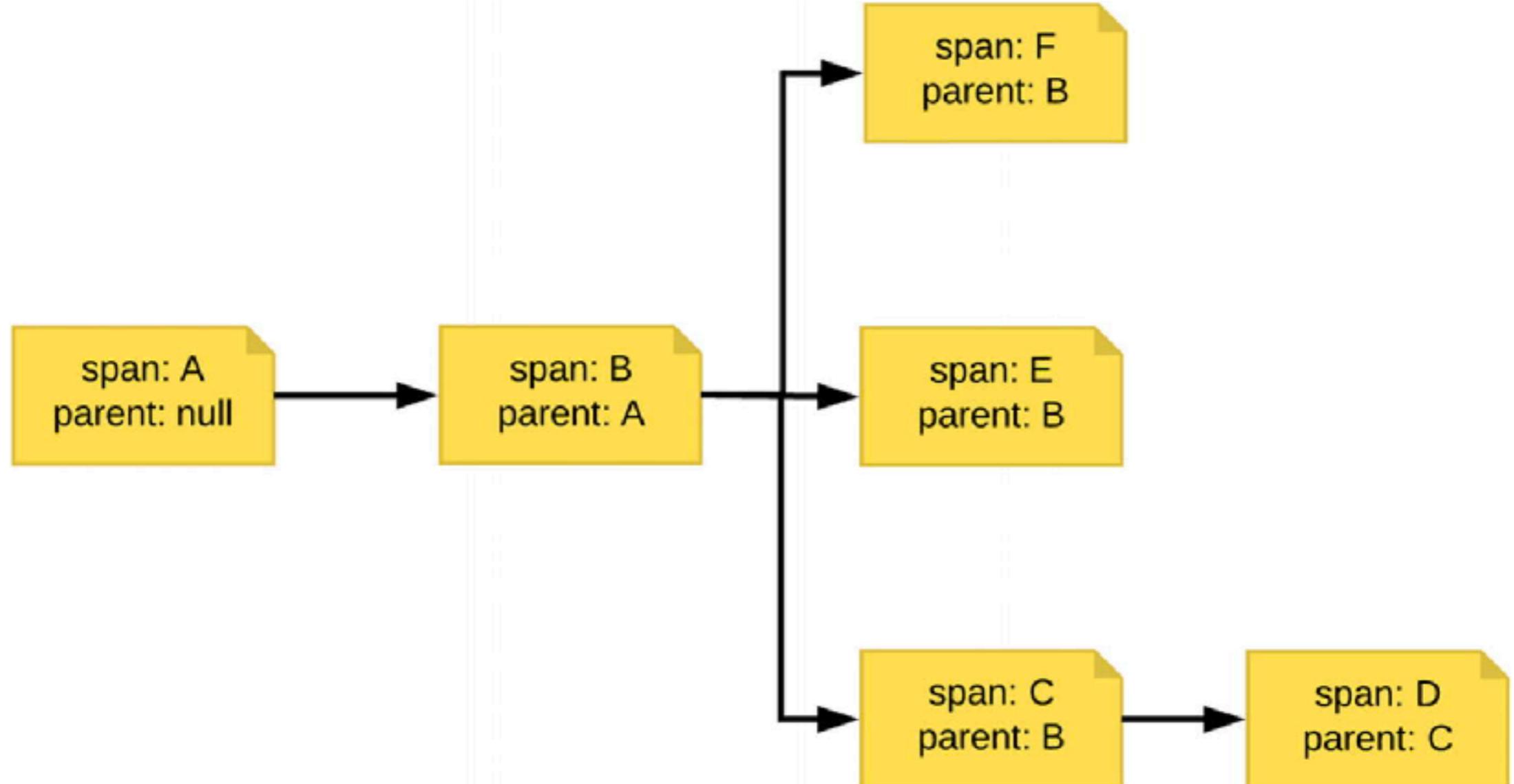
Distributed tracing



Distributed tracing



Distributed tracing



Distributed tracing tools

Format standard with OpenTelemetry
Zipkin, Jaeger, Tempo, AWS X-Ray, Elastic APM



JAEGER



Grafana Tempo

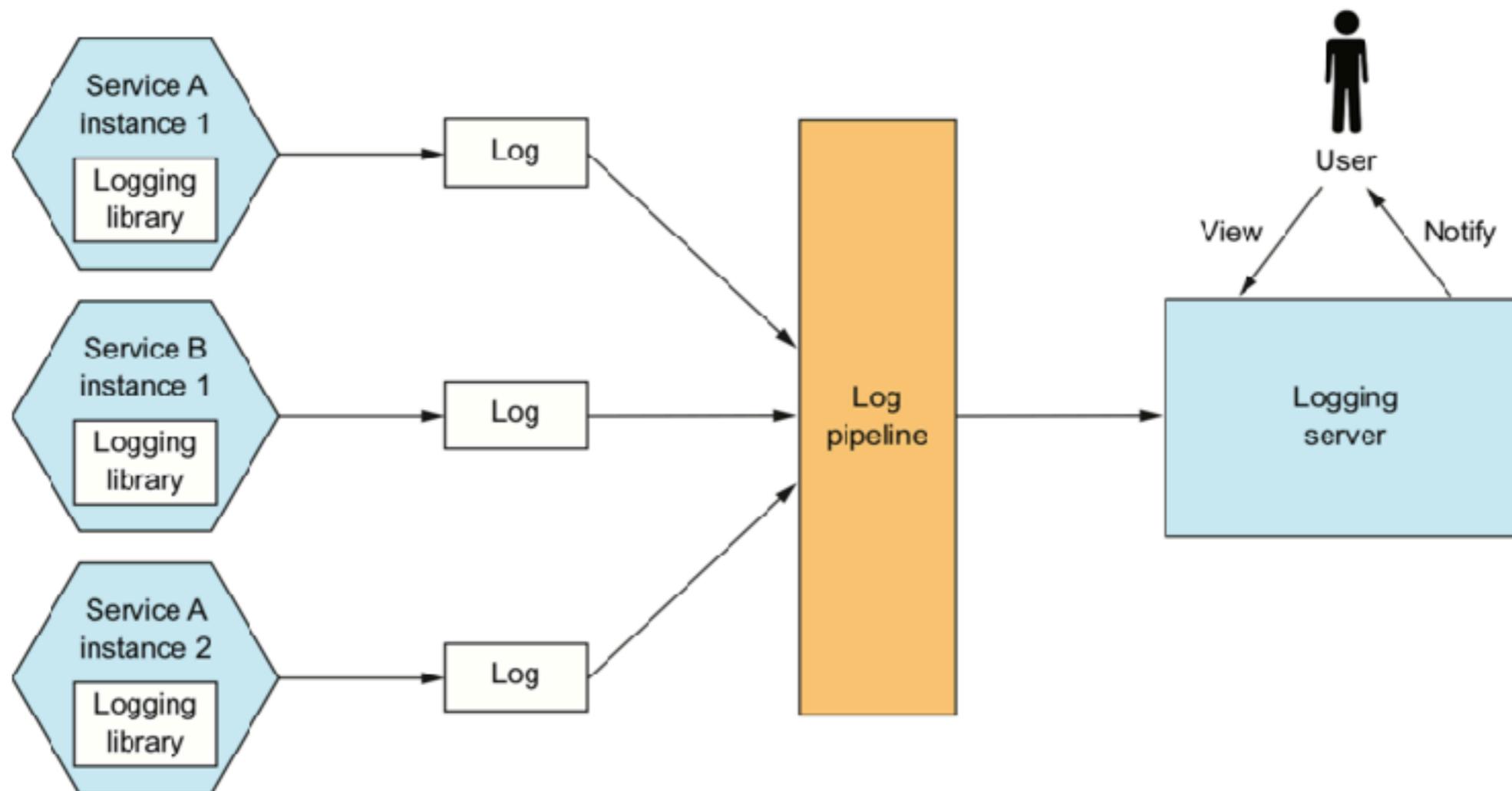


OpenTelemetry



Log aggregation

Log service activity and write logs into a centralized logging server. (searching, alerting)



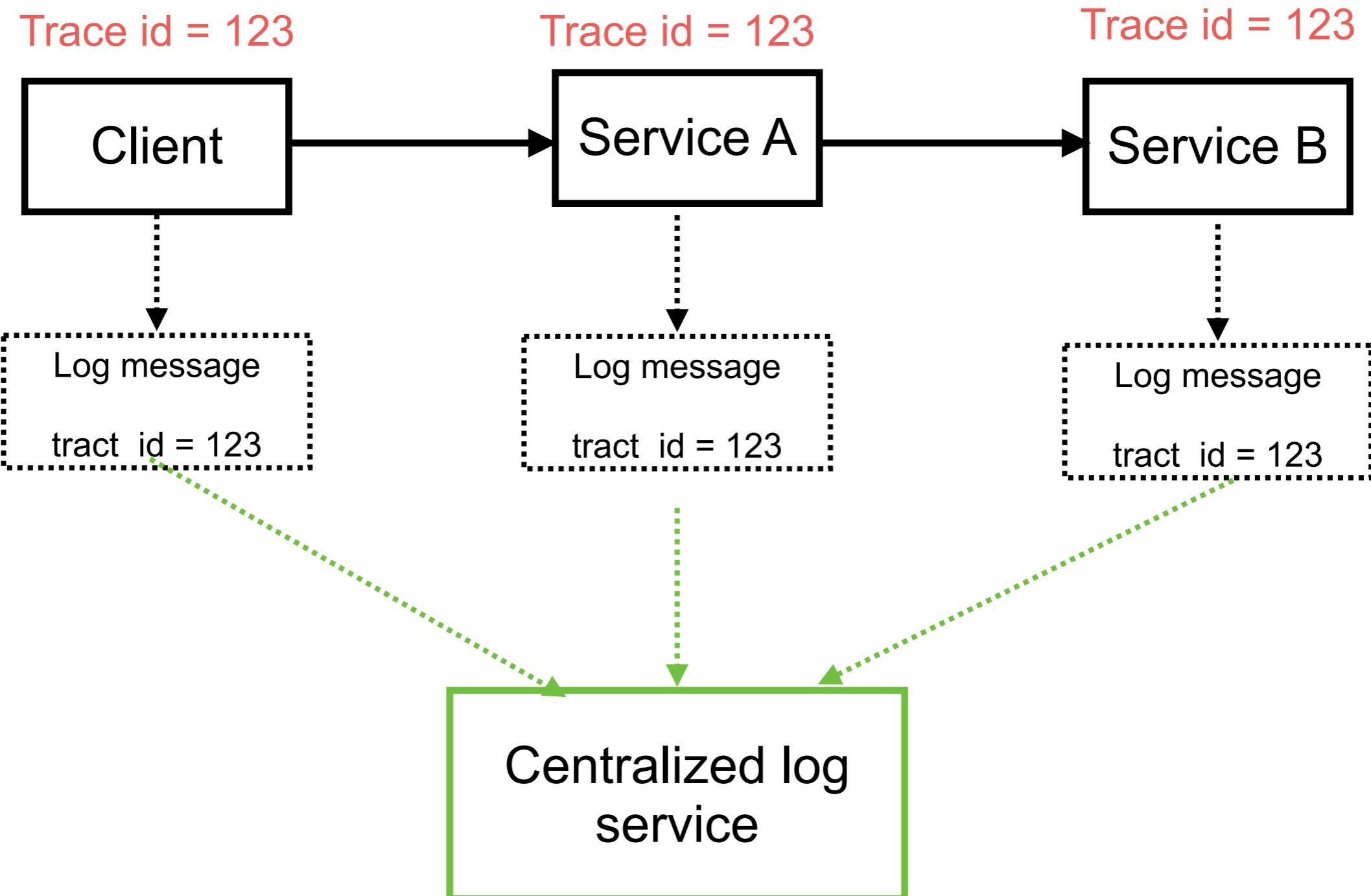
Effective Log Aggregation

- Define event to log
- Use structured logging
- Exclude sensitive information
- Log at the correct level
- Be specific in your message
- Don't log large message
- Make sure you keep trace Id in the log

https://github.com/OWASP/CheatSheetSeries/blob/master/cheatsheets/Logging_Cheat_Sheet.md



Log aggregation



Consistent Structure across all logs

Property	Description	Example
Timestamp	Date and time of the log	2023-07-01
Log level	DEBUG, INFO, ERROR	
Trace Id or Correlation Id	Unique identifier that refer to other logs from all services	
Event/Action Name	Identify to event or action of log	Authentication fail
Service ID/ Name	Identify to service	
Request path	Path for the request	/api/products



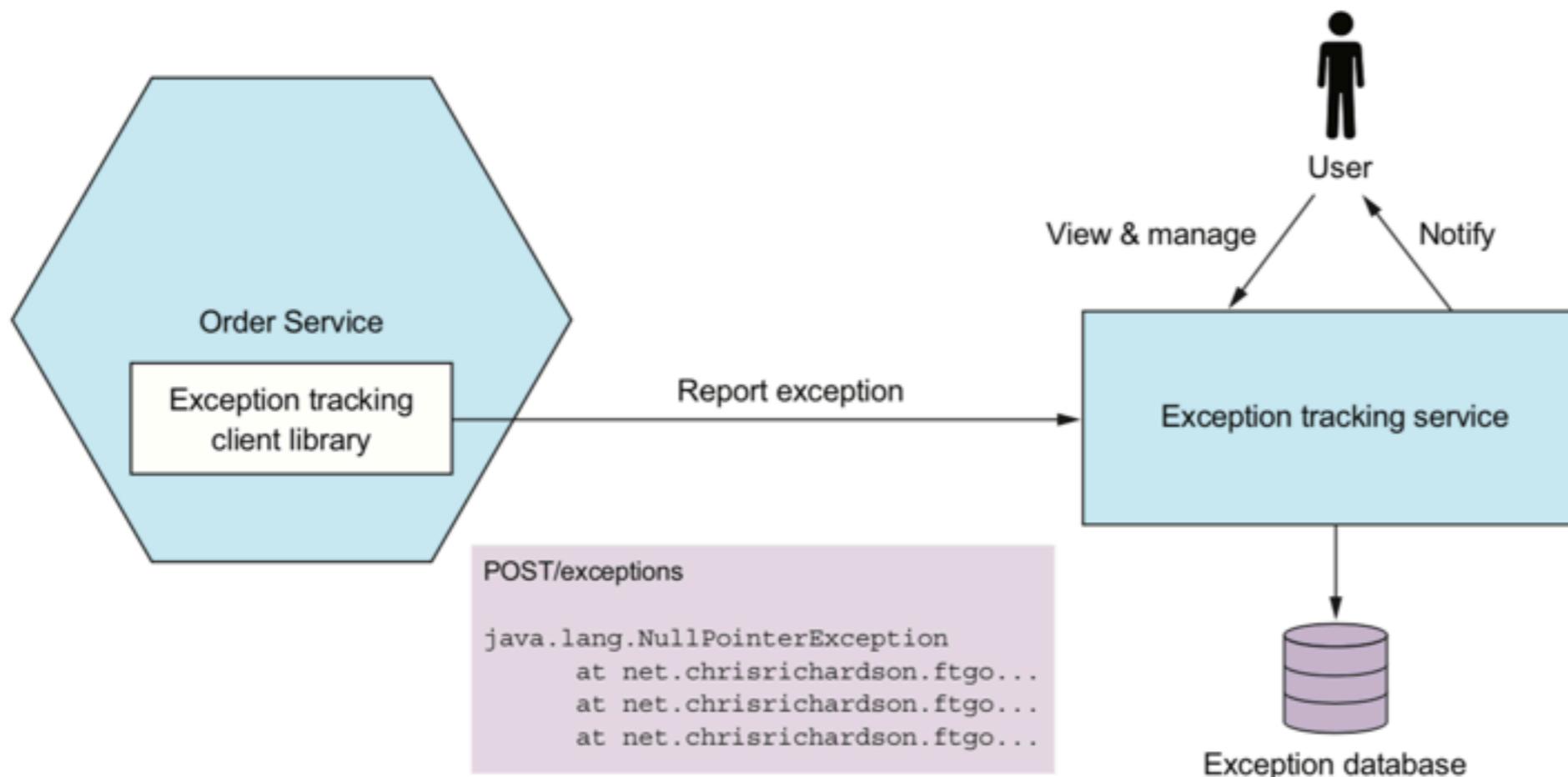
Don't keep !!

```
com.framework.FrameworkException: Error in web request
  at com.framework.ApplicationStarter.lambda$start$0(ApplicationStarter.java:15)
  at spark.RouteImpl$1.handle(RouteImpl.java:72)
  at spark.http.matching.Routes.execute(Routes.java:61)
  at spark.http.matching.MatcherFilter.doFilter(MatcherFilter.java:134)
  at spark.embeddedserver.jetty.JettyHandler.doHandle(JettyHandler.java:50)
  at org.eclipse.jetty.server.session.SessionHandler.doScope(SessionHandler.java:1568)
  at org.eclipse.jetty.server.handler.ScopedHandler.handle(ScopedHandler.java:144)
  at org.eclipse.jetty.server.handler.HandlerWrapper.handle(HandlerWrapper.java:132)
  at org.eclipse.jetty.server.Server.handle(Server.java:503)
  at org.eclipse.jetty.server.HttpChannel.handle(HttpChannel.java:364)
  at org.eclipse.jetty.server.HttpConnection.onFillable(HttpConnection.java:260)
  at org.eclipse.jetty.io.AbstractConnection$ReadCallback.succeeded(AbstractConnection.java:305)
  at org.eclipse.jetty.io.FillInterest.fillable(FillInterest.java:103)
  at org.eclipse.jetty.io.ChannelEndPoint$2.run(ChannelEndPoint.java:118)
  at org.eclipse.jetty.util.thread.QueuedThreadPool.runJob(QueuedThreadPool.java:765)
  at org.eclipse.jetty.util.thread.QueuedThreadPool$2.run(QueuedThreadPool.java:683)
  at java.base/java.lang.Thread.run(Thread.java:834)
Caused by: com.project.module.MyProjectFooBarException: The number of FooBars cannot be zero
  at com.project.module.MyProject.anotherMethod(MyProject.java:20)
  at com.project.module.MyProject.someMethod(MyProject.java:12)
  at com.framework.ApplicationStarter.lambda$start$0(ApplicationStarter.java:13)
  ... 16 more
Caused by: java.lang.ArithmaticException: The denominator must not be zero
  at org.apache.commons.lang3.math.Fraction.getFraction(Fraction.java:143)
  at com.project.module.MyProject.anotherMethod(MyProject.java:18)
  ... 18 more
```



Exception tracking

Report exceptions to exception tracking service
Help to identify the root cause



Exception tracking services



Audit logging

Log of user actions

Help customer support

Ensure compliance

Detect suspicious behavior



How to implement the audit logging ?

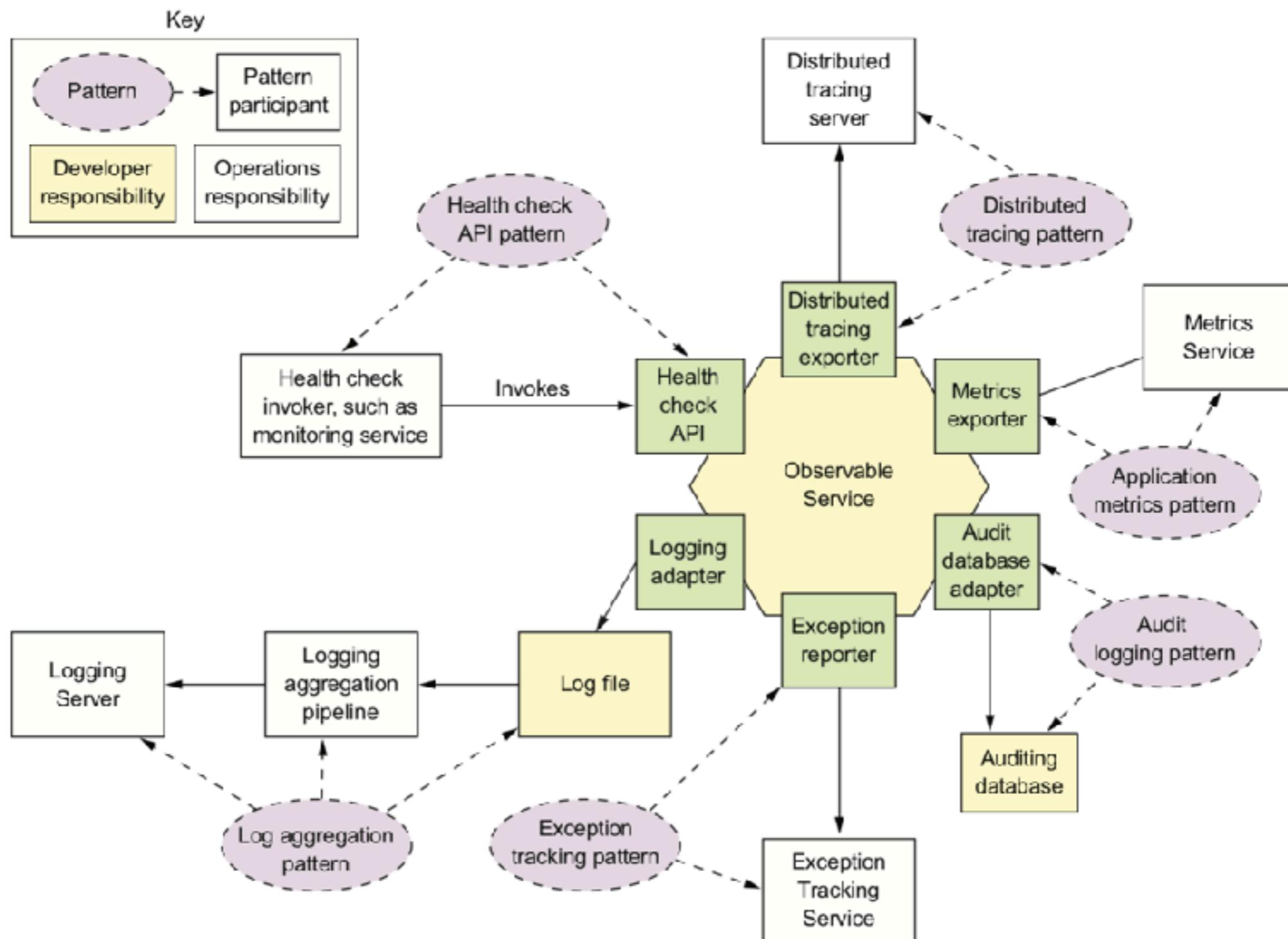
Add logging code in business logic

Use AOP (Aspect-Oriented Programming)

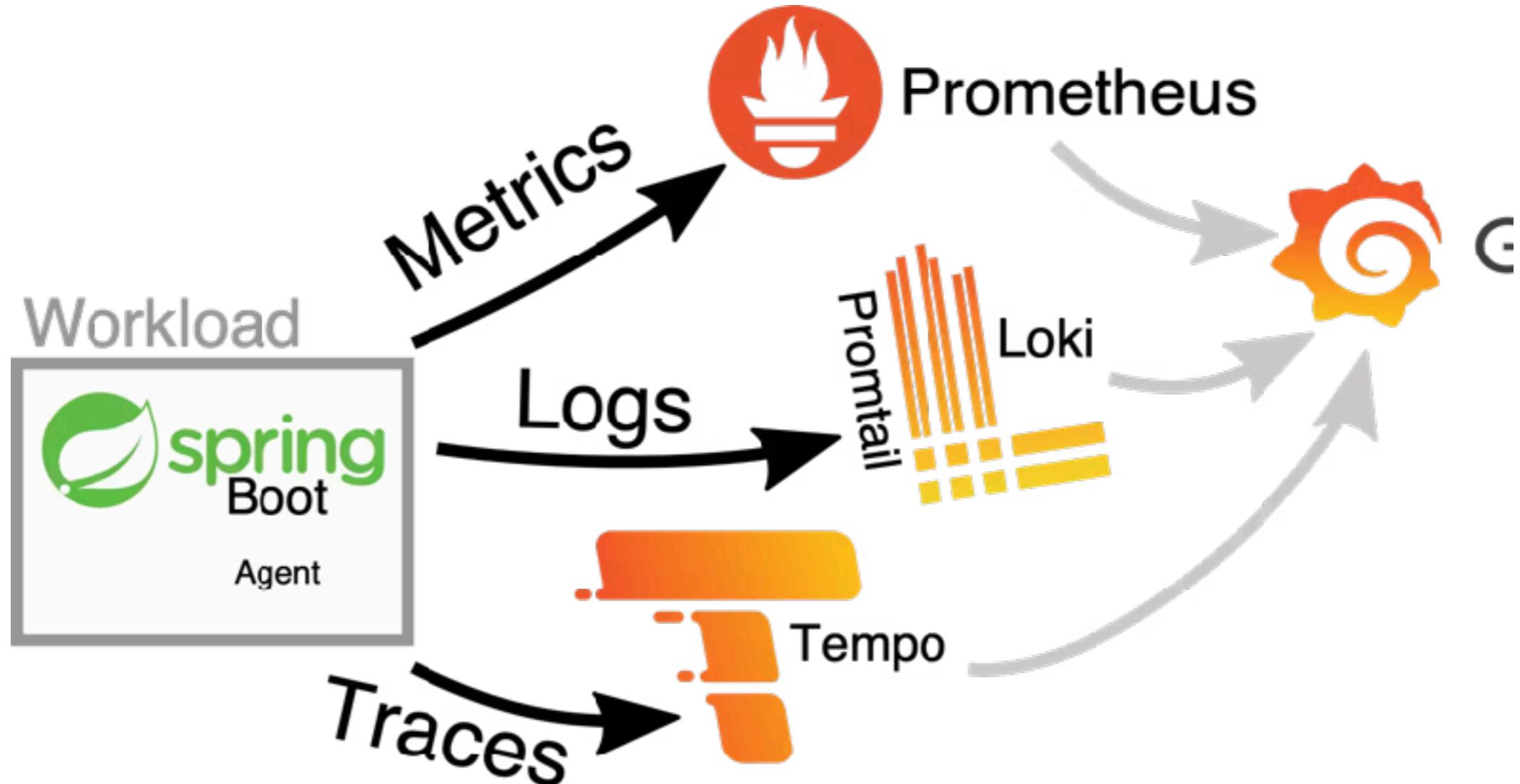
Use event sourcing



Observable services



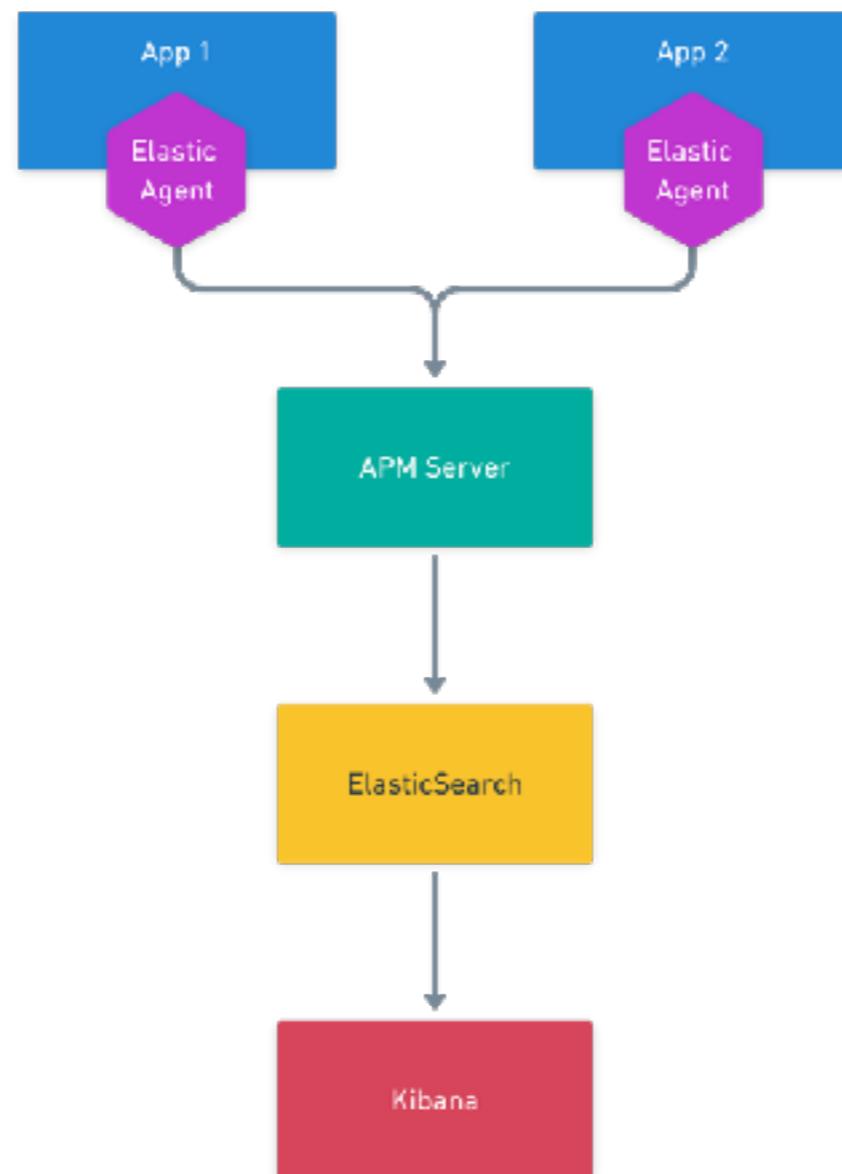
Grafana platform



<https://grafana.com/>



Elastic APM



<https://www.elastic.co/observability/application-performance-monitoring>



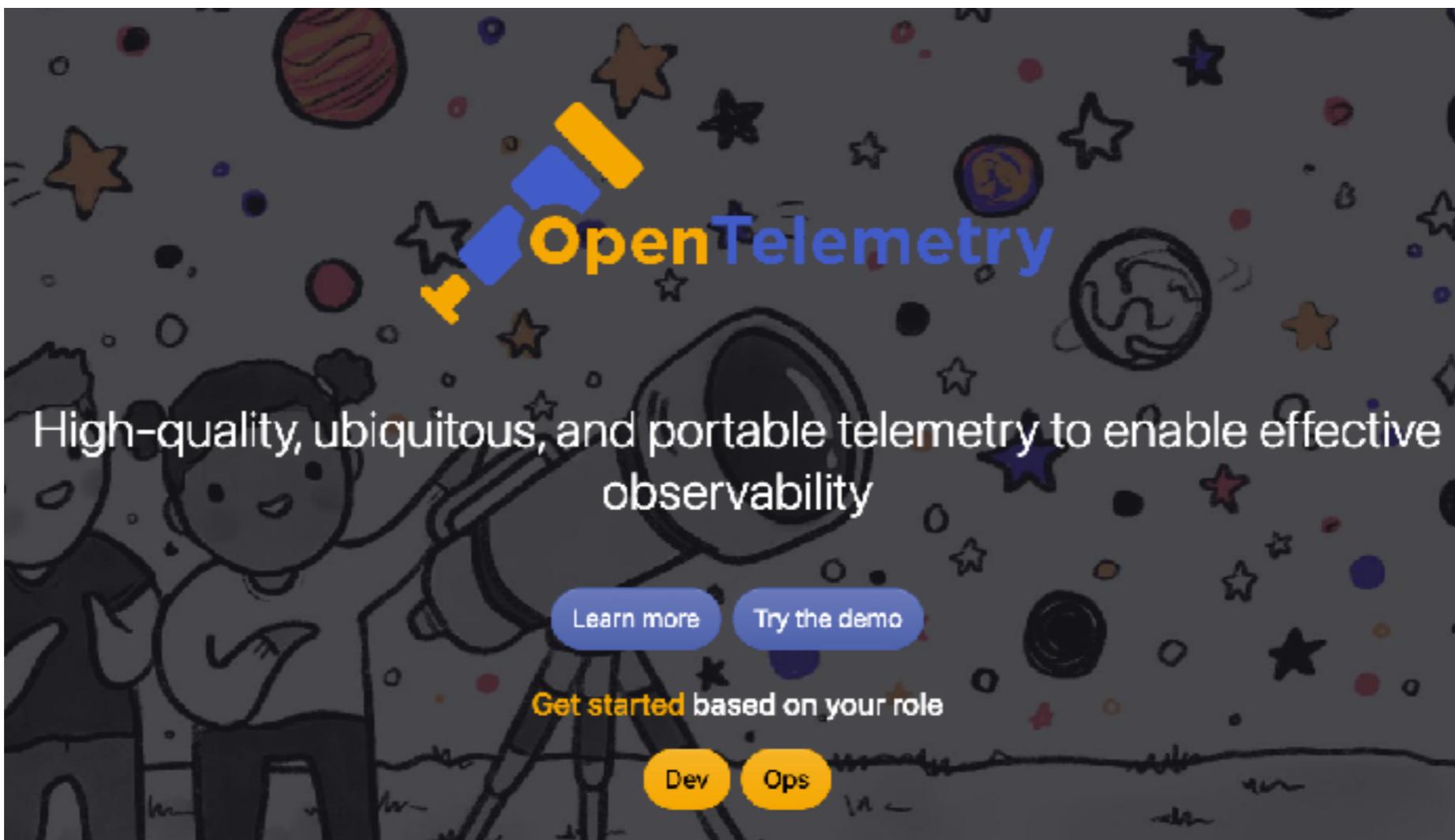
Observable service workshop

<https://github.com/up1/workshop-develop-microservices-2023>



Distributed Tracing

Working with OpenTelemetry



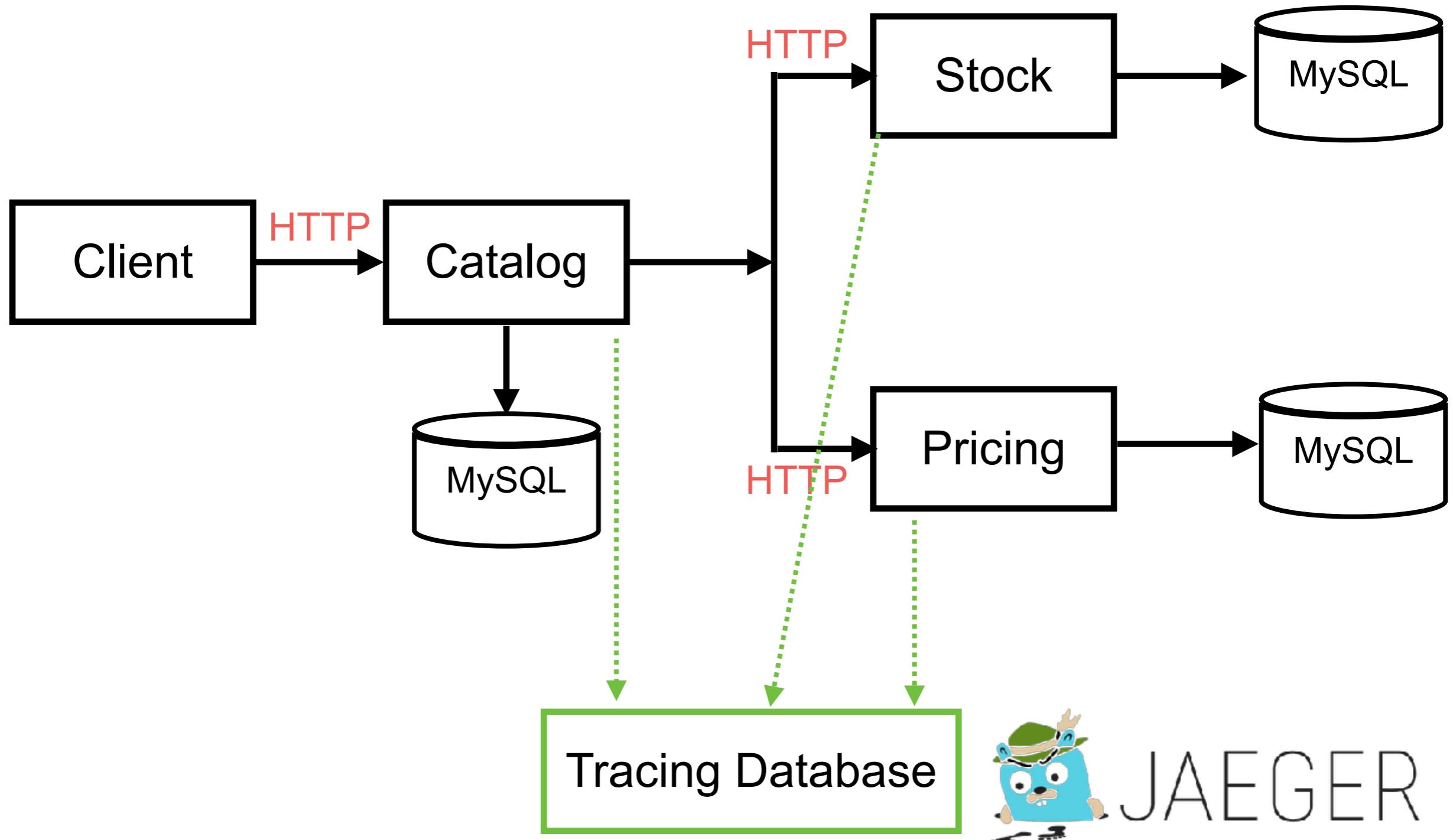
<https://opentelemetry.io/>



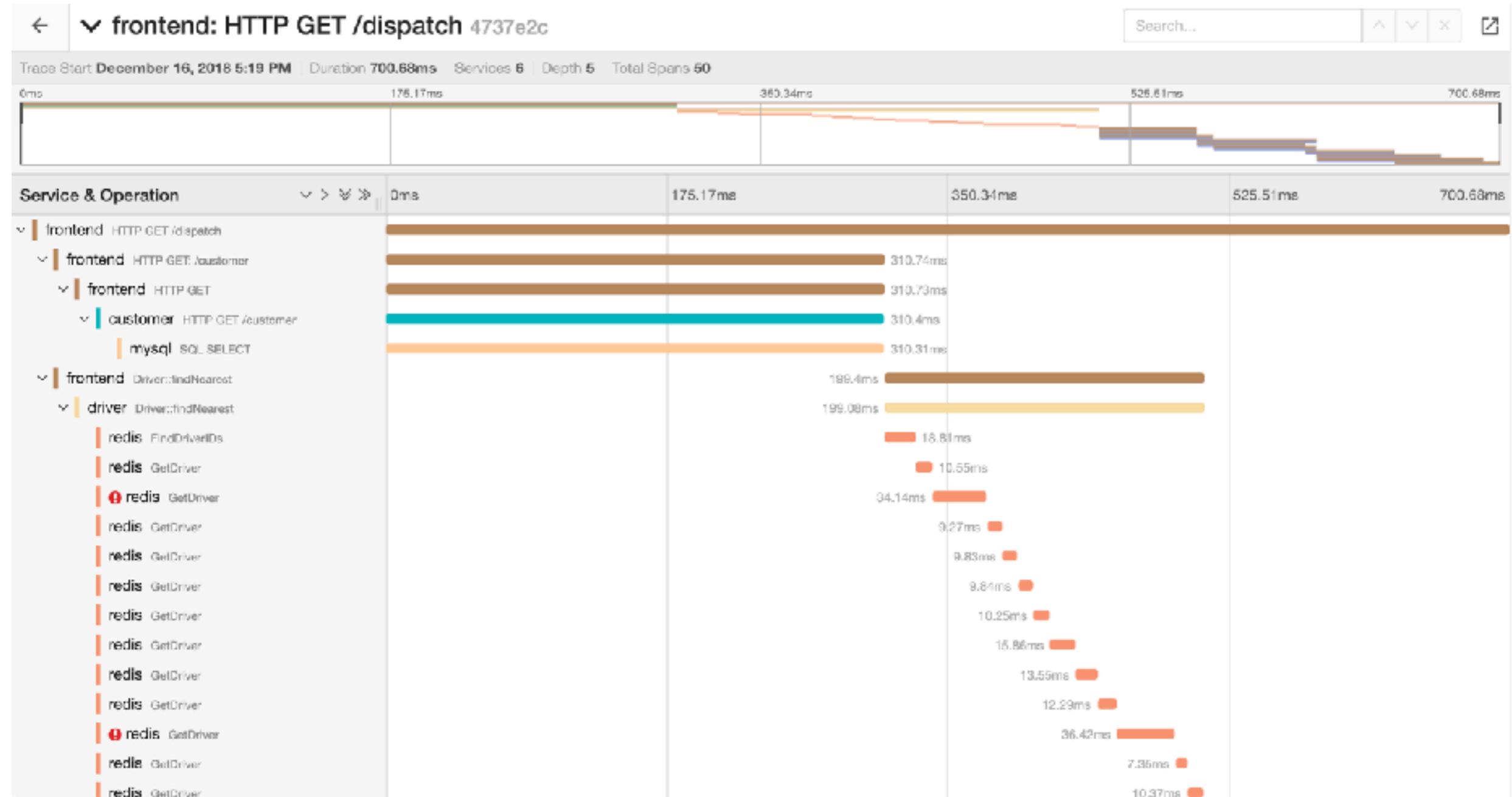
Microservices

© 2022 - 2023 Siam Chamnankit Company Limited. All rights reserved.

Distributed Tracing



Distributed Tracing with Jaeger



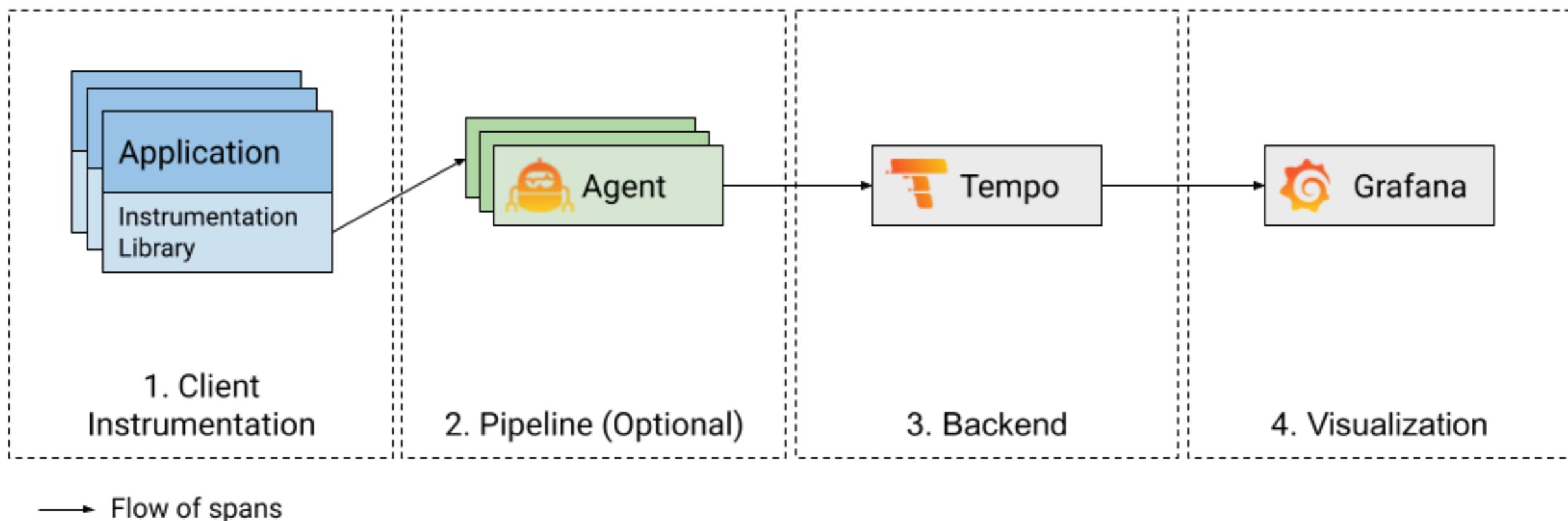
<https://www.jaegertracing.io>



Microservices

© 2022 - 2023 Siam Chamnkit Company Limited. All rights reserved.

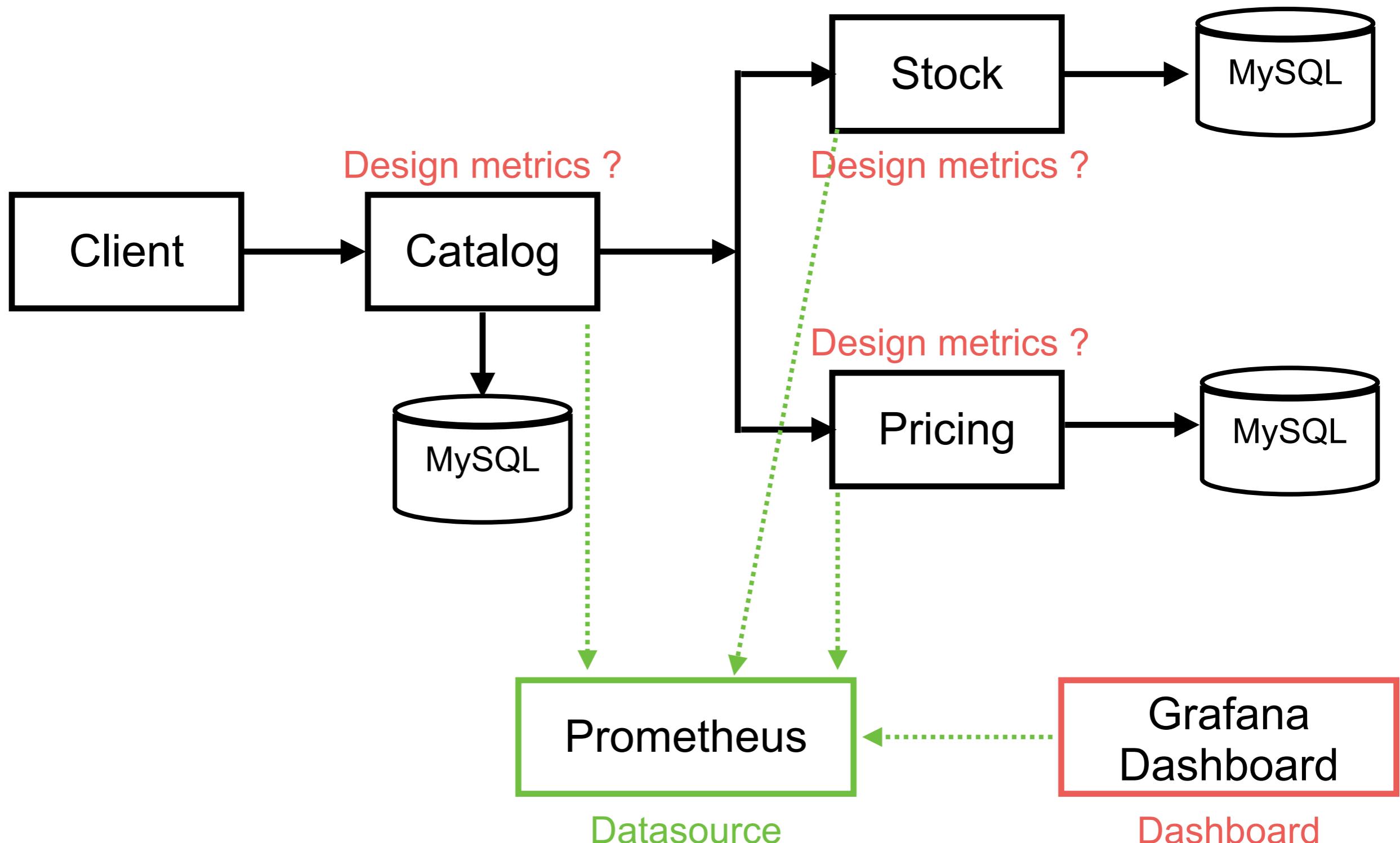
Distributed Tracing with Grafana



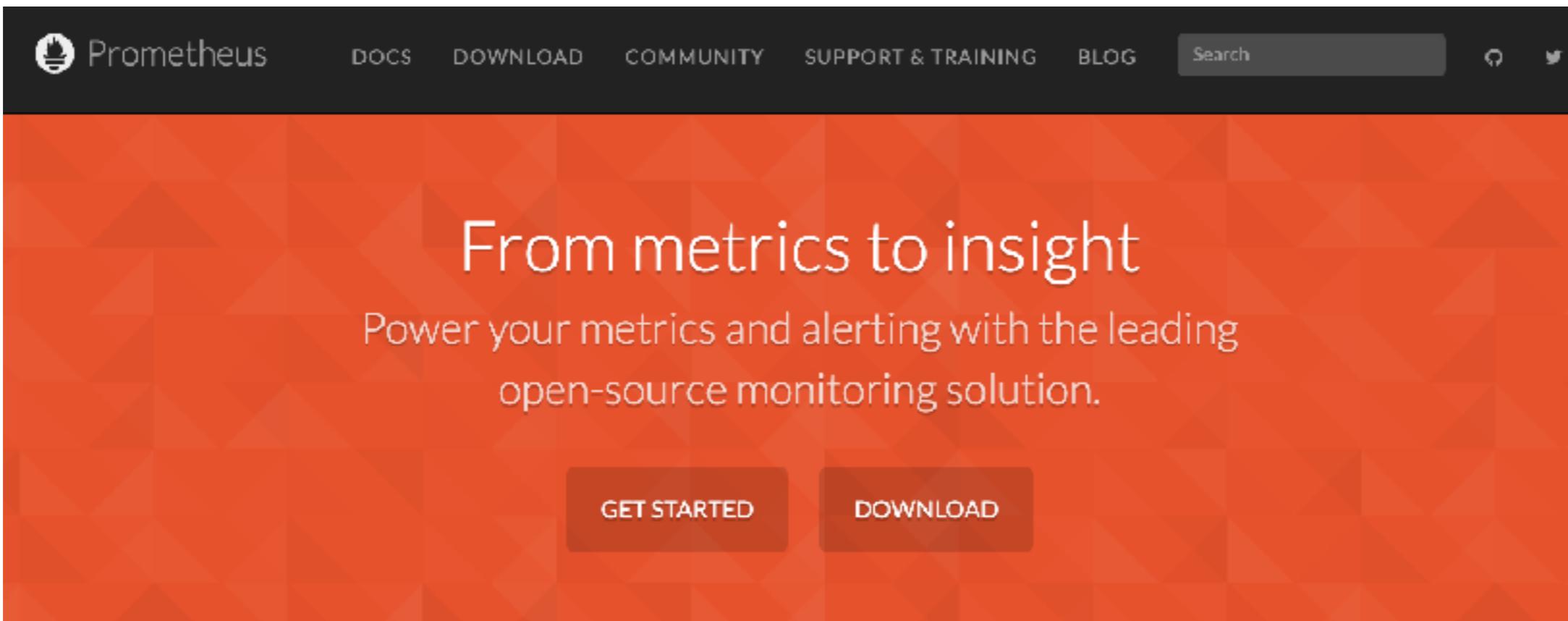
<https://www.jaegertracing.io>



Application metric



Prometheus



The banner features the Prometheus logo at the top left. A search bar and social media icons (GitHub and Twitter) are on the right. The main text "From metrics to insight" is centered above the subtitle "Power your metrics and alerting with the leading open-source monitoring solution." Below the subtitle are two buttons: "GET STARTED" and "DOWNLOAD".

From metrics to insight

Power your metrics and alerting with the leading open-source monitoring solution.

GET STARTED DOWNLOAD

Dimensional data

Prometheus implements a highly dimensional data model. Time series are identified by a metric name and a set of key-value pairs.

Powerful queries

PromQL allows slicing and dicing of collected time series data in order to generate ad-hoc graphs, tables, and alerts.

Great visualization

Prometheus has multiple modes for visualizing data: a built-in expression browser, Grafana integration, and a console template language.

Efficient storage

Prometheus stores time series in memory and on local disk in an efficient custom format. Scaling is achieved by functional sharding and

<https://prometheus.io/>



Grafana

Grafana Labs Products Open source Solutions Learn Company Downloads Contact us Sign in

Compose and scale observability with one or all pieces of the stack

Grafana

- Plugins
- Dashboards
- Alerts
- Usage insights
- Reports
- Governance

Metrics Logs Traces

API

Applications

Infrastructure

Cloud

Self-managed

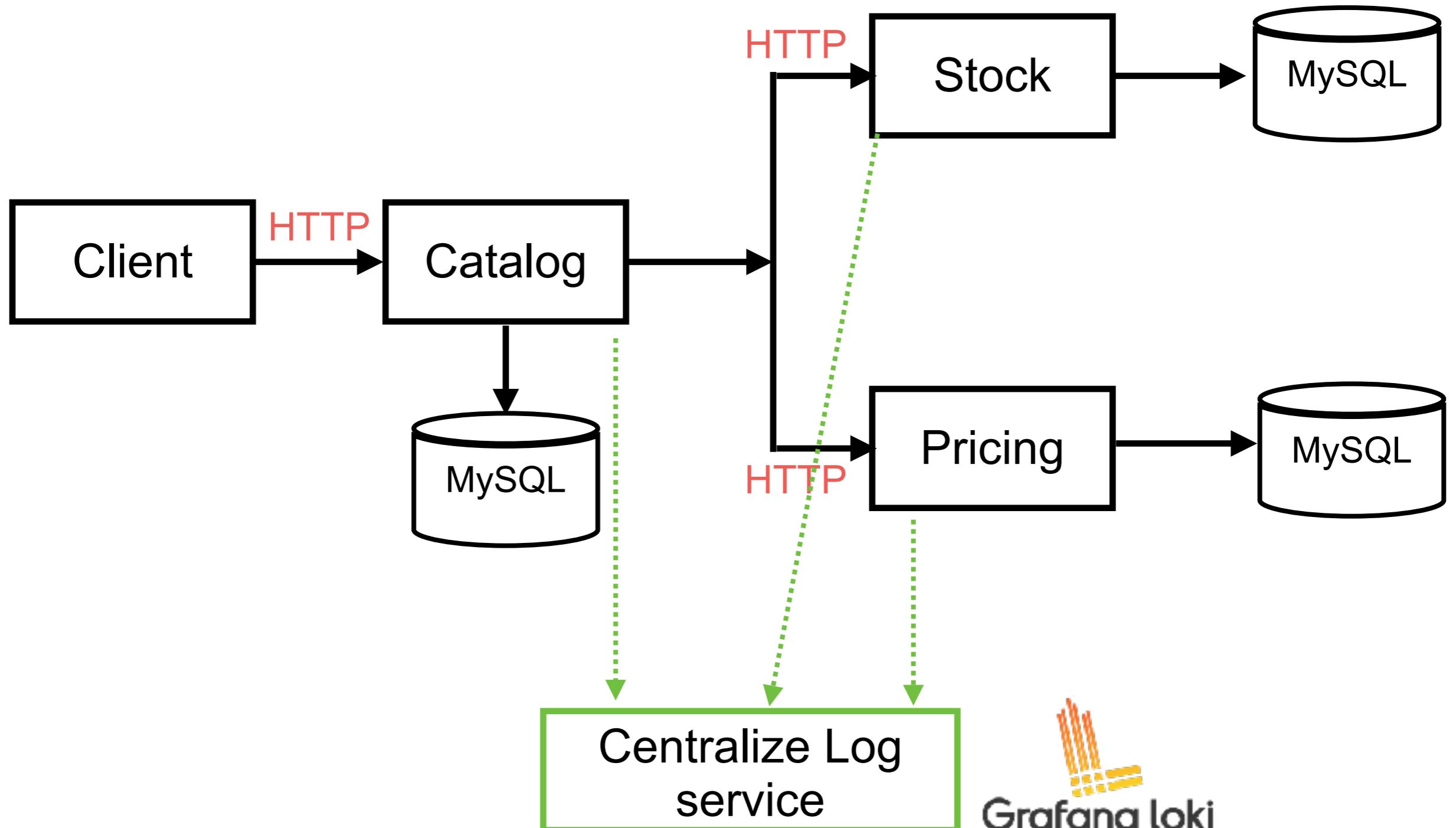
<https://grafana.com/>



Microservices

© 2022 - 2023 Siam Chamnankit Company Limited. All rights reserved.

Log aggregation



Secure services



Develop secure services

Authentication

Authorization

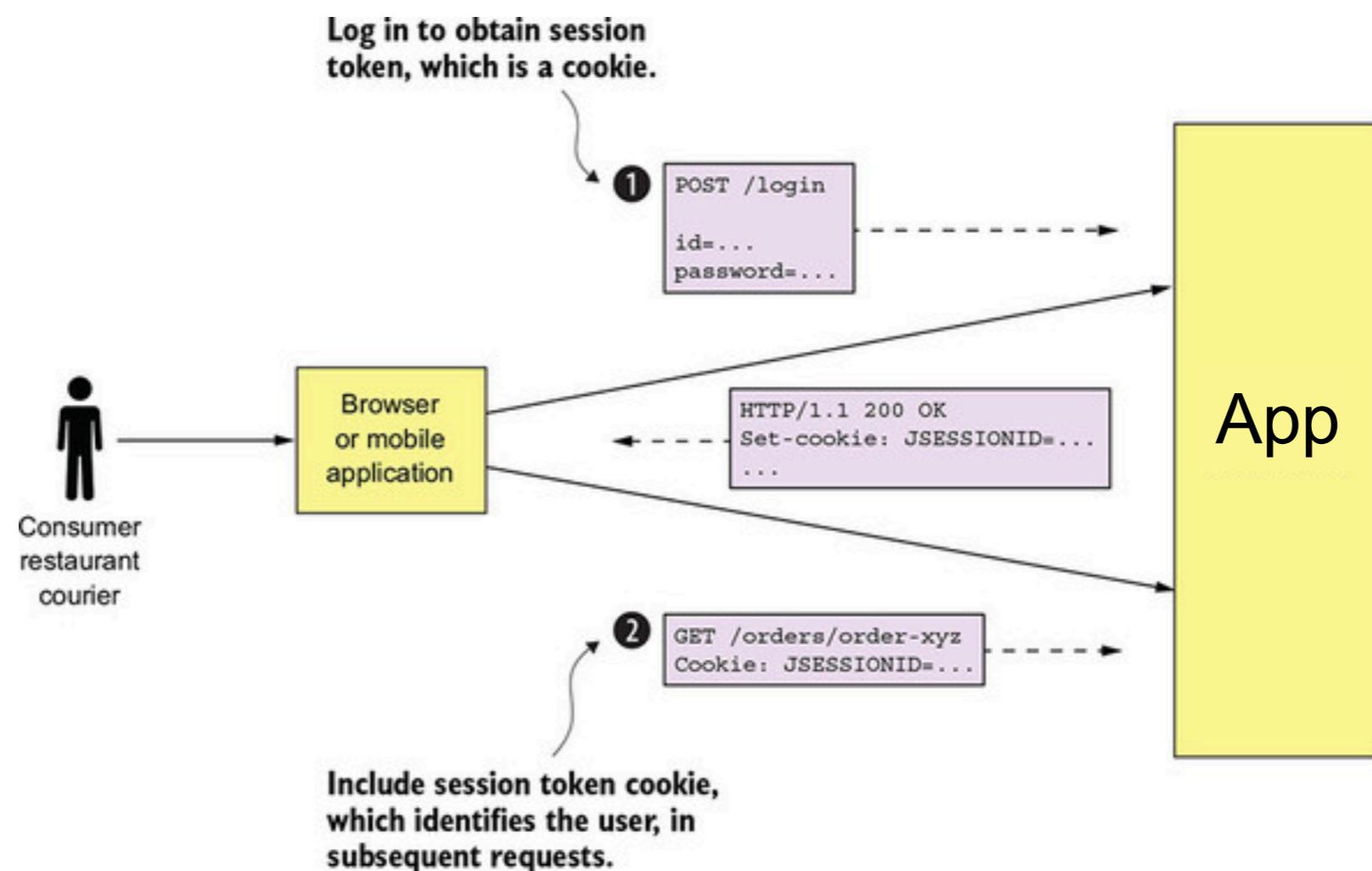
Auditing

Secure interprocess communication



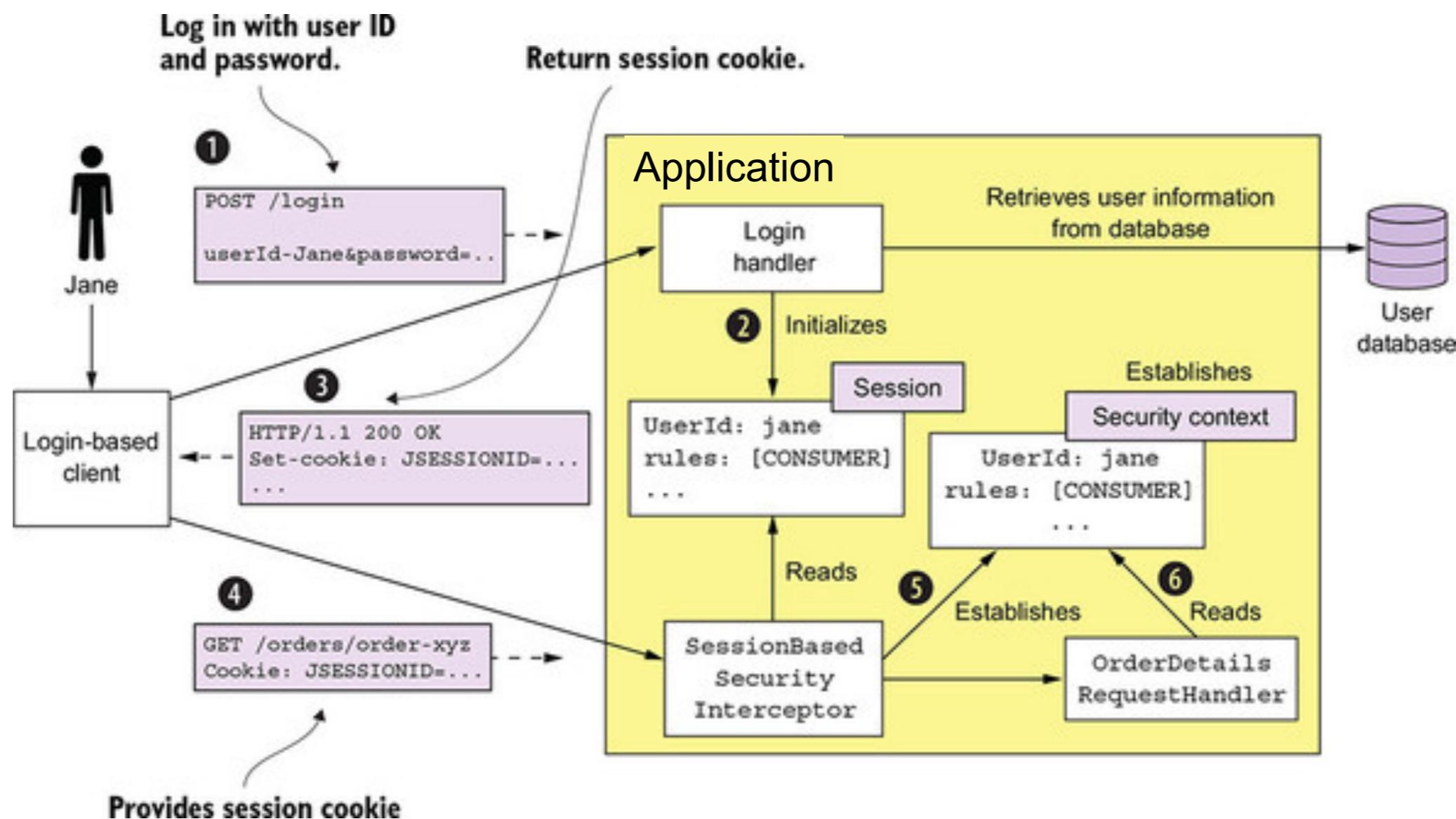
Security in traditional application

Keep security information in browser's cookie



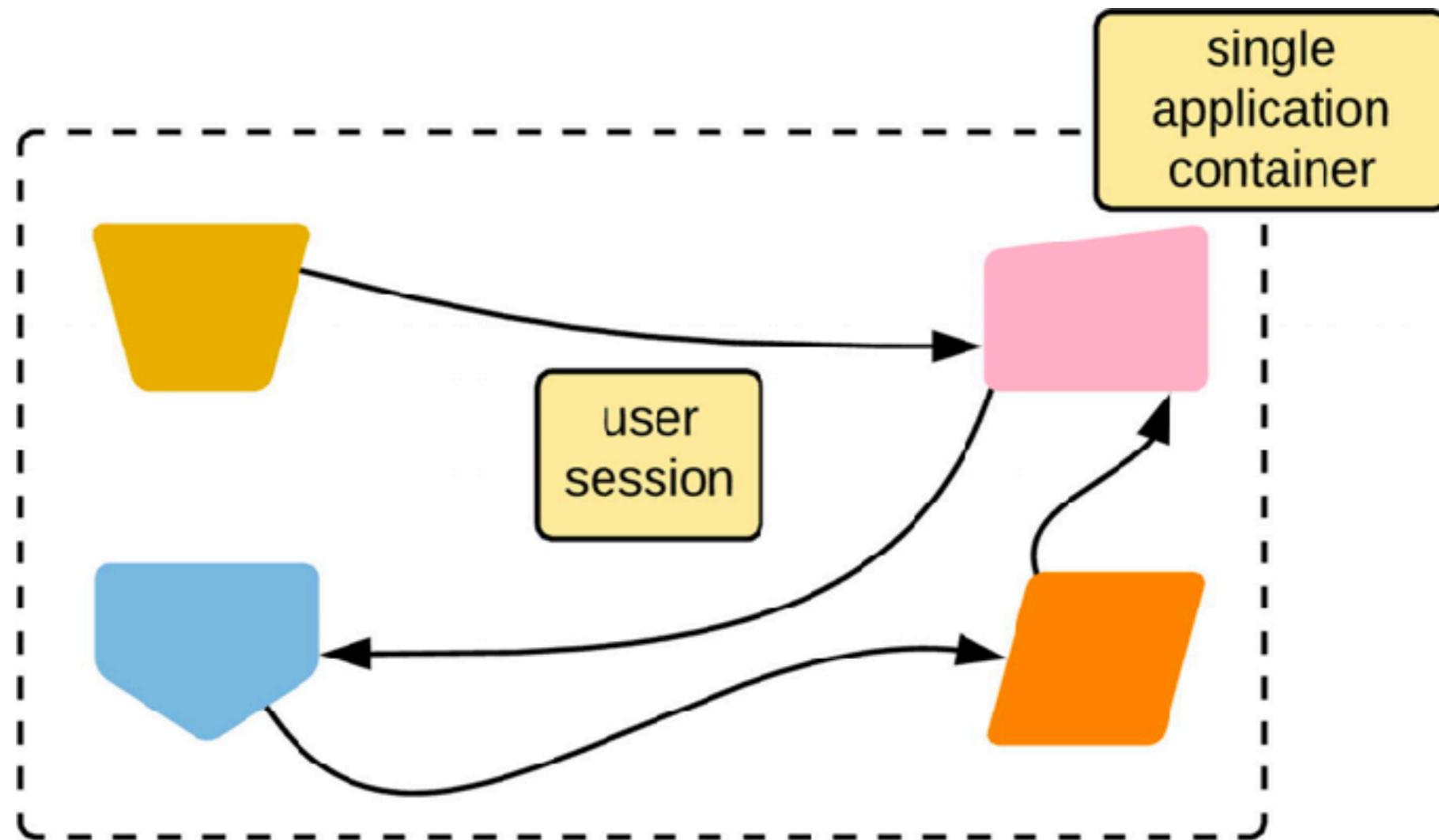
Security in traditional application

Implement security process in application



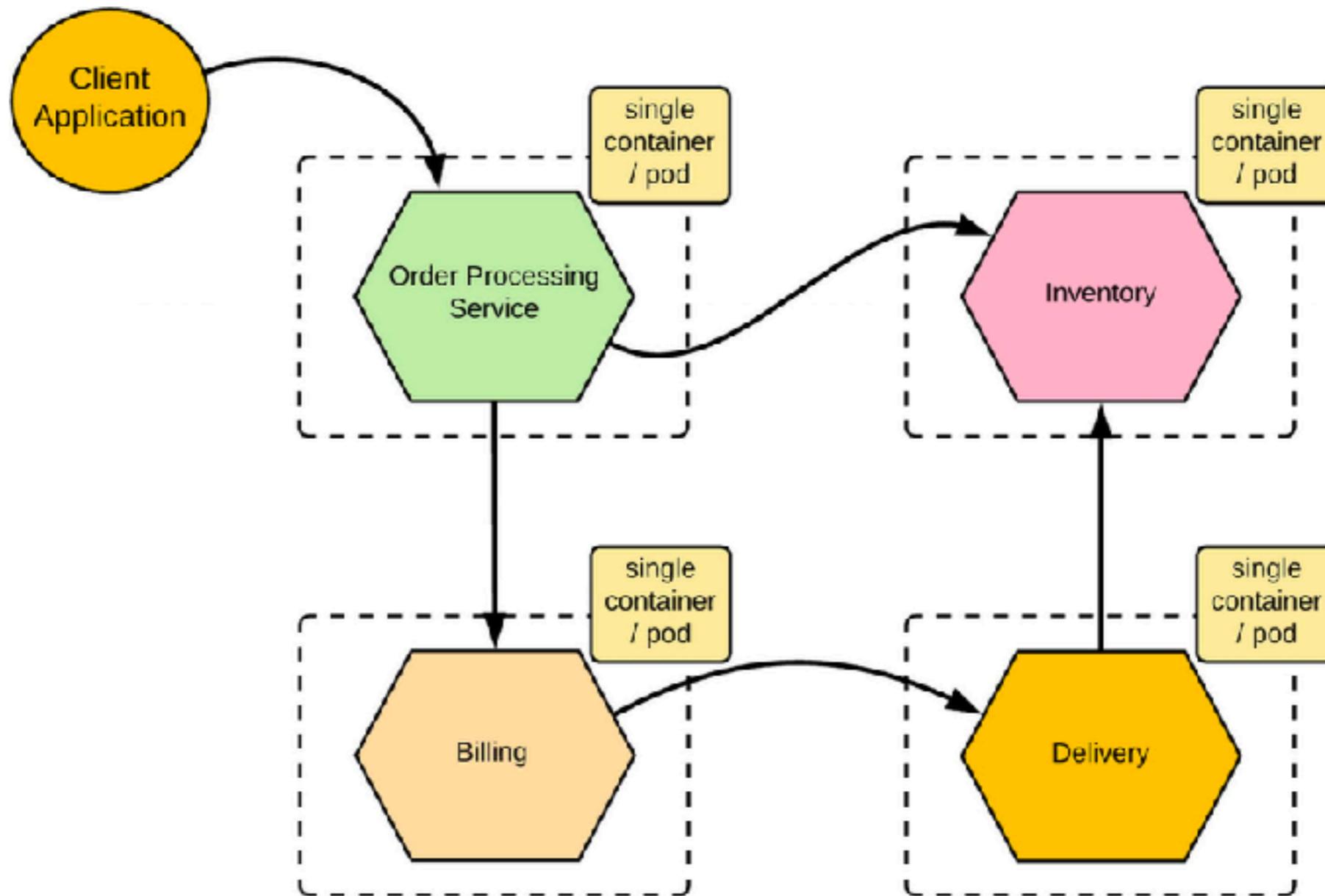
Security in traditional application

Sharing a user session in application



Security in multiple services ?

Interaction between multiple services



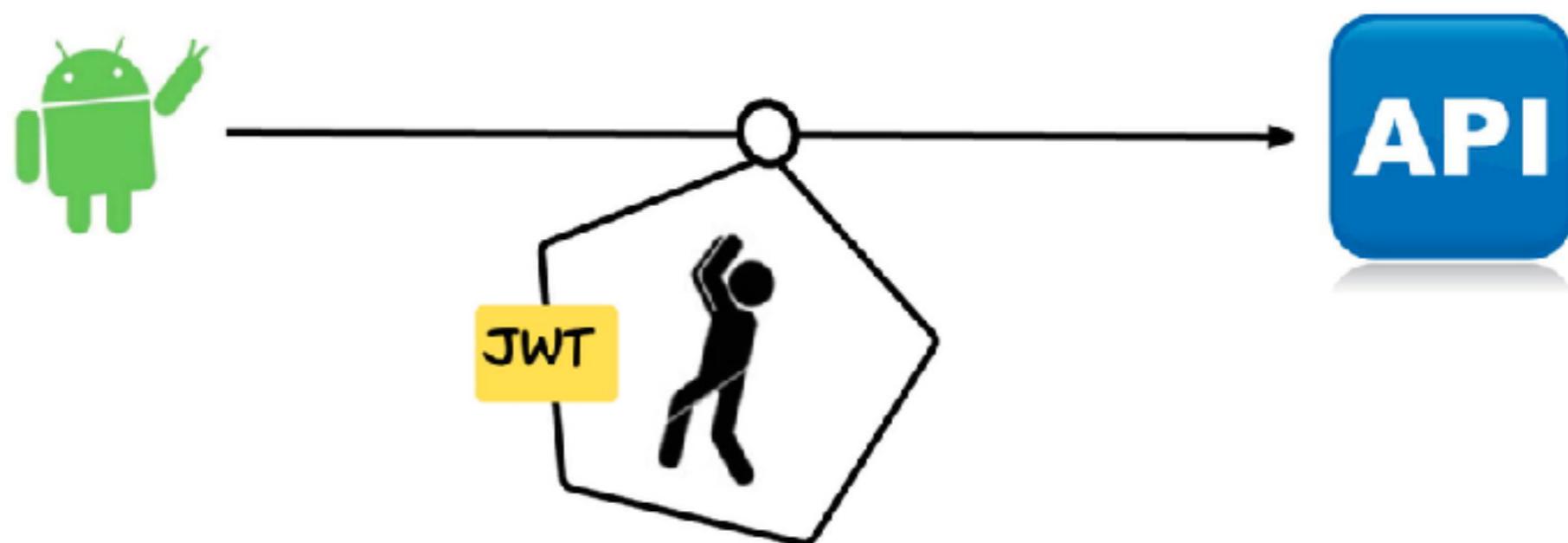
Secure service-to-service communication

JSON Web Token (JWT) + OAuth
Transport Layer Security (TLS) mutual authentication

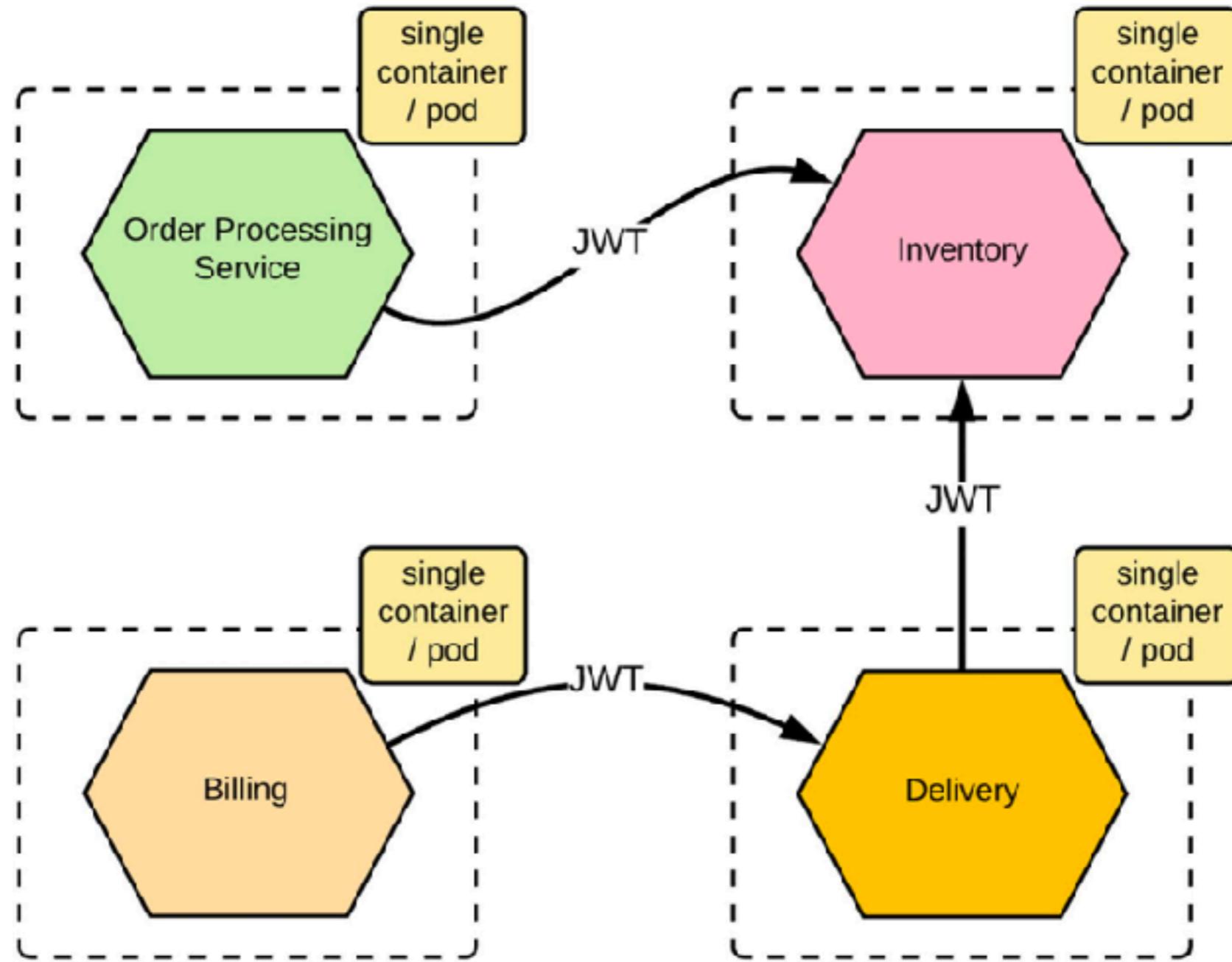


JSON Web Token (JWT)

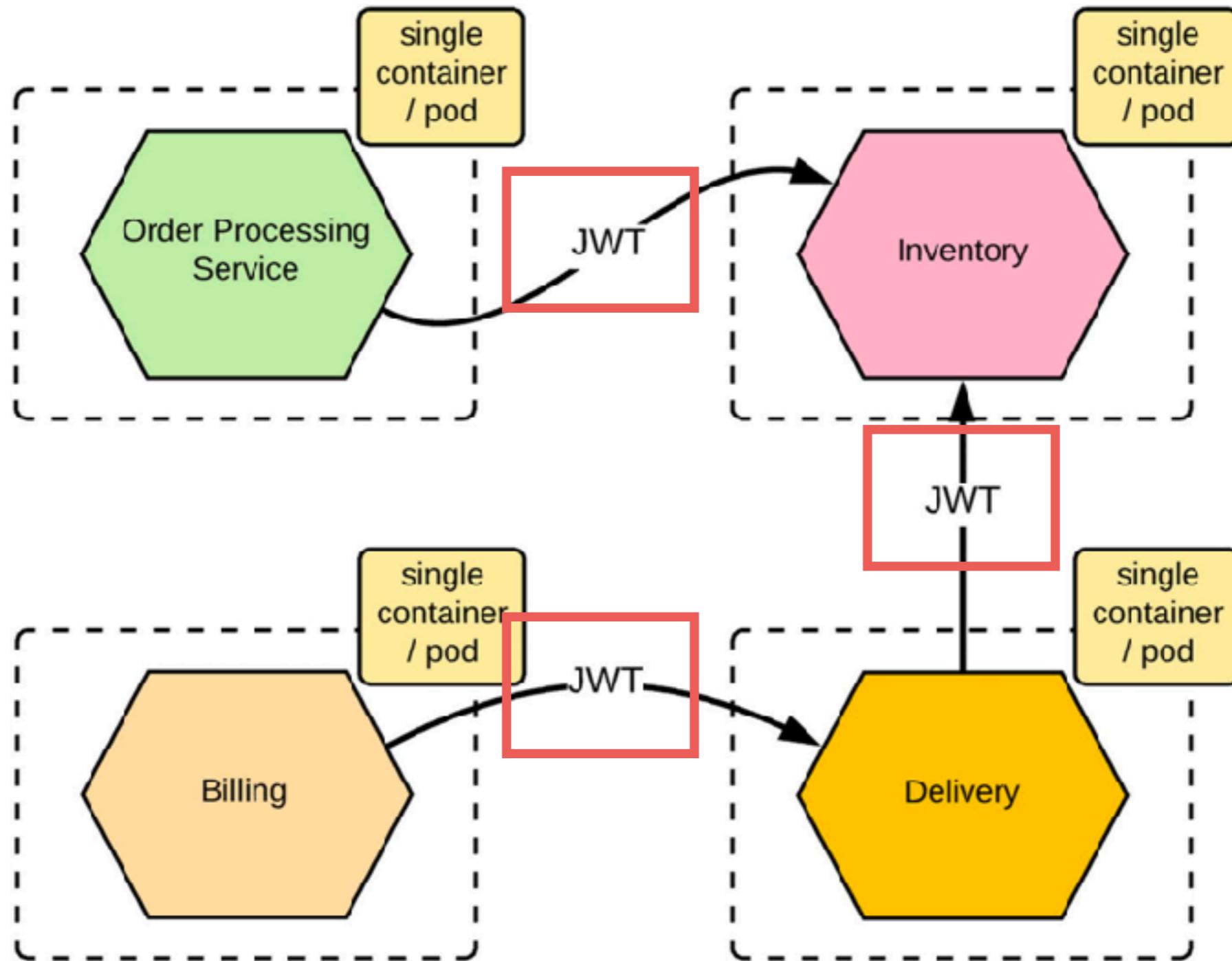
Define a container to transport data
between interested parties



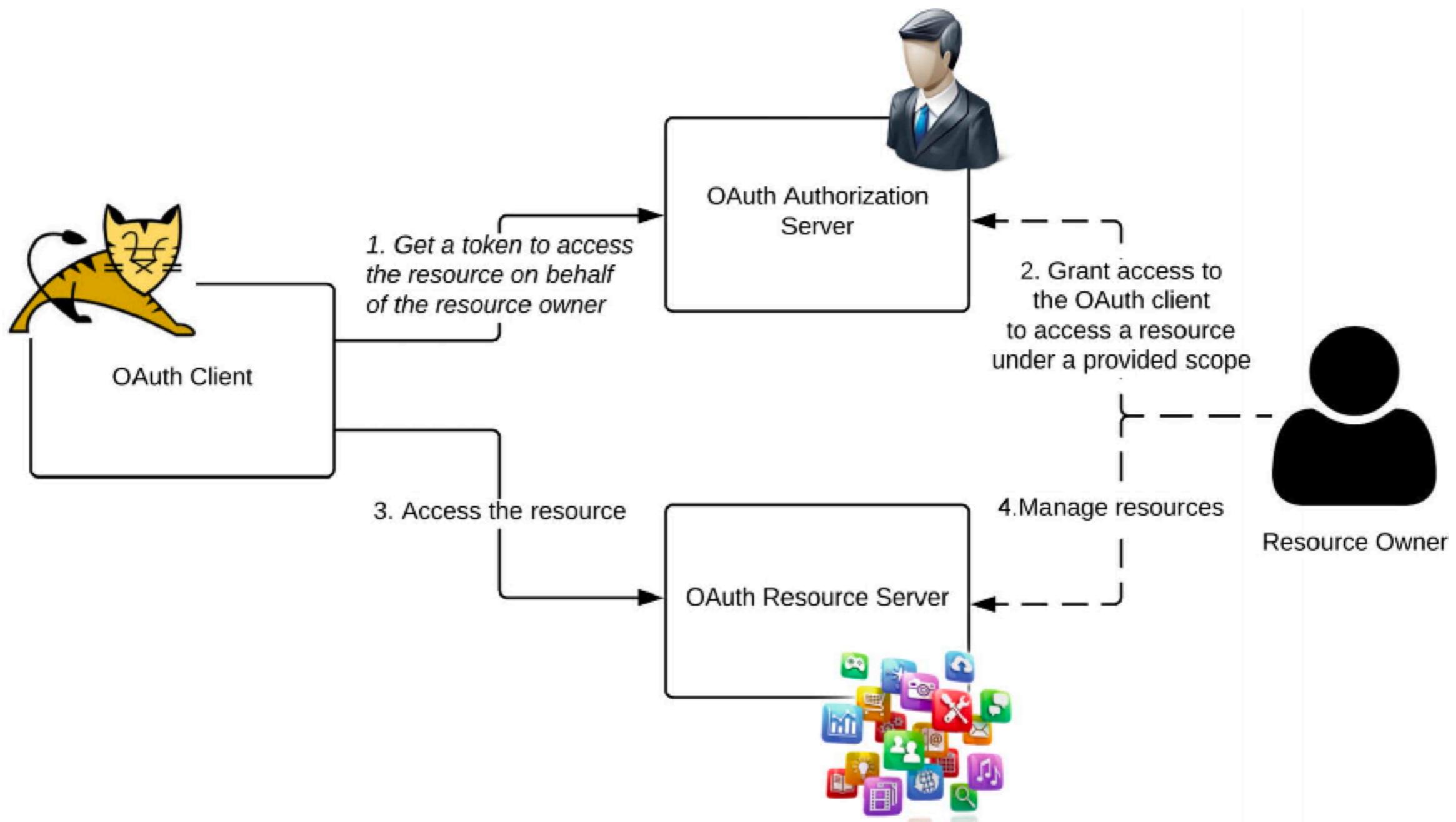
Passing user context as JWT



Problem ?

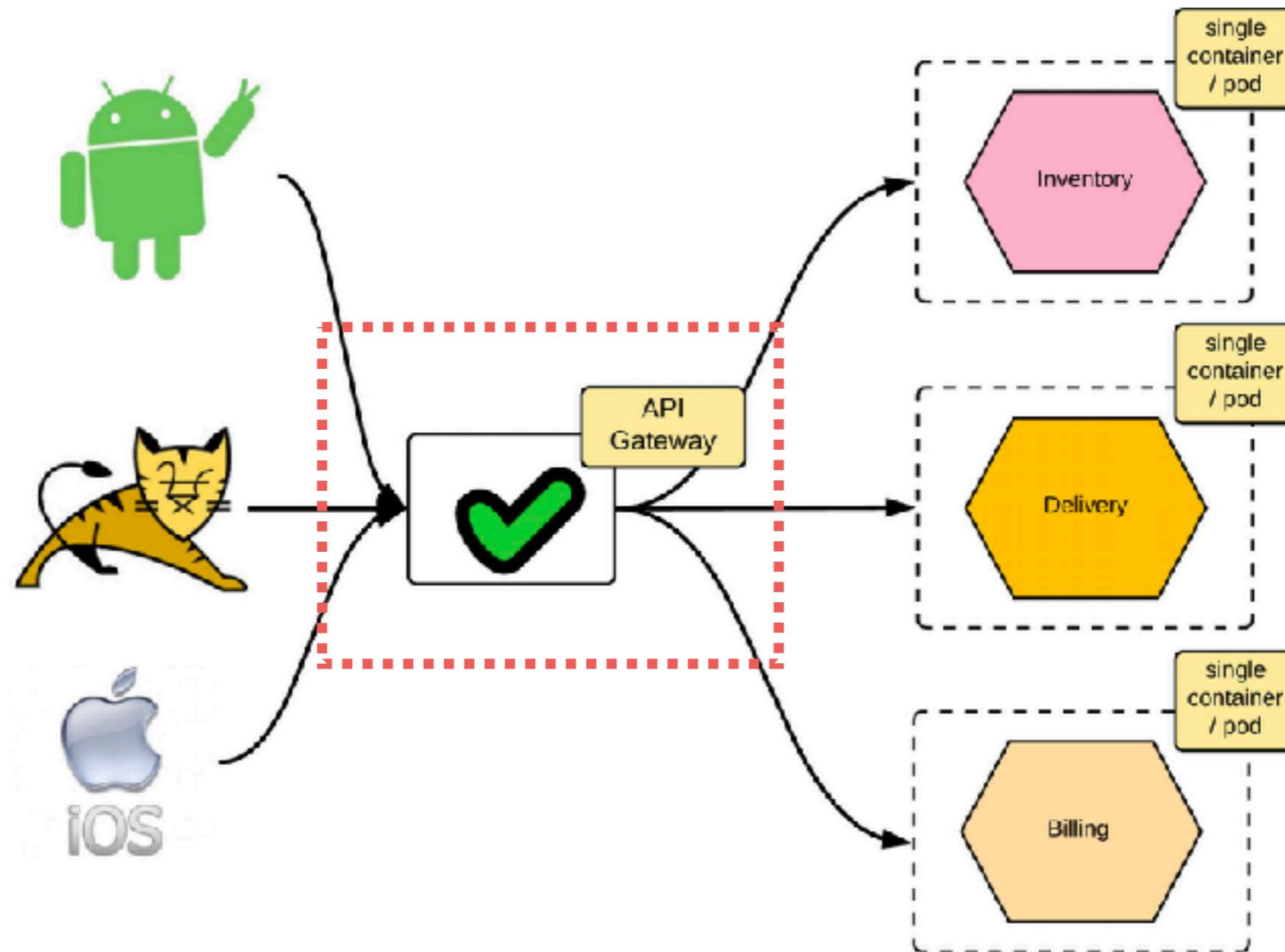


OAuth 2.0



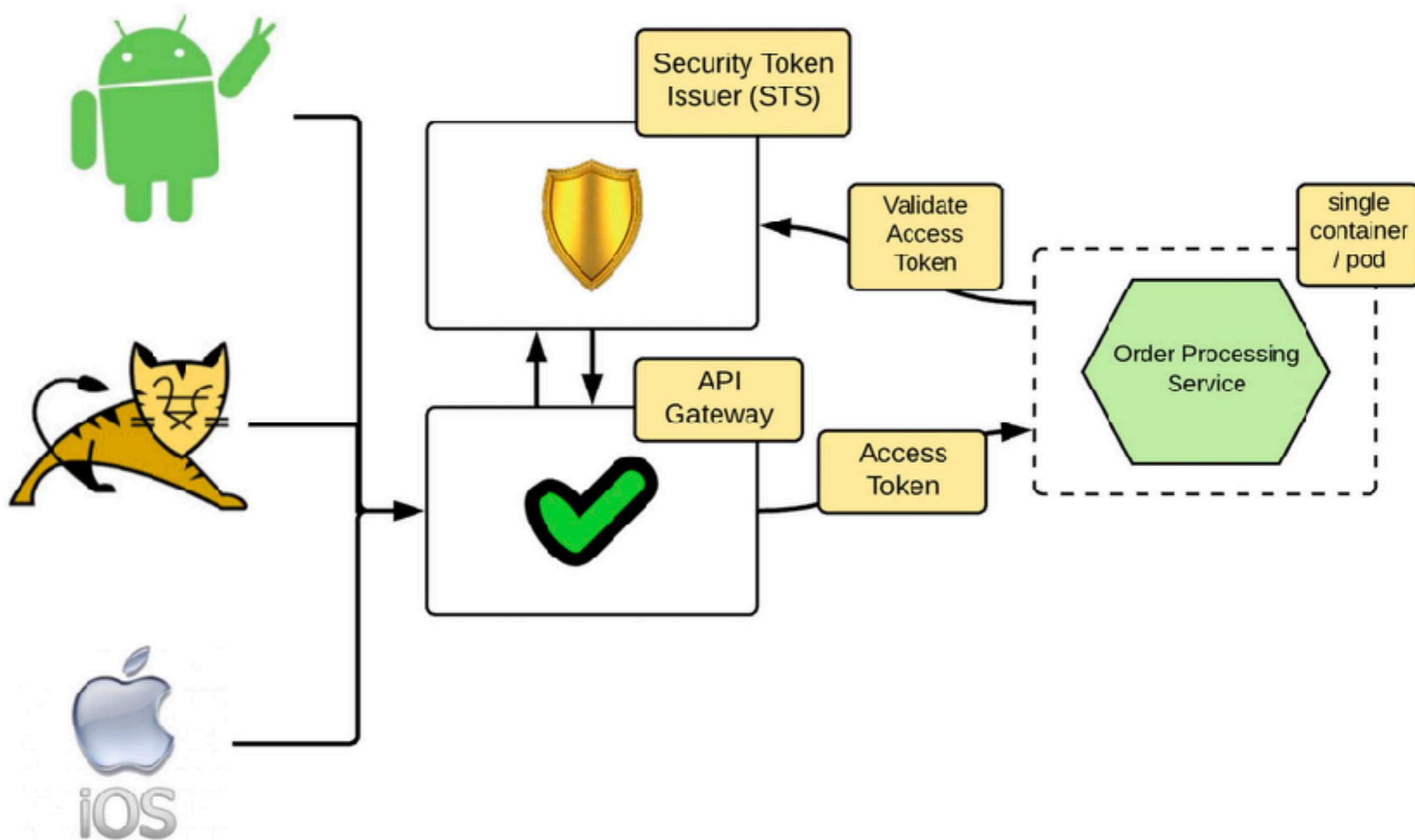
Centralize pattern

Using API gateway pattern

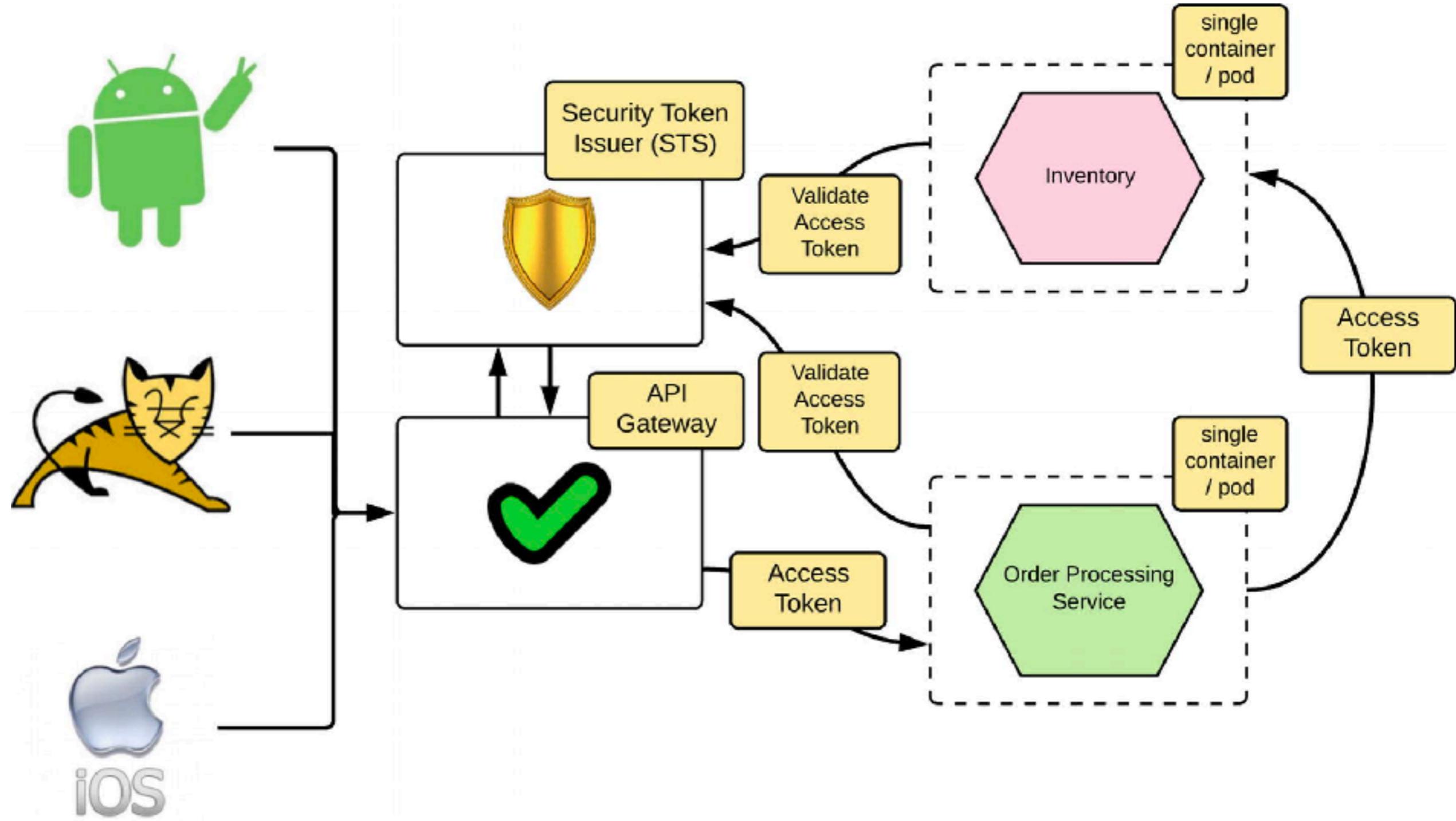


Centralize pattern

Using API gateway pattern

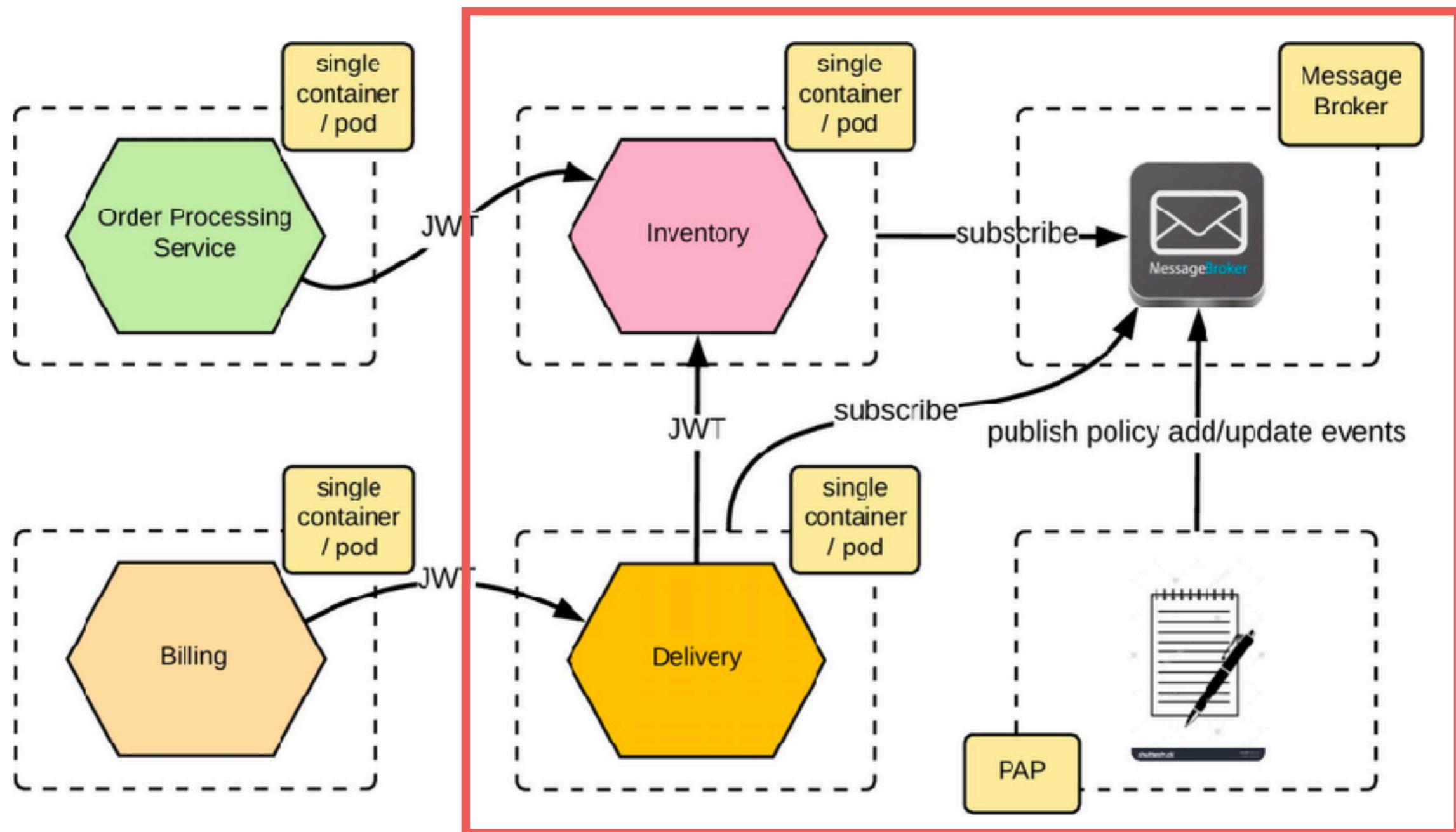


API gateway with OAuth 2.0



Access control of services

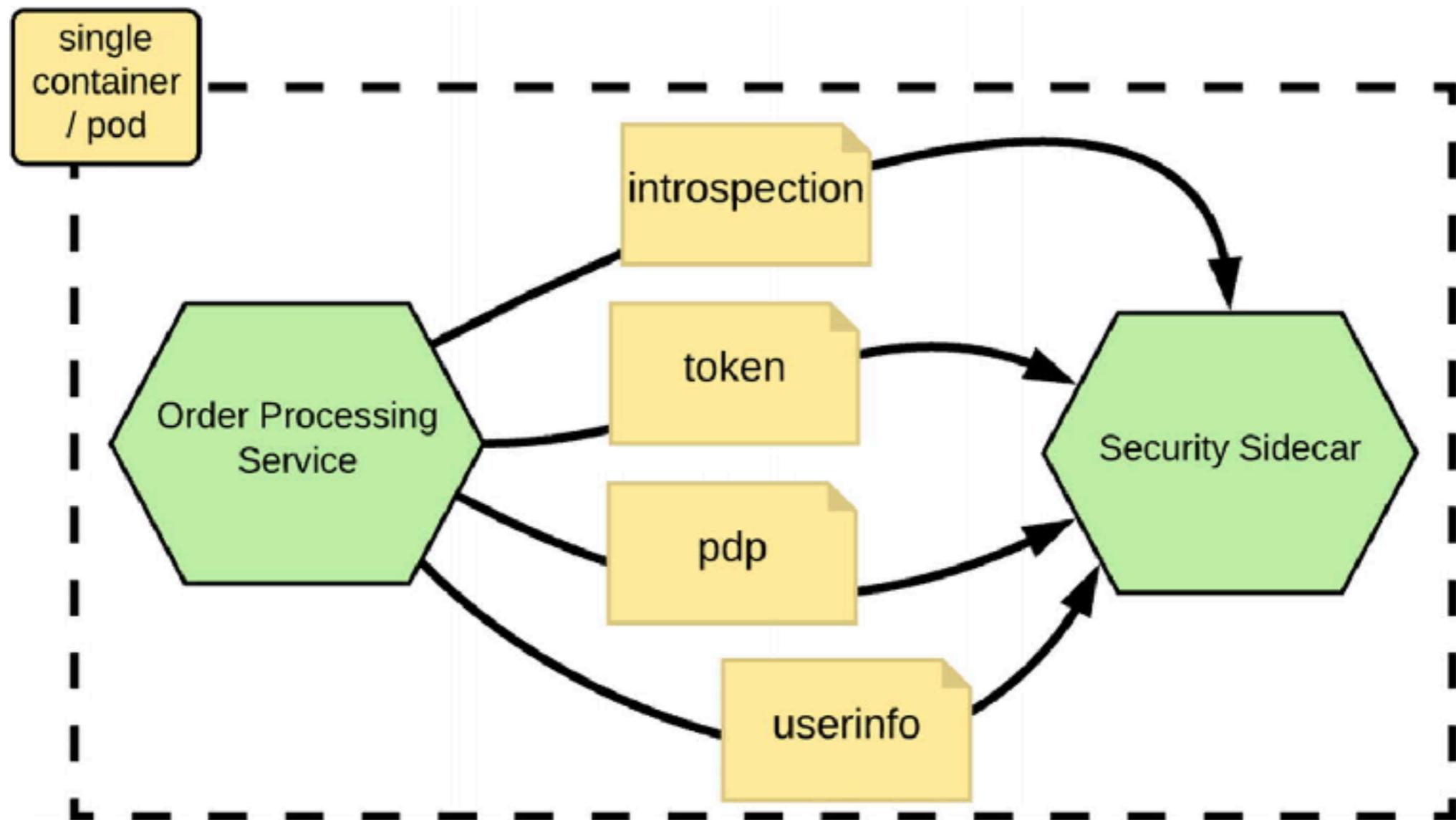
Policy Administration Point



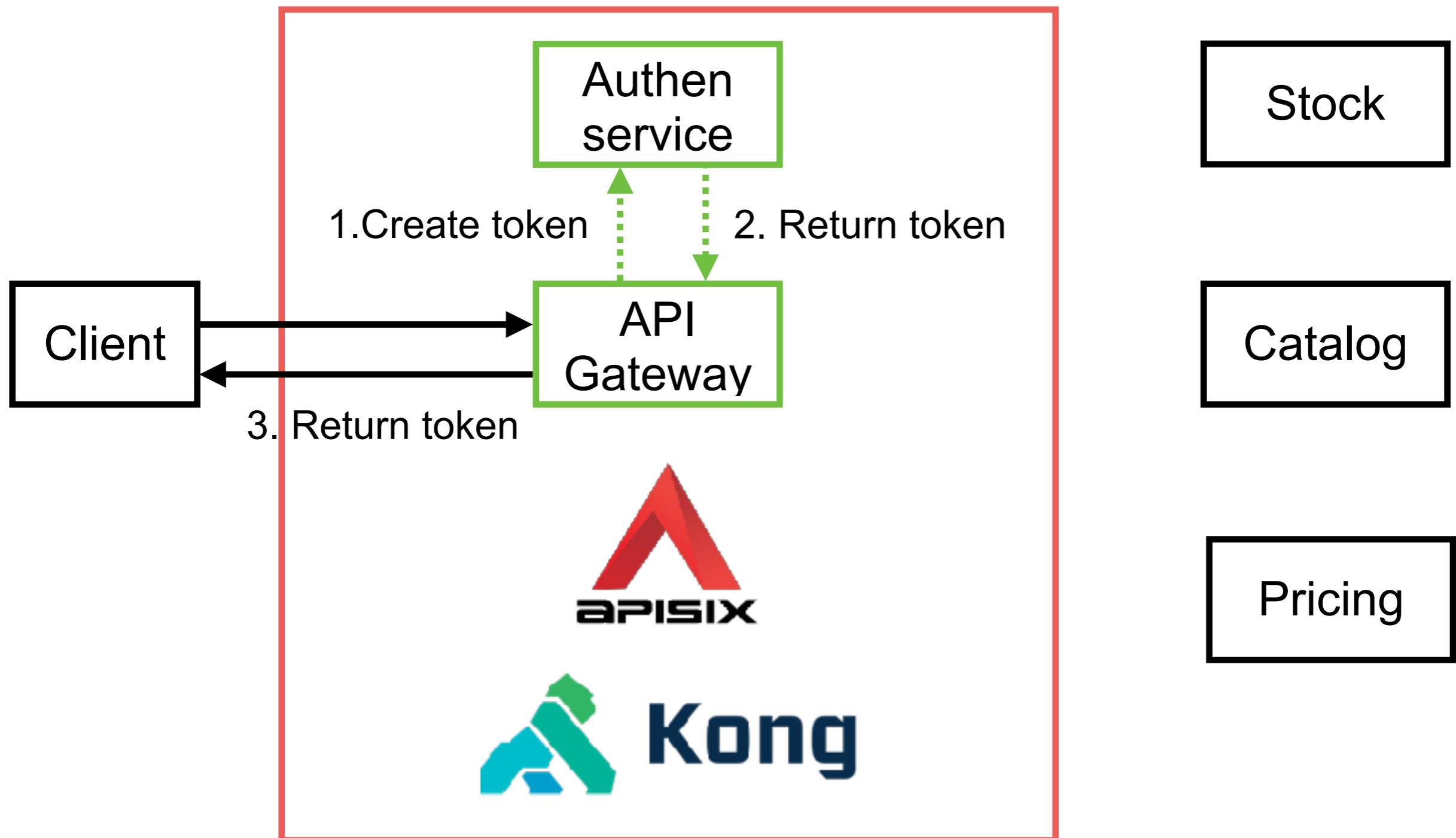
Working sidecar



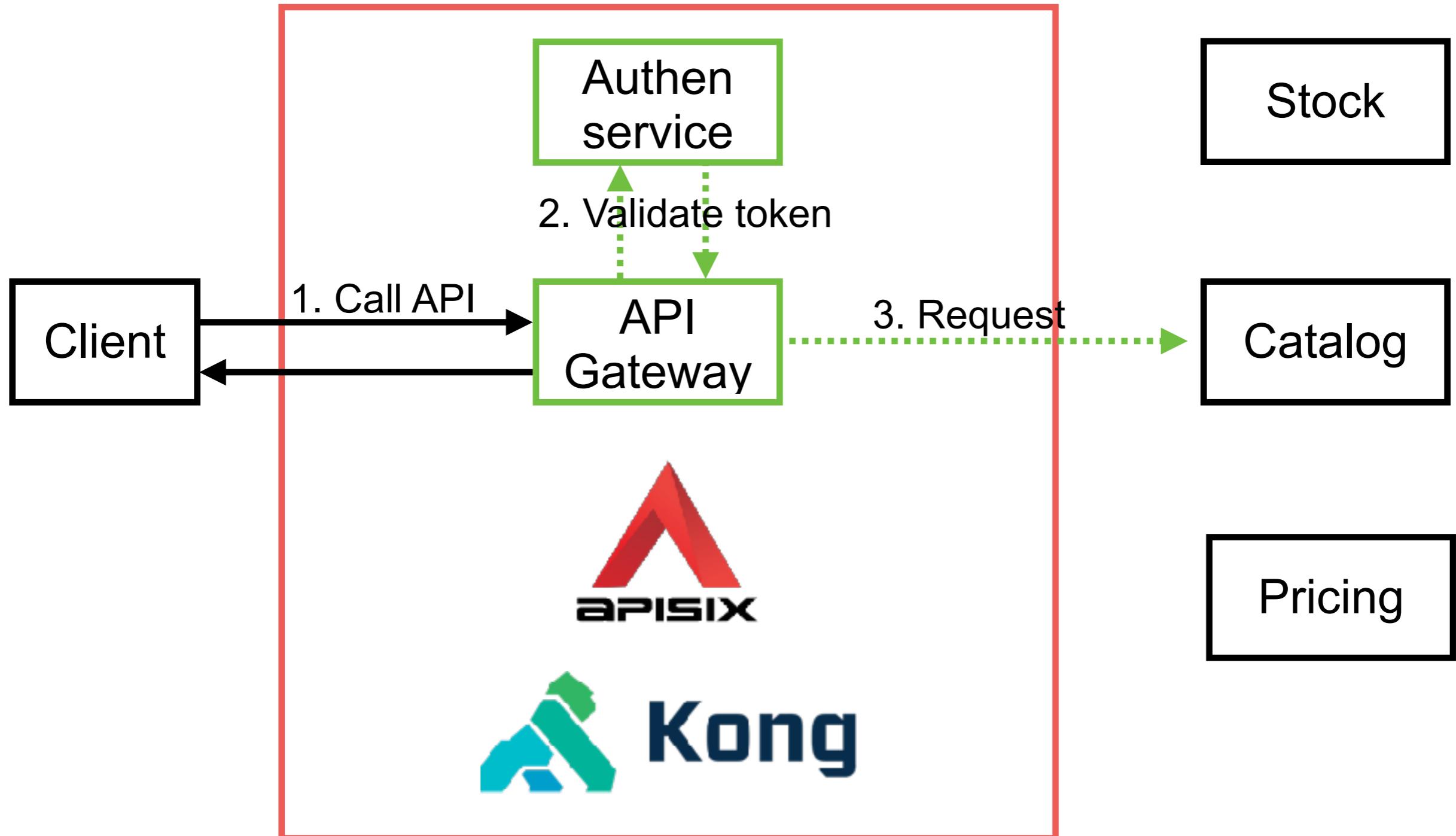
Security sidecar

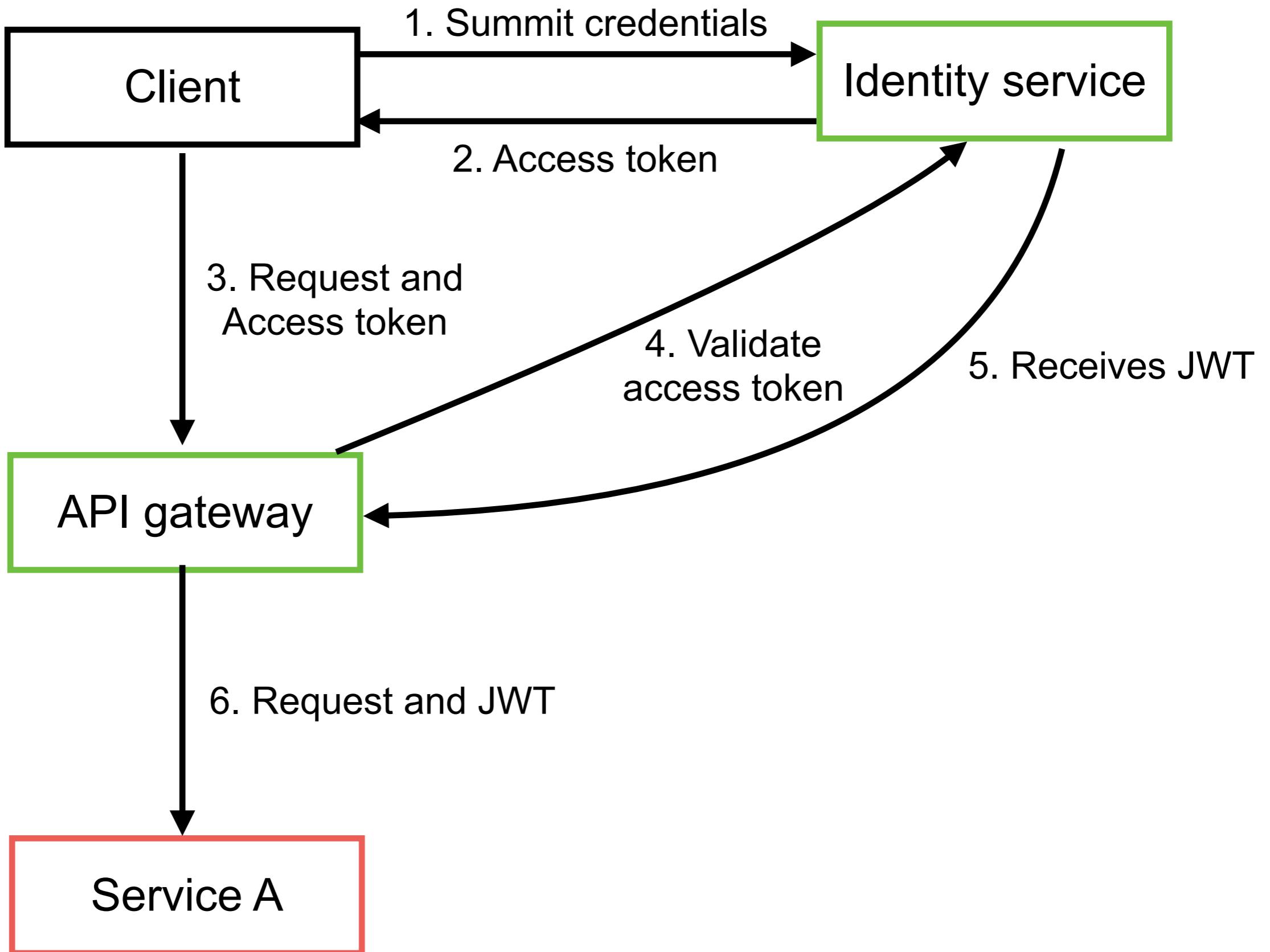


Workshop with Secure service



Workshop with Secure service

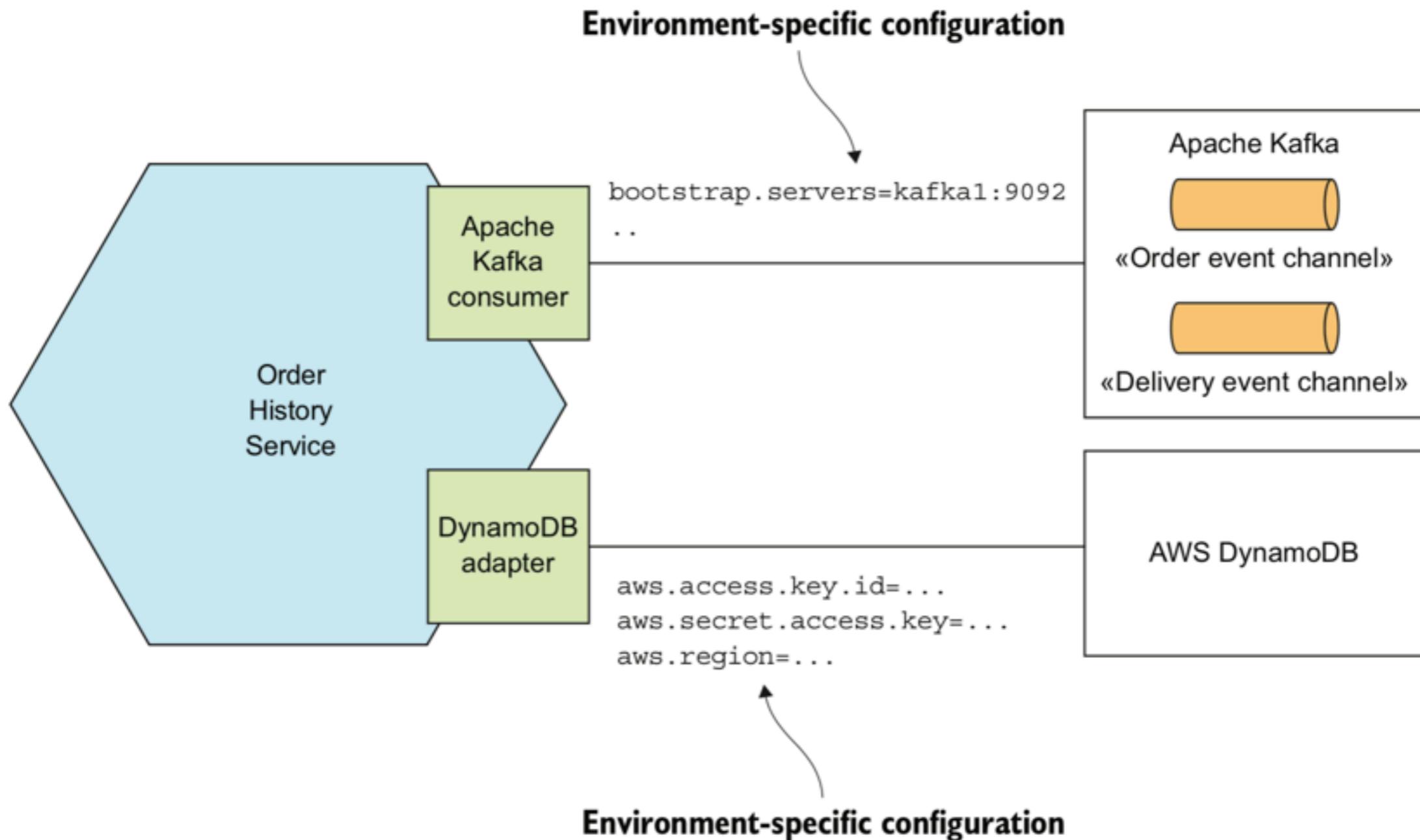




Configurable services



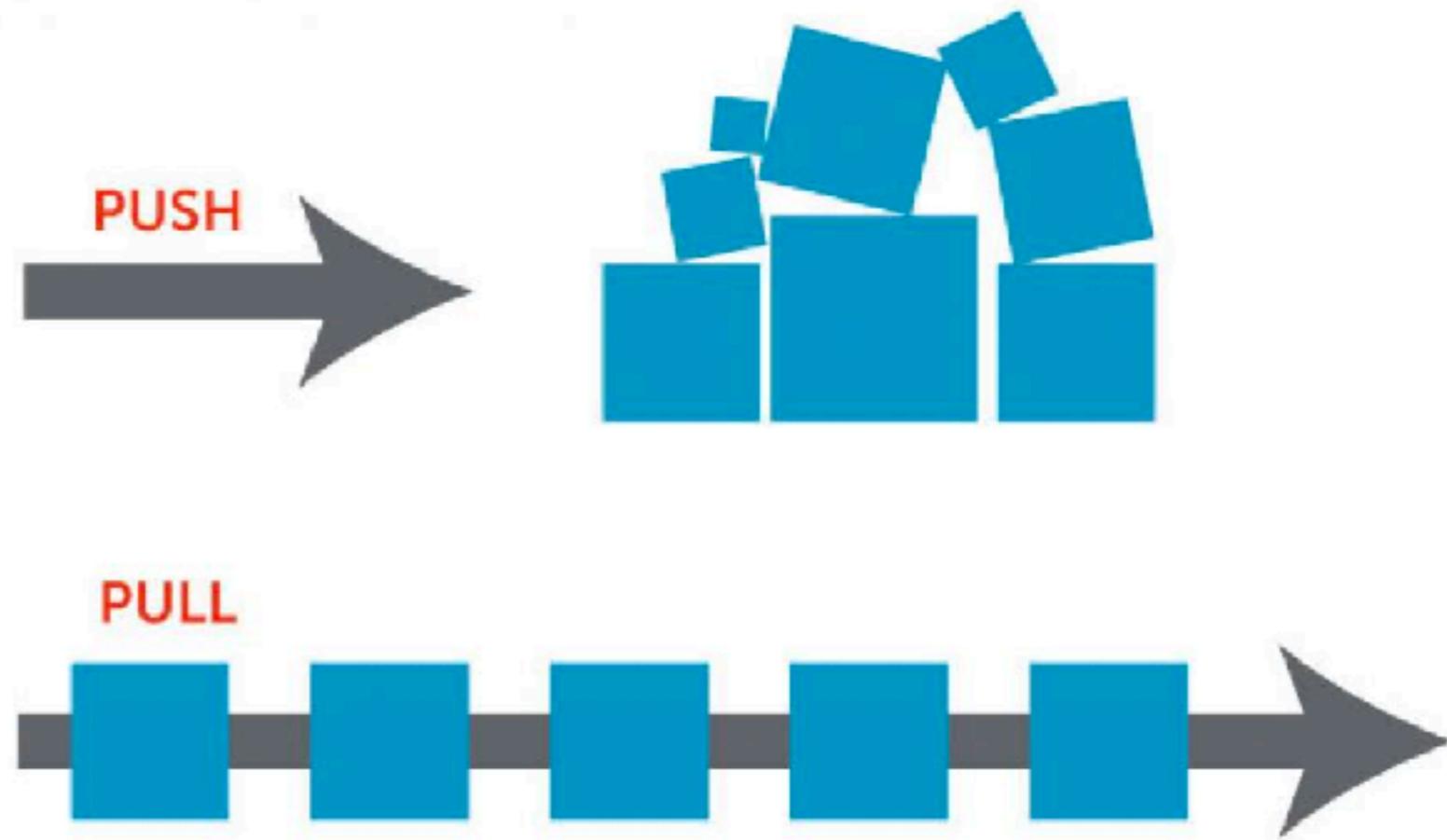
Design configurable services



External configuration models

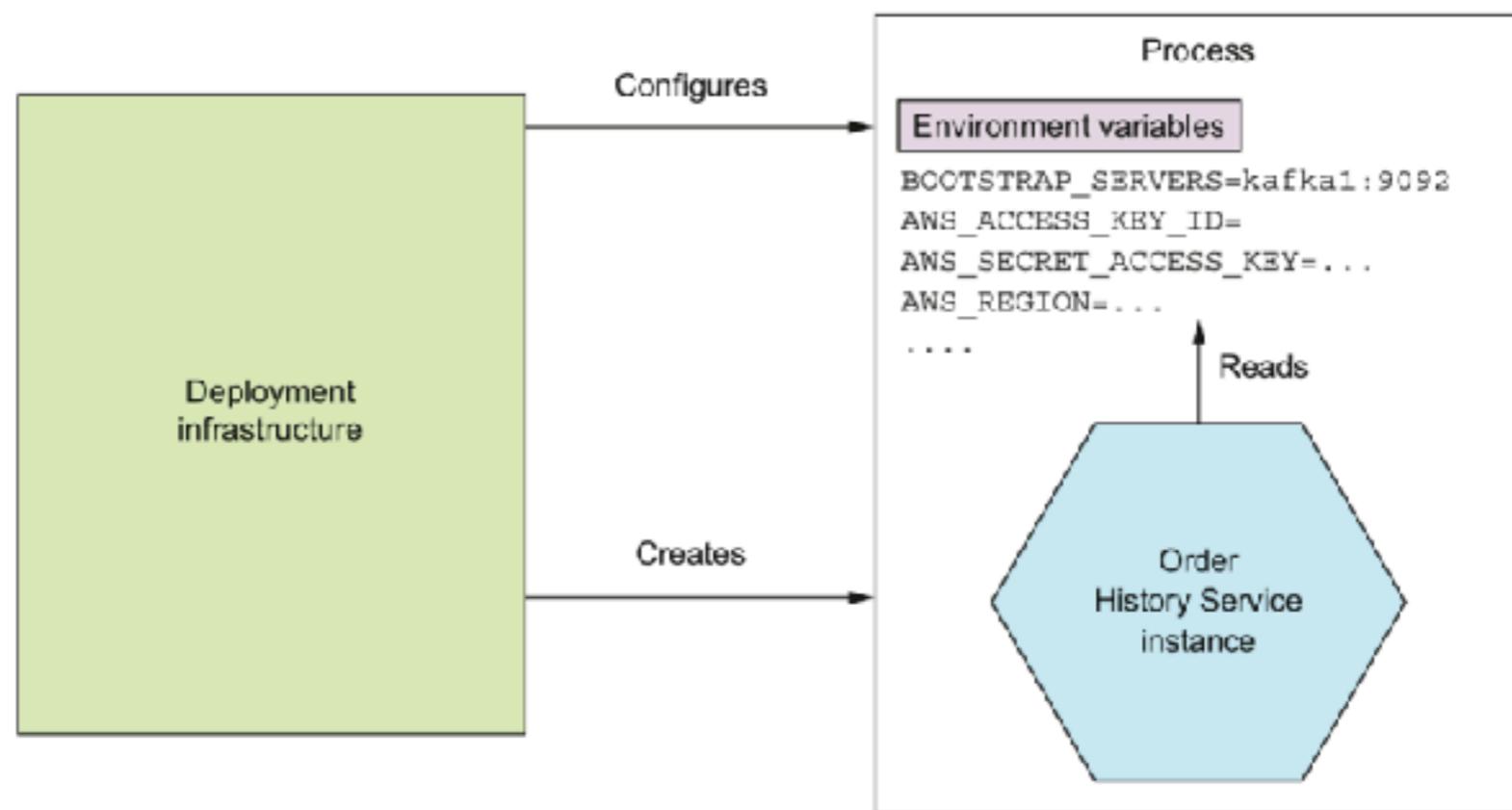
Push model

Pull model



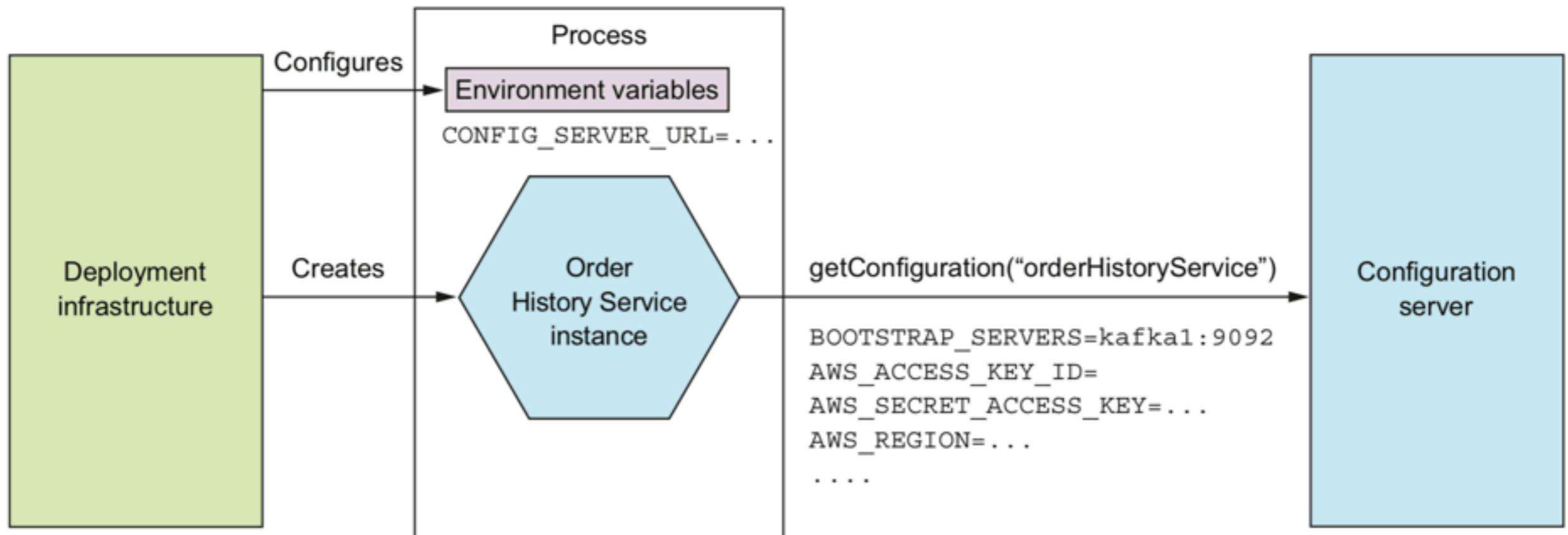
Push model

Pass the configuration to service
OS environment variables
Configuration files



Pull model

Service read configuration from configuration server



Benefits of configuration server

Centralized configuration

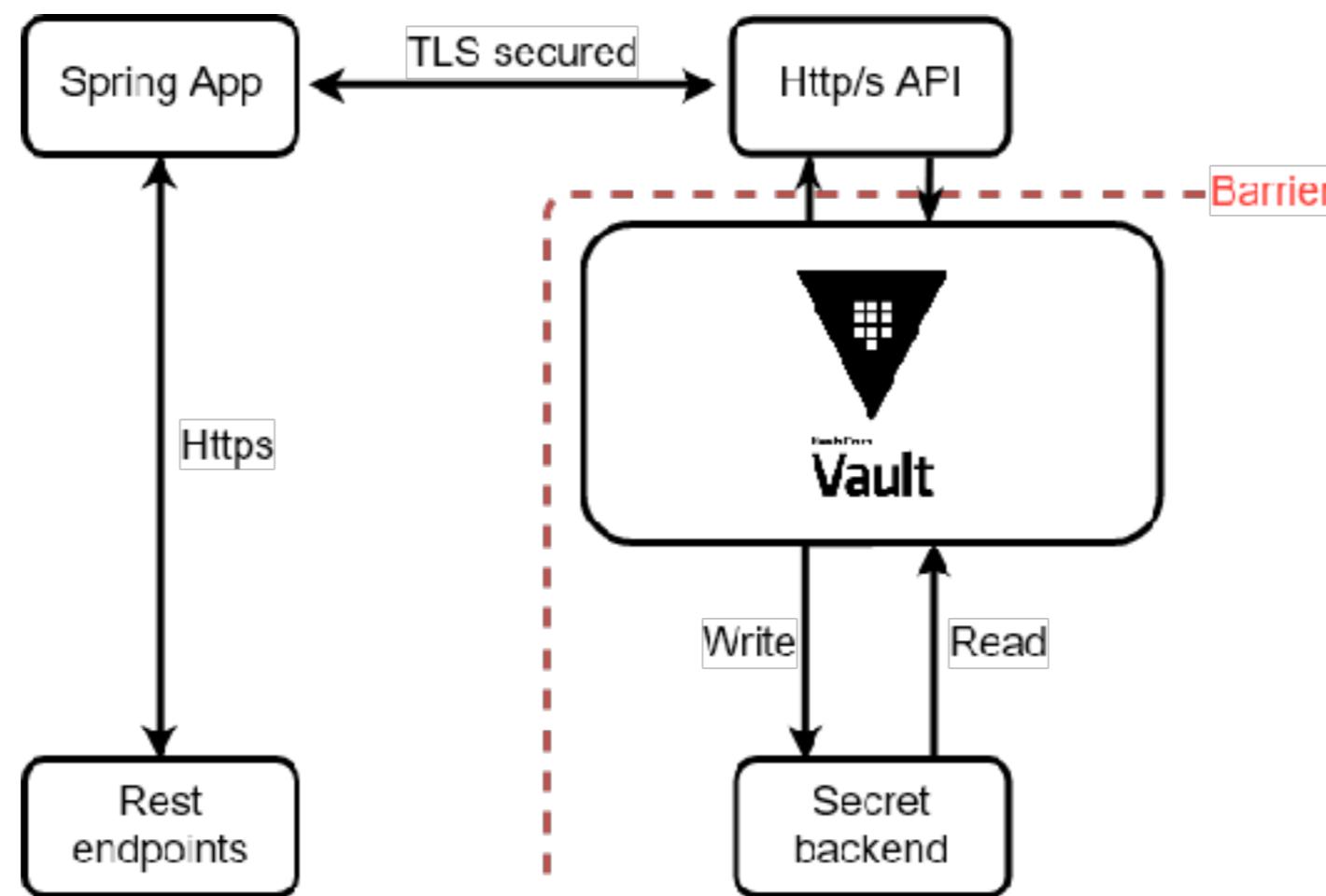
Transparent decryption of sensitive data

Dynamic reconfiguration



Workshop with Configuration

Plain text
Secure data



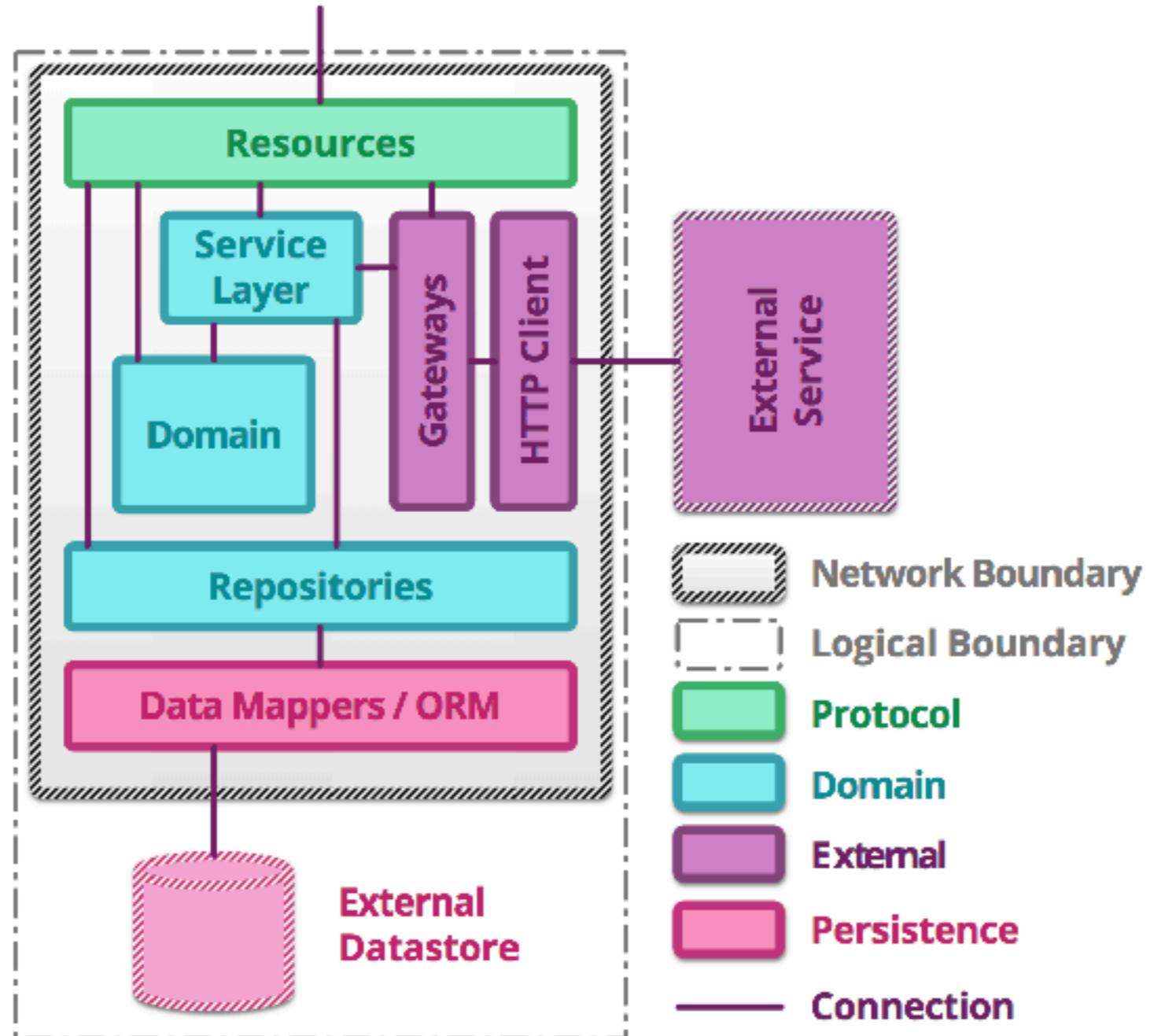
<https://www.vaultproject.io/>



Microservice Testing ?



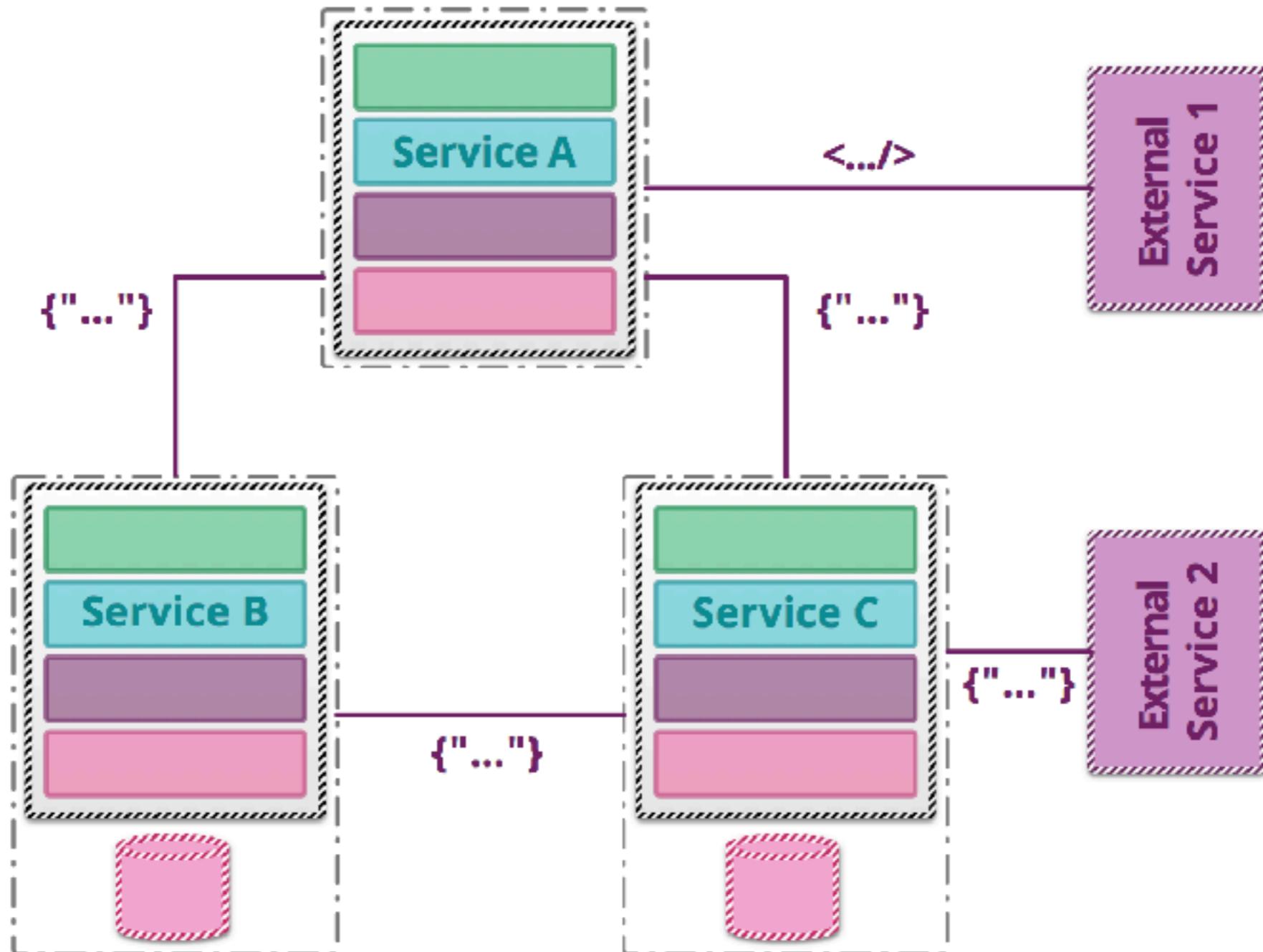
Service structure



<https://martinfowler.com/articles/microservice-testing>

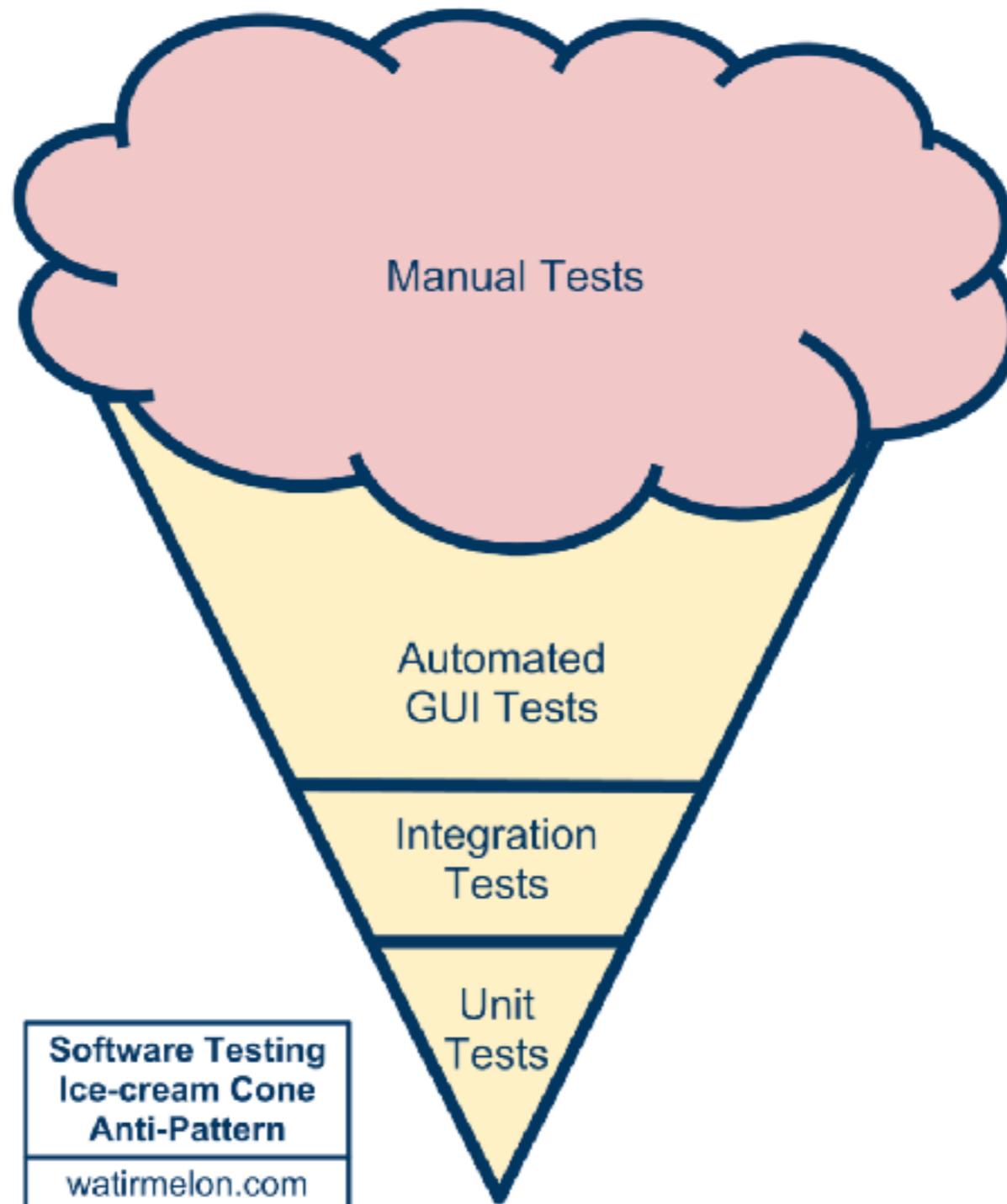


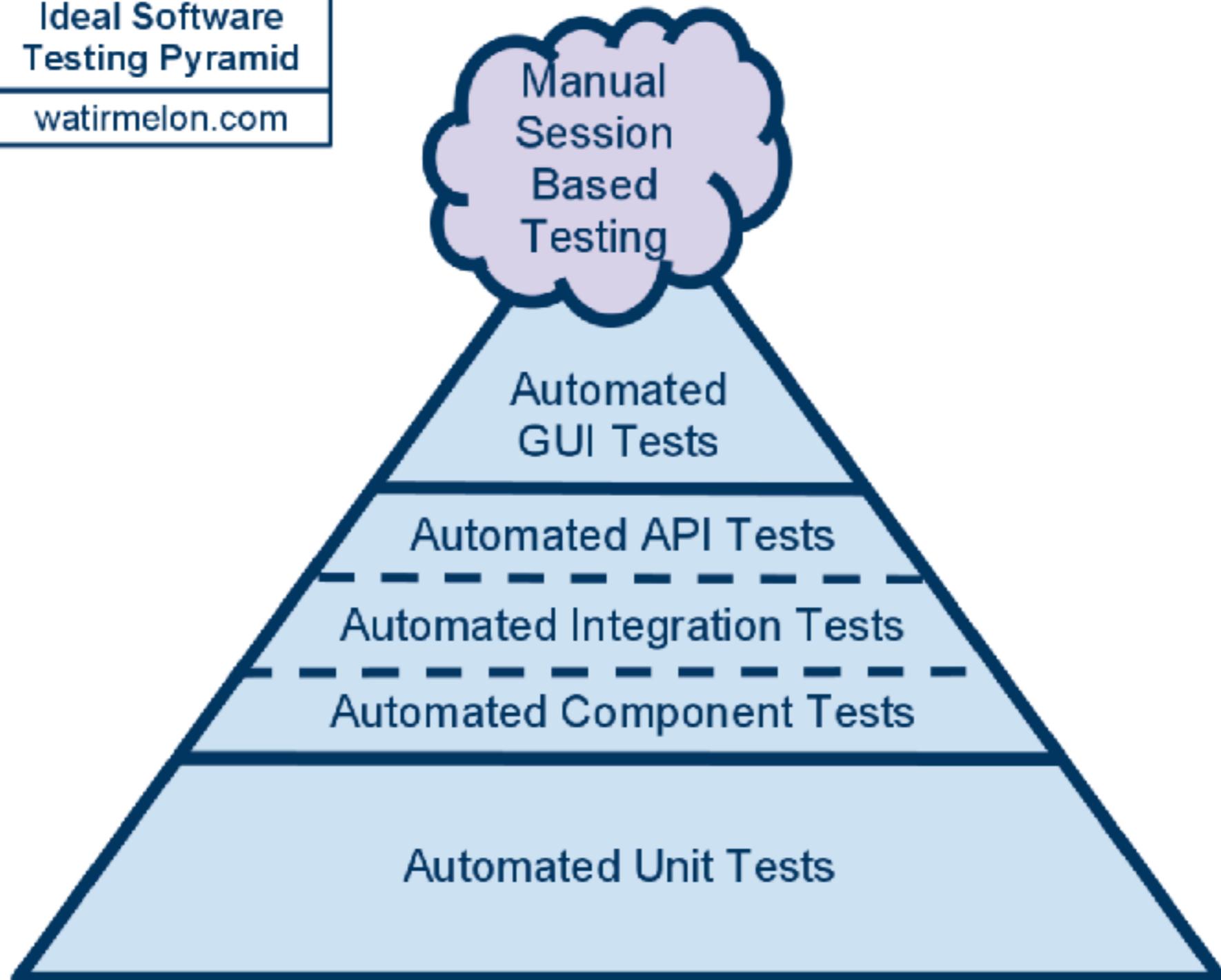
Multiple services

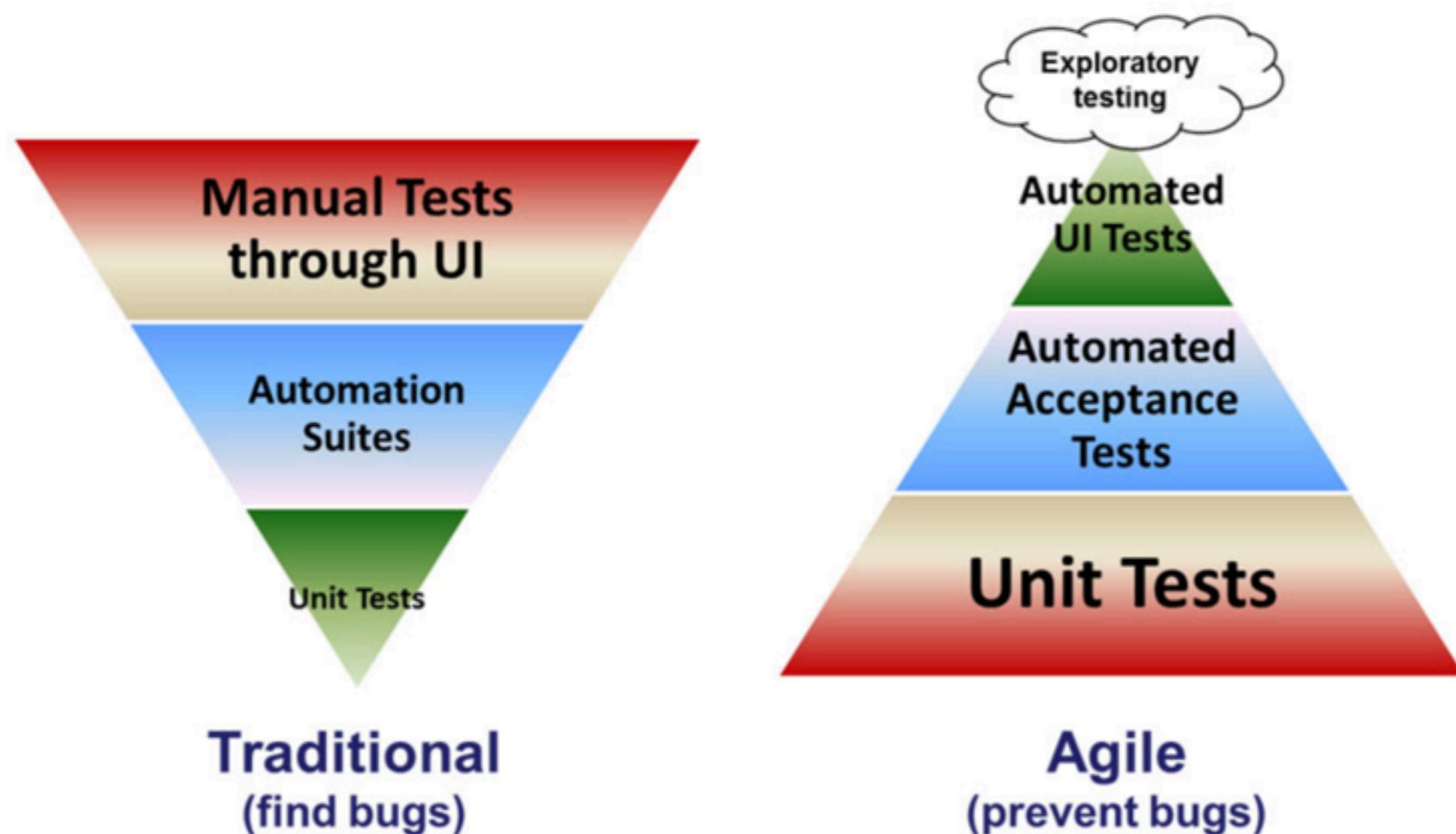


<https://martinfowler.com/articles/microservice-testing>

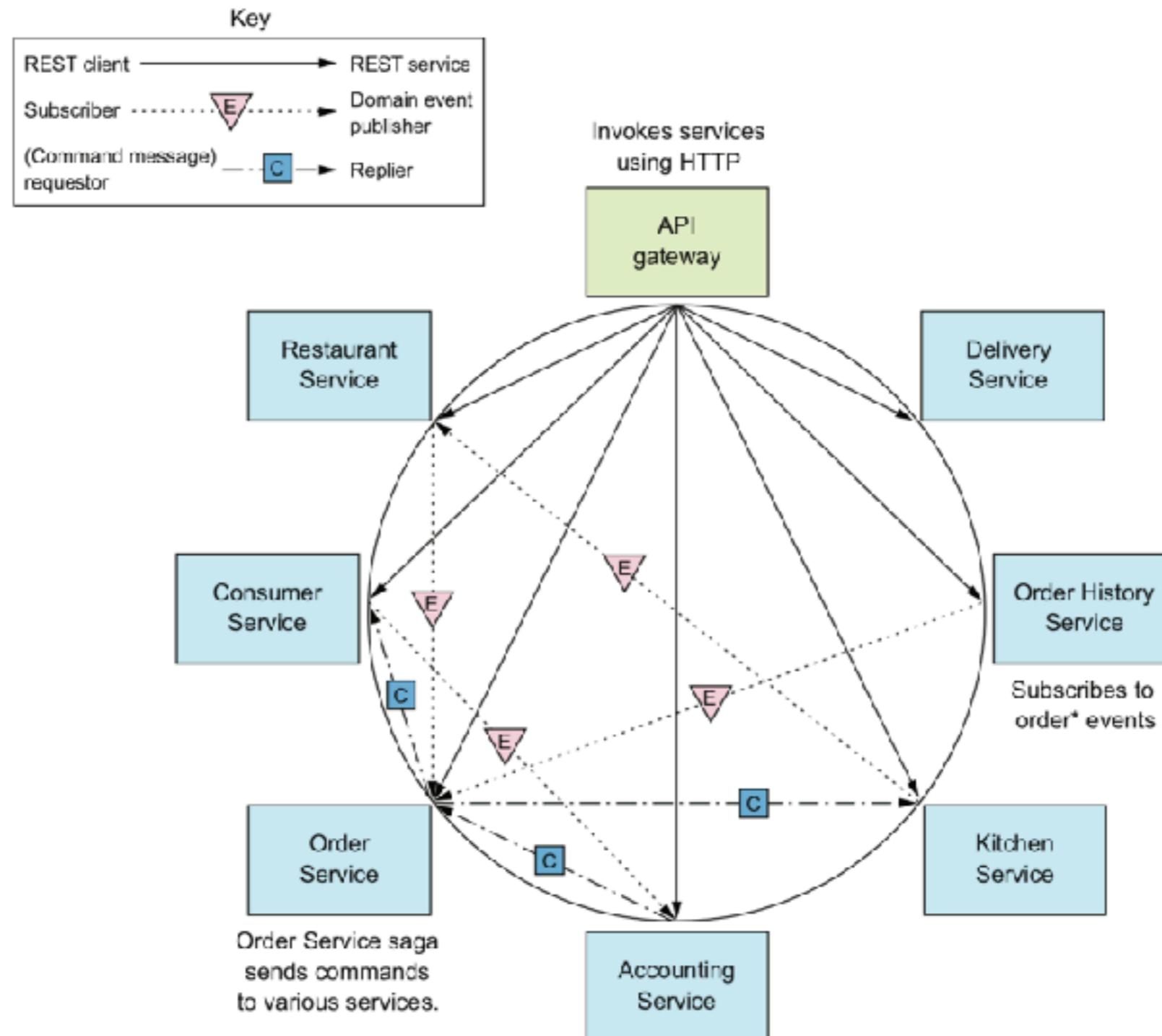




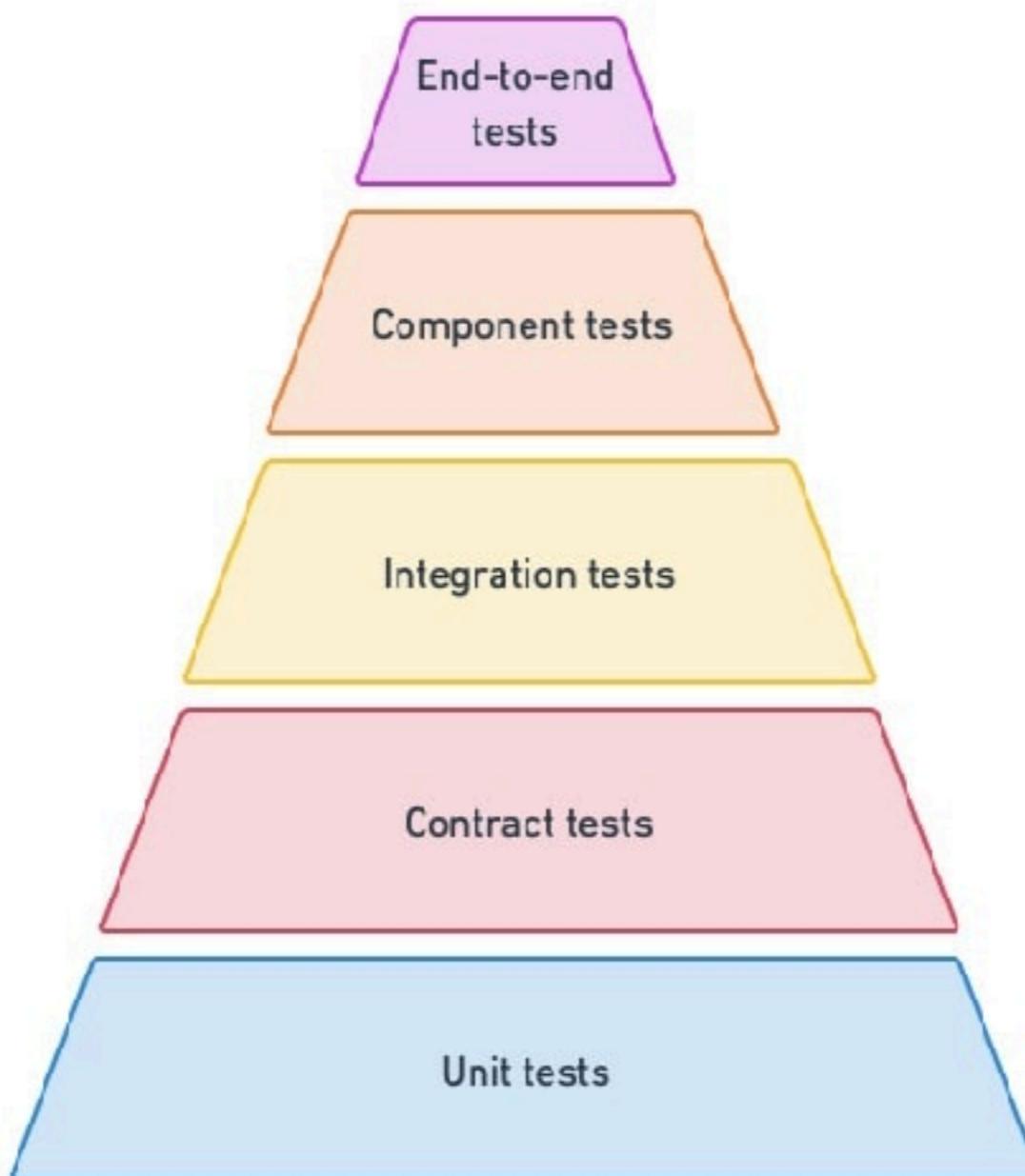




Testing in Microservice ?



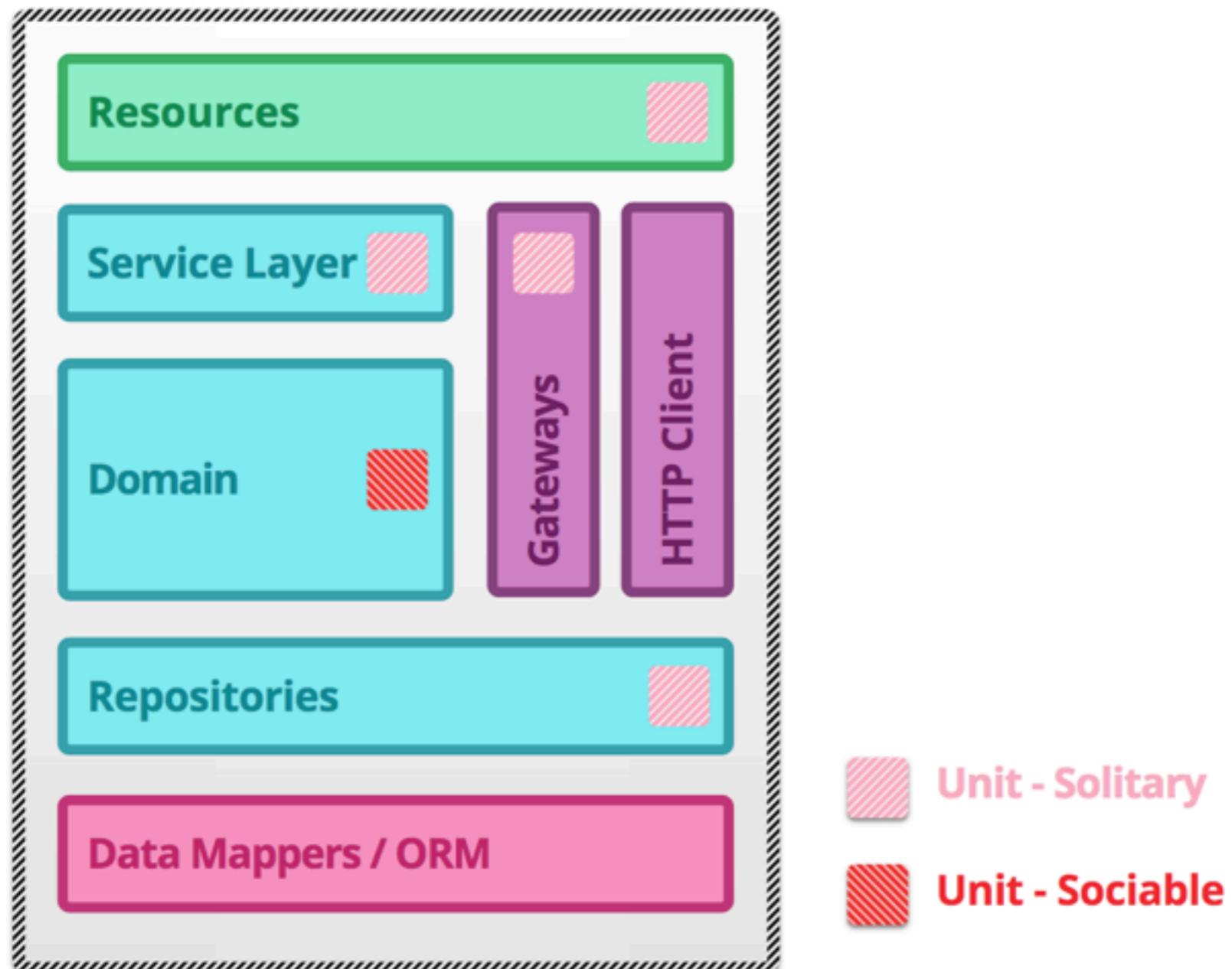
Microservice Testing



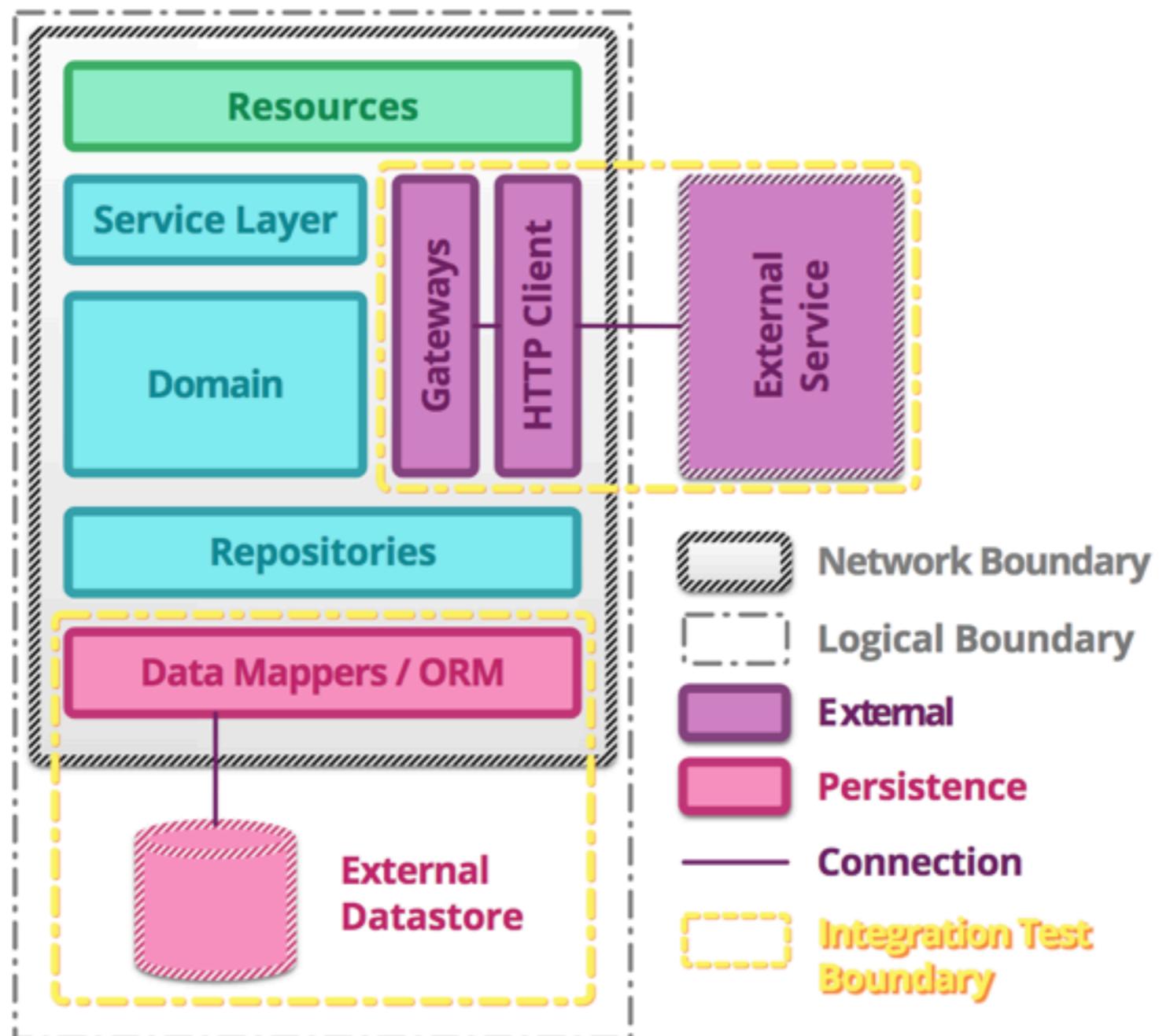
<https://semaphoreci.com/blog/test-microservices>



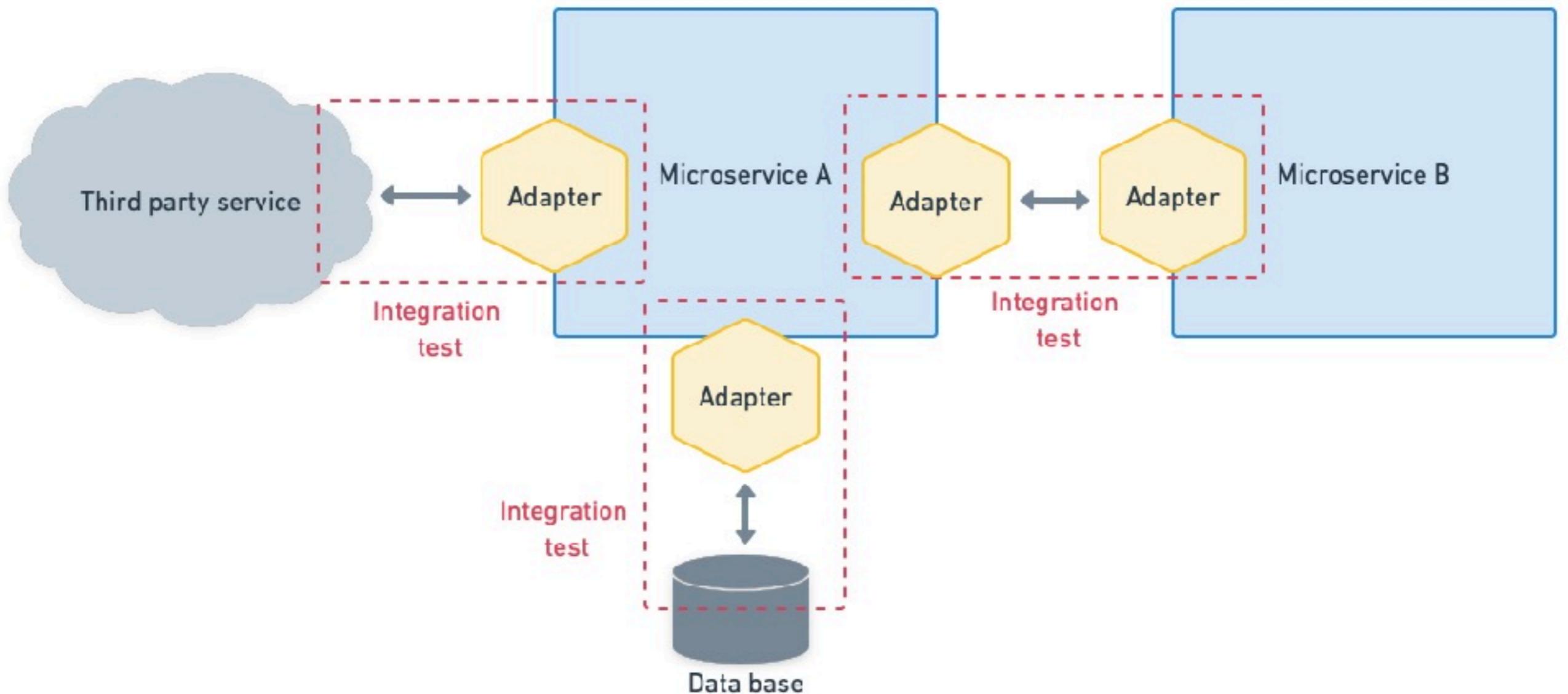
Unit testing



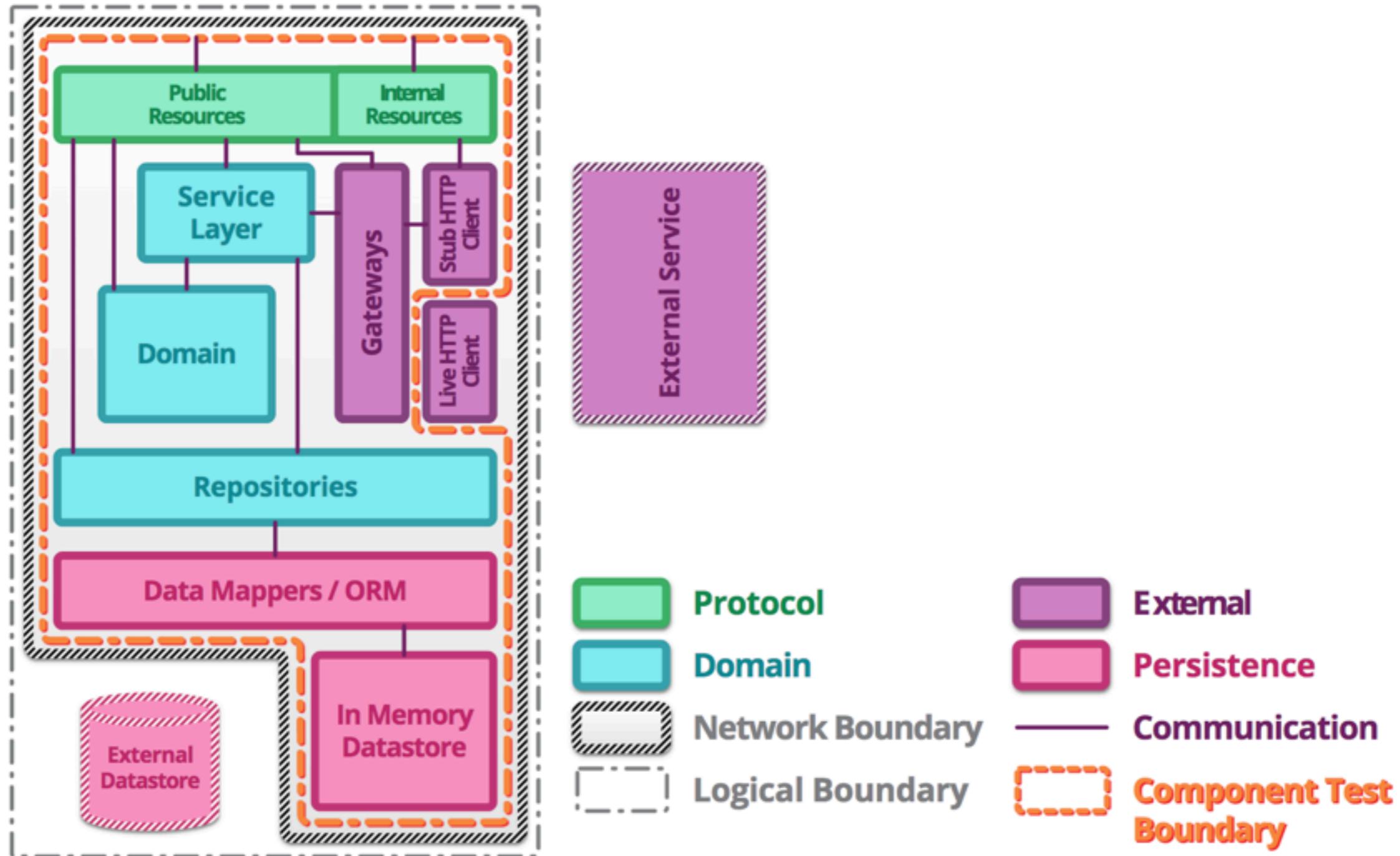
Integration testing



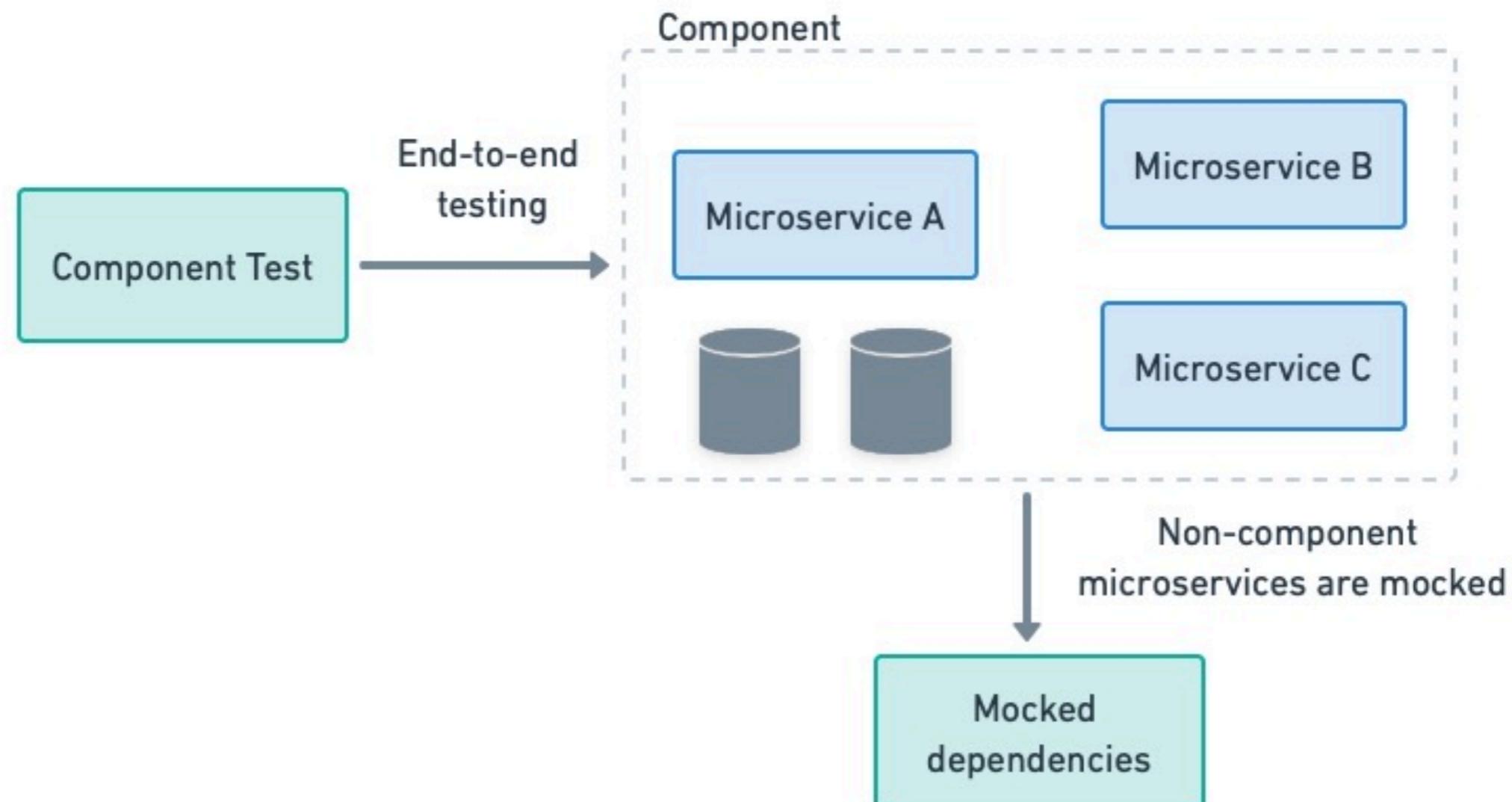
Integration testing



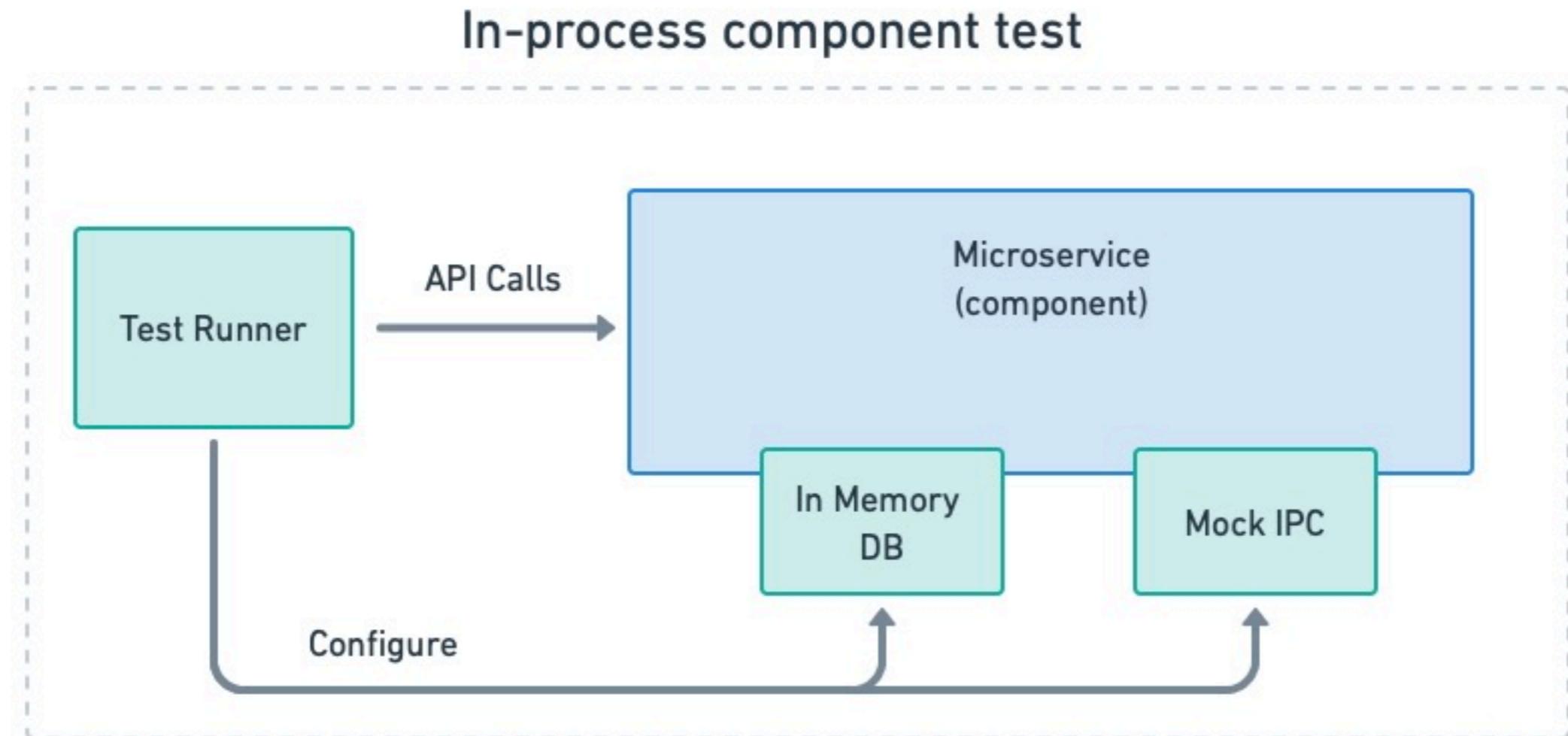
Component testing



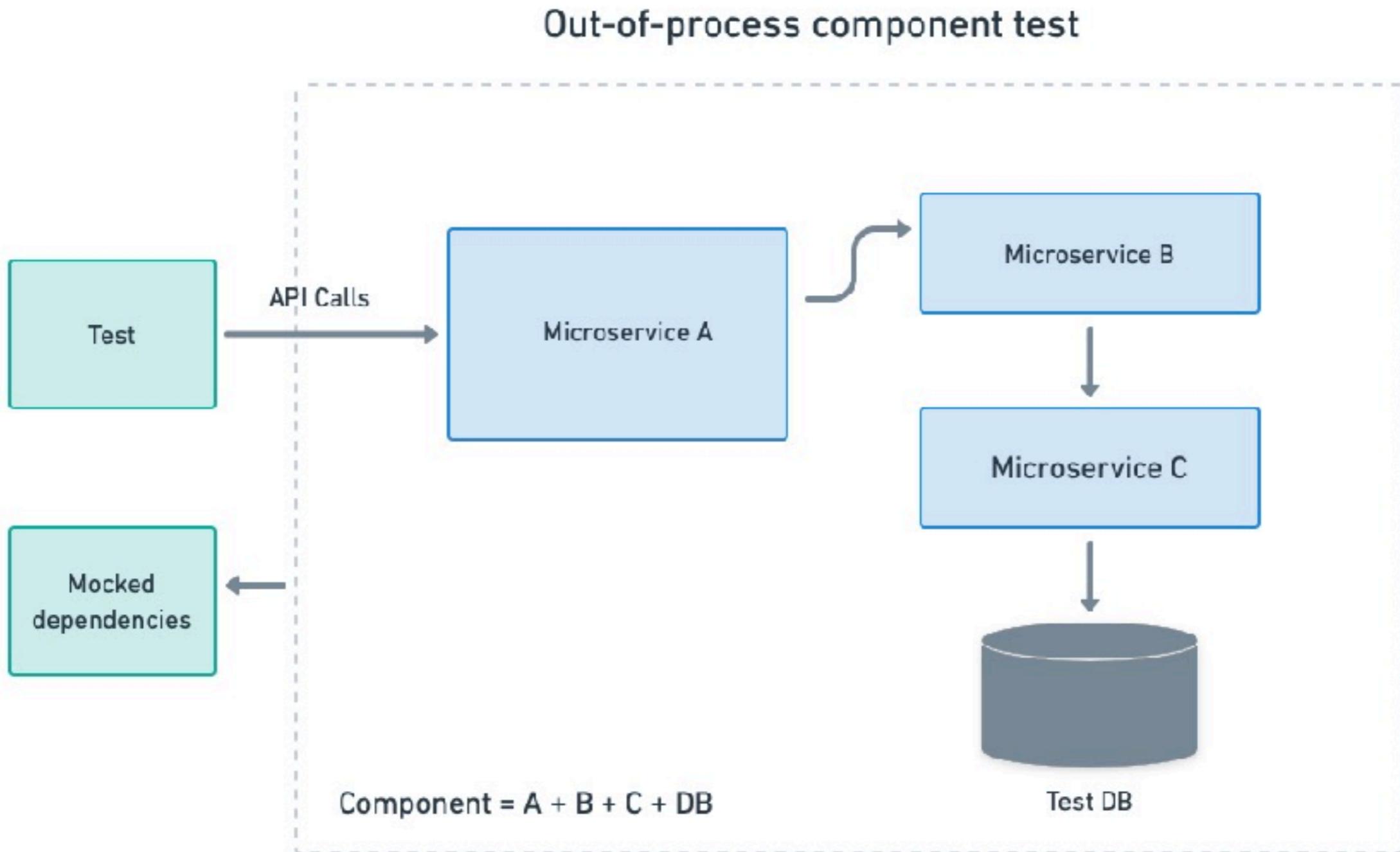
Component testing



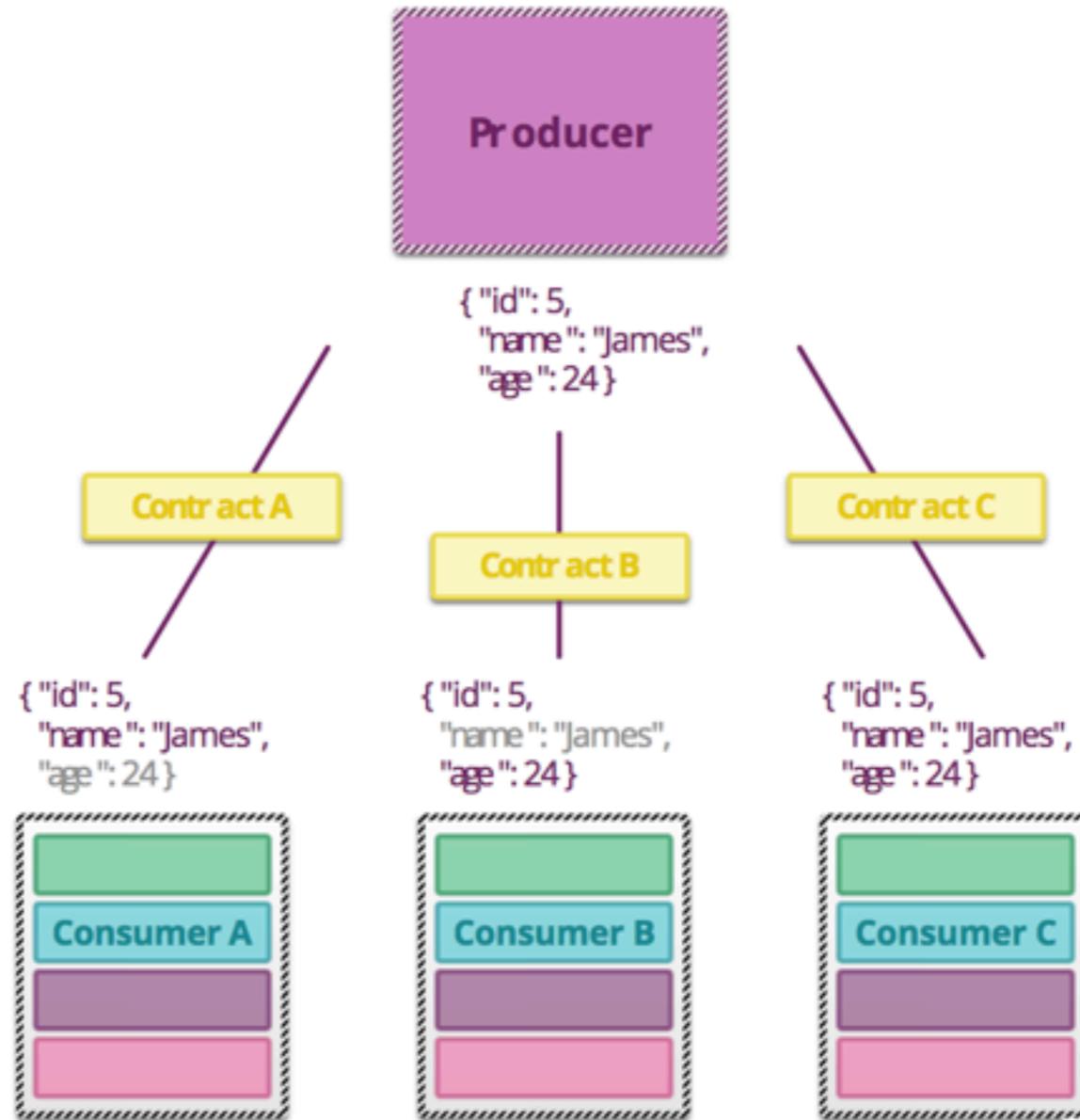
Component testing (in-process)



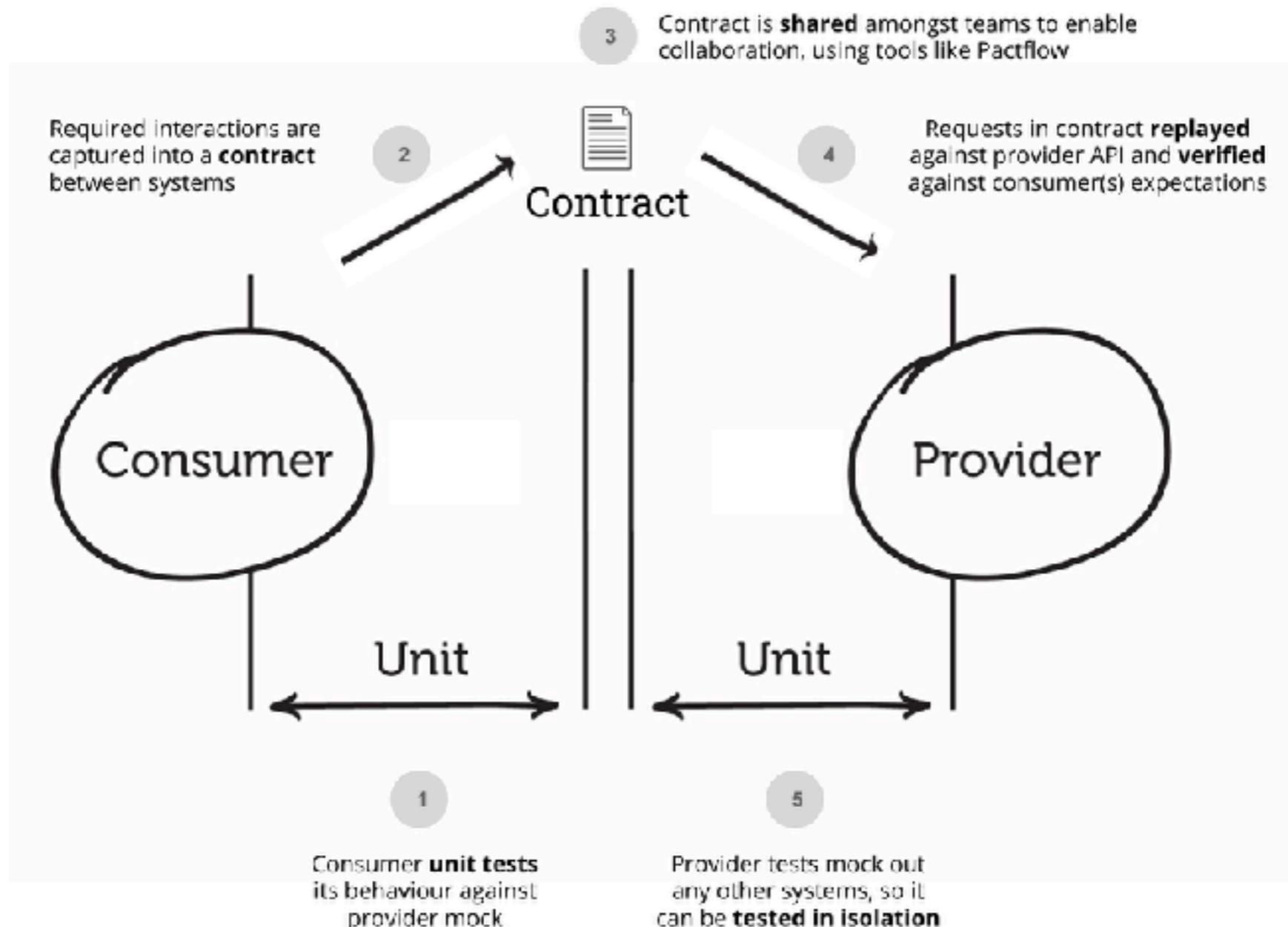
Component testing (out-process)



Contract testing



Contract testing with Pact



<https://docs.pact.io/>



Pact broker

Store and share all contracts and verification results

Consumer ↓↑	Provider ↓↑	Latest pact published	Last verified
Foo	Animals	2 minutes ago	2 days ago
Foo	Bar	7 days ago	15 days ago ▲
Foo	Hello World App	1 day ago	
Foo	Wiffles	less than a minute ago	7 days ago
Some other app	A service	26 days ago	less than a minute ago
The Android App	The back end	less than a minute ago	

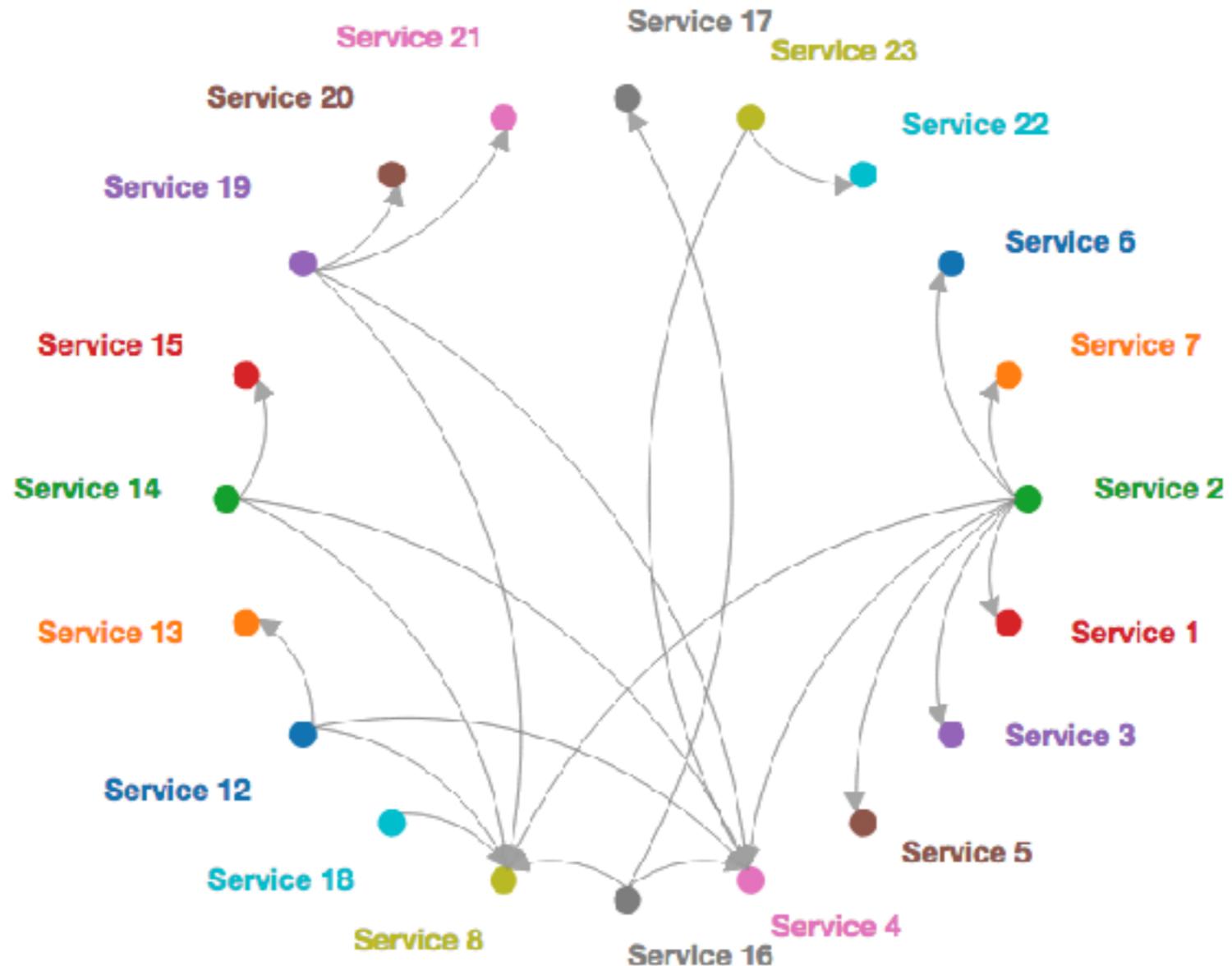
https://docs.pact.io/pact_broker



Microservices

© 2022 - 2023 Siam Chamnankit Company Limited. All rights reserved.

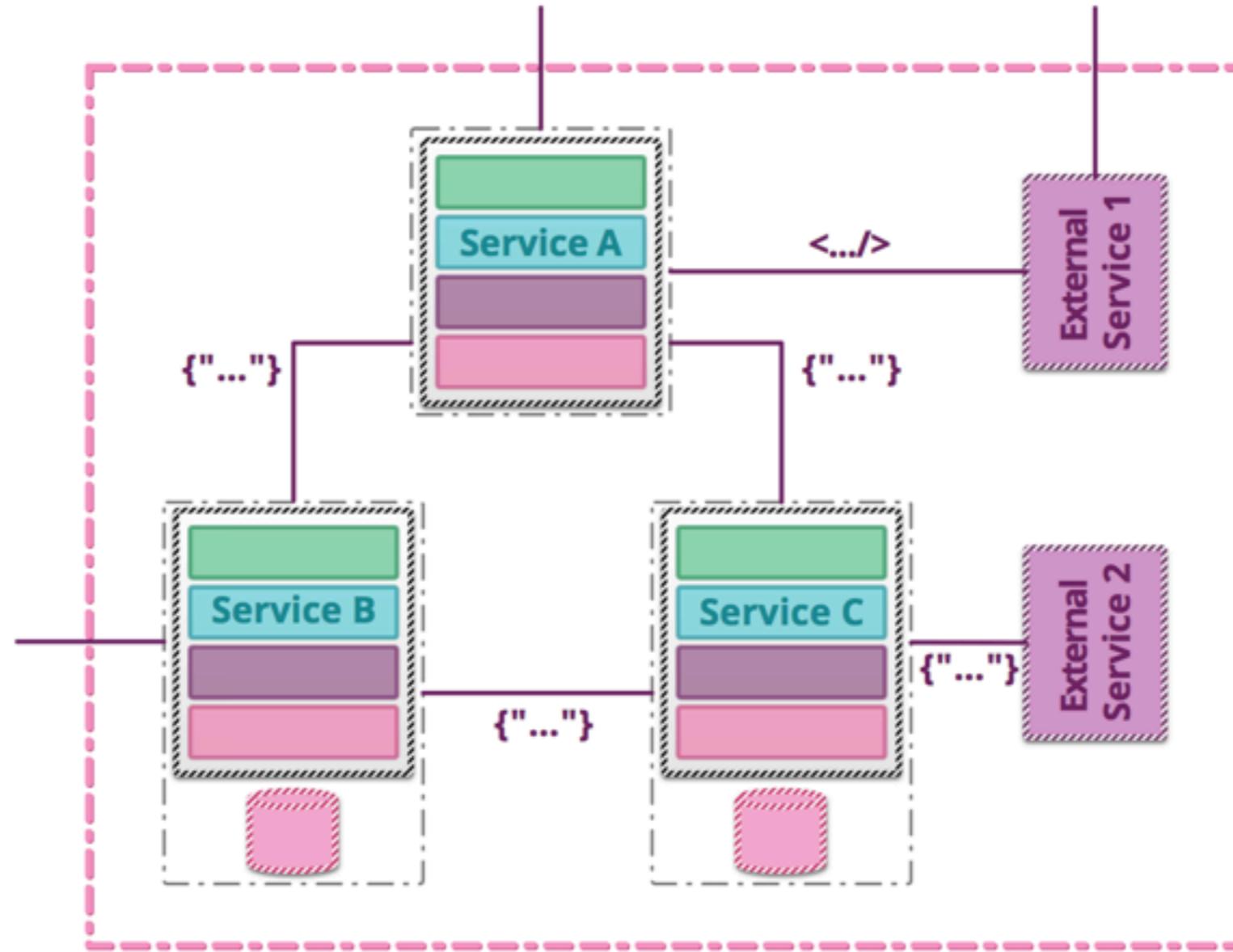
Pact broker



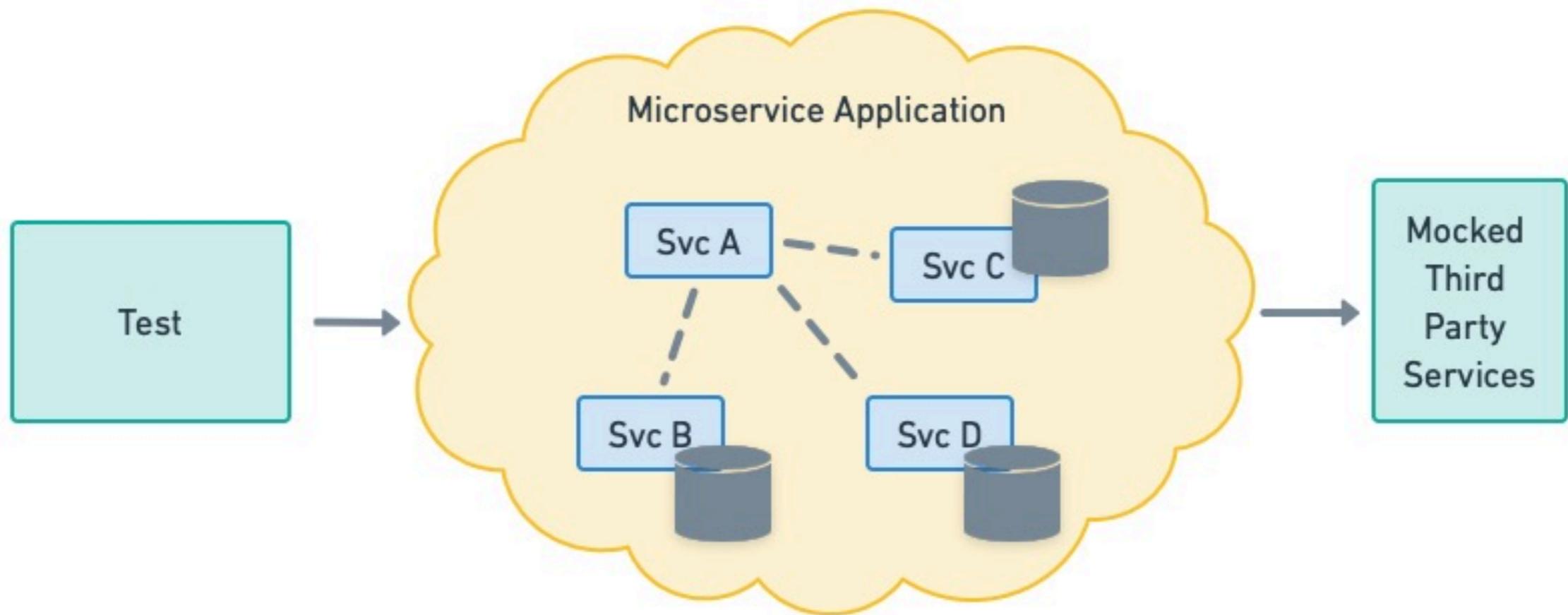
https://docs.pact.io/pact_broker



End-to-End testing



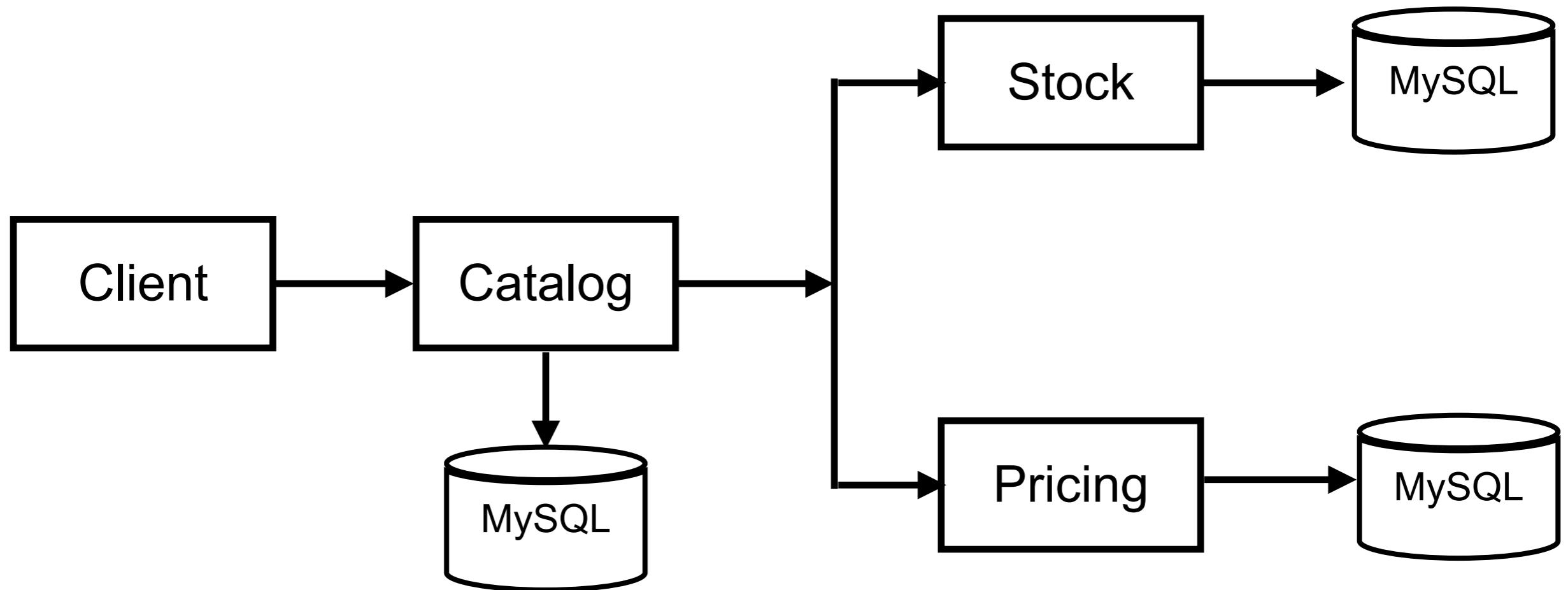
End-to-End testing



What is your testing strategy ?

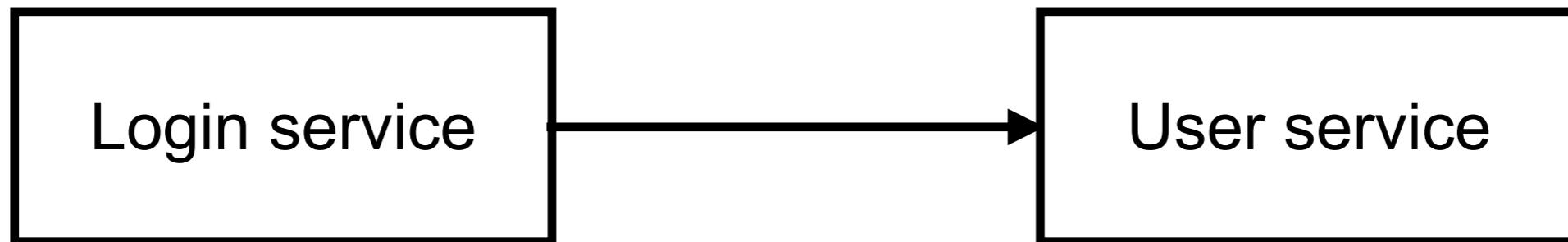


How to test ?



Testing workshop

Component testing
Contract testing



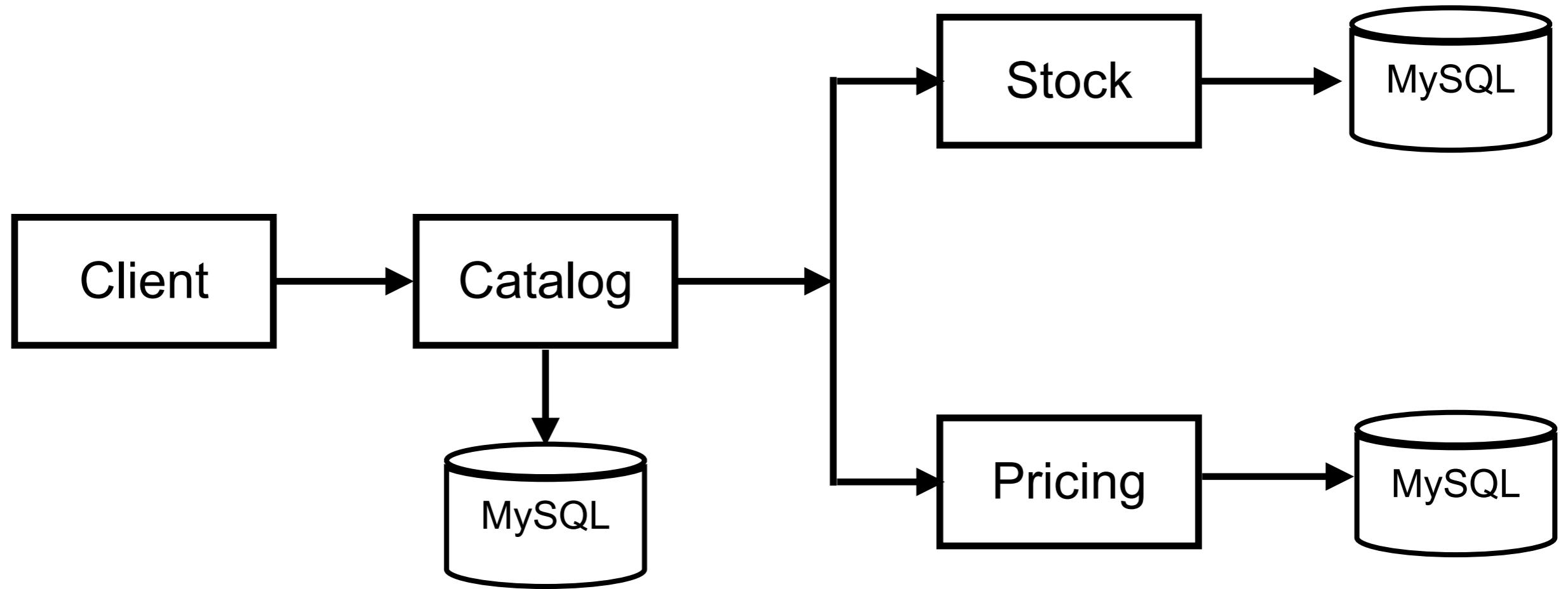
<https://github.com/up1/course-contract-testing>



Performance testing ?
Security testing ?



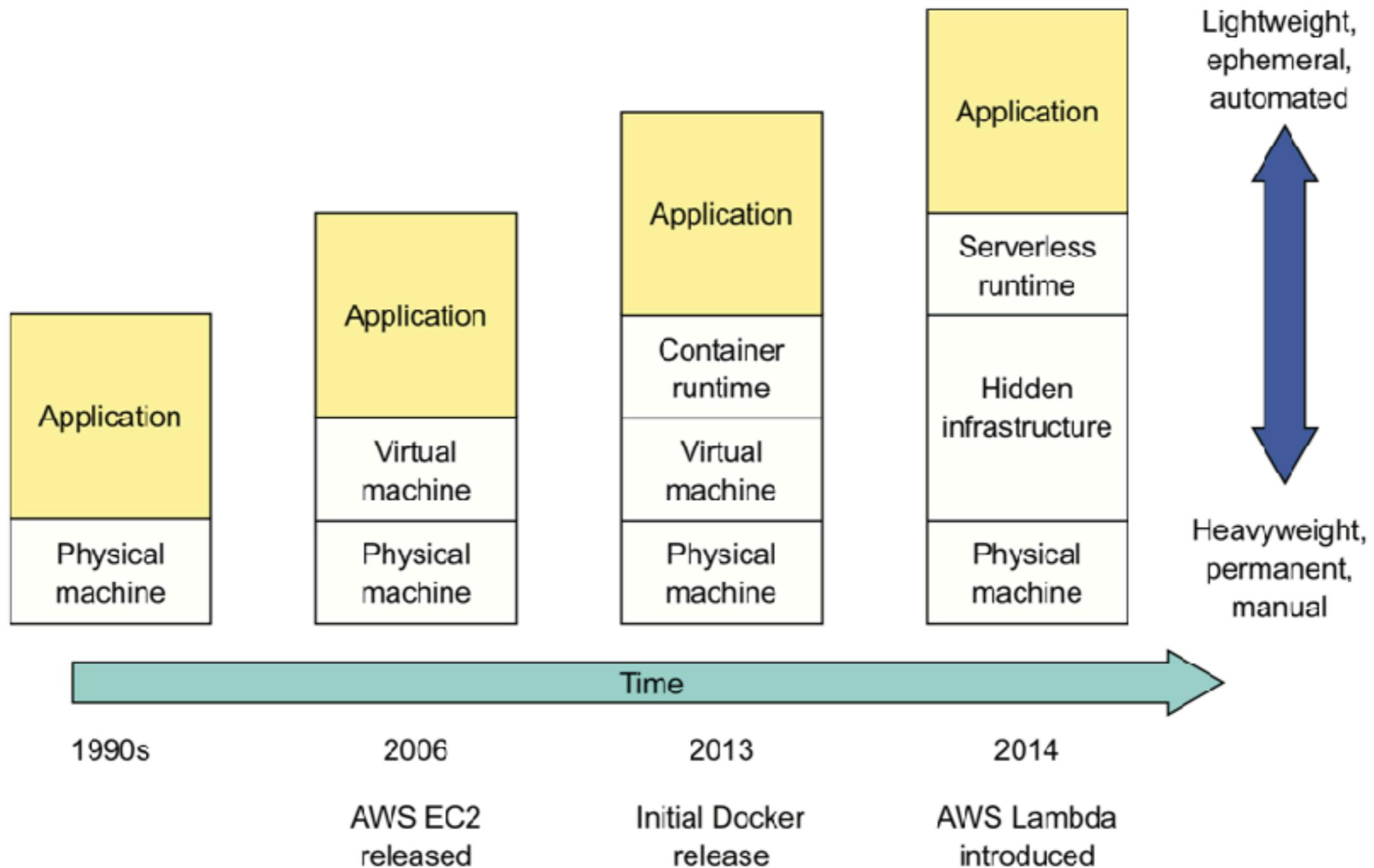
How to performance test ?



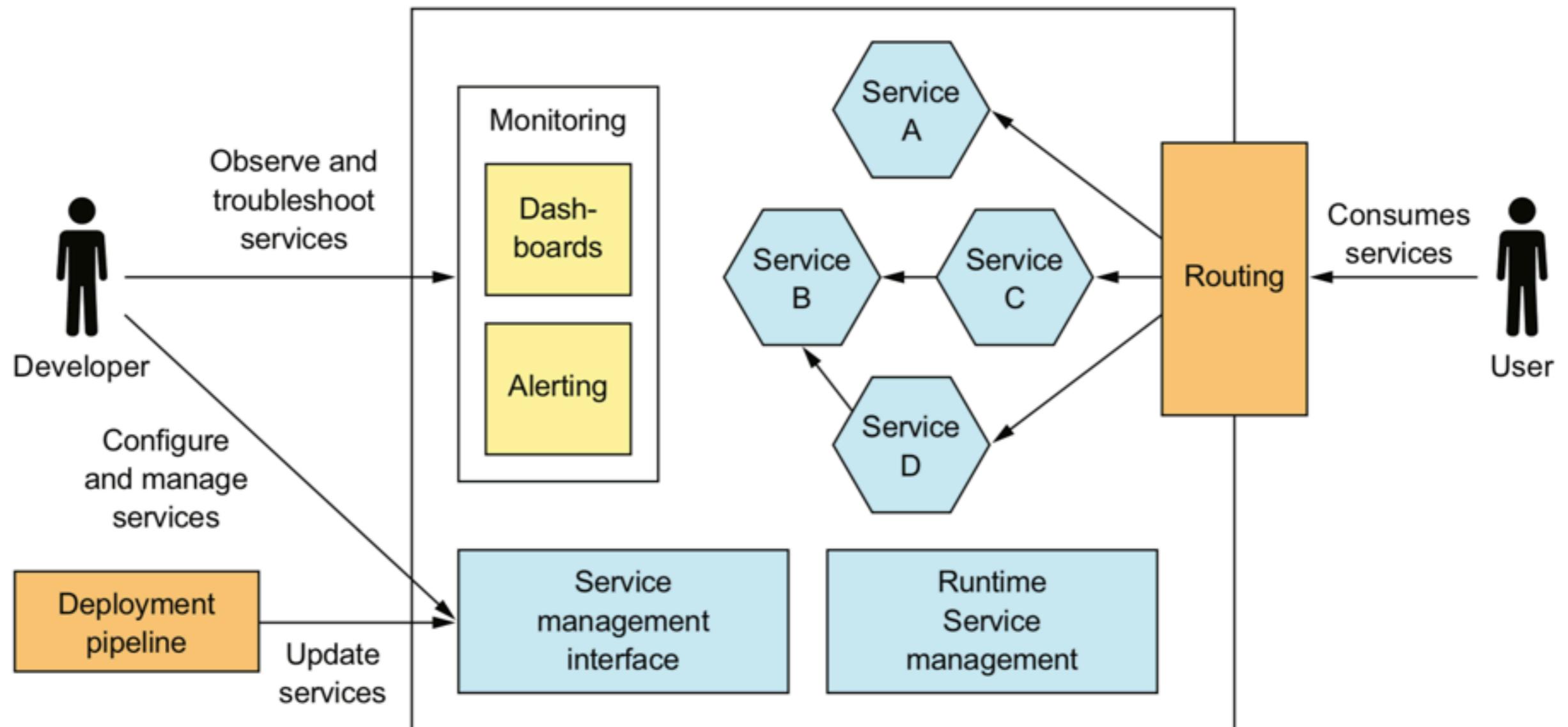
Module 3 : Deploy



Infrastructure



View of production environment



Deploy Microservice

Language-specific packaging

Virtual Machine (VM)

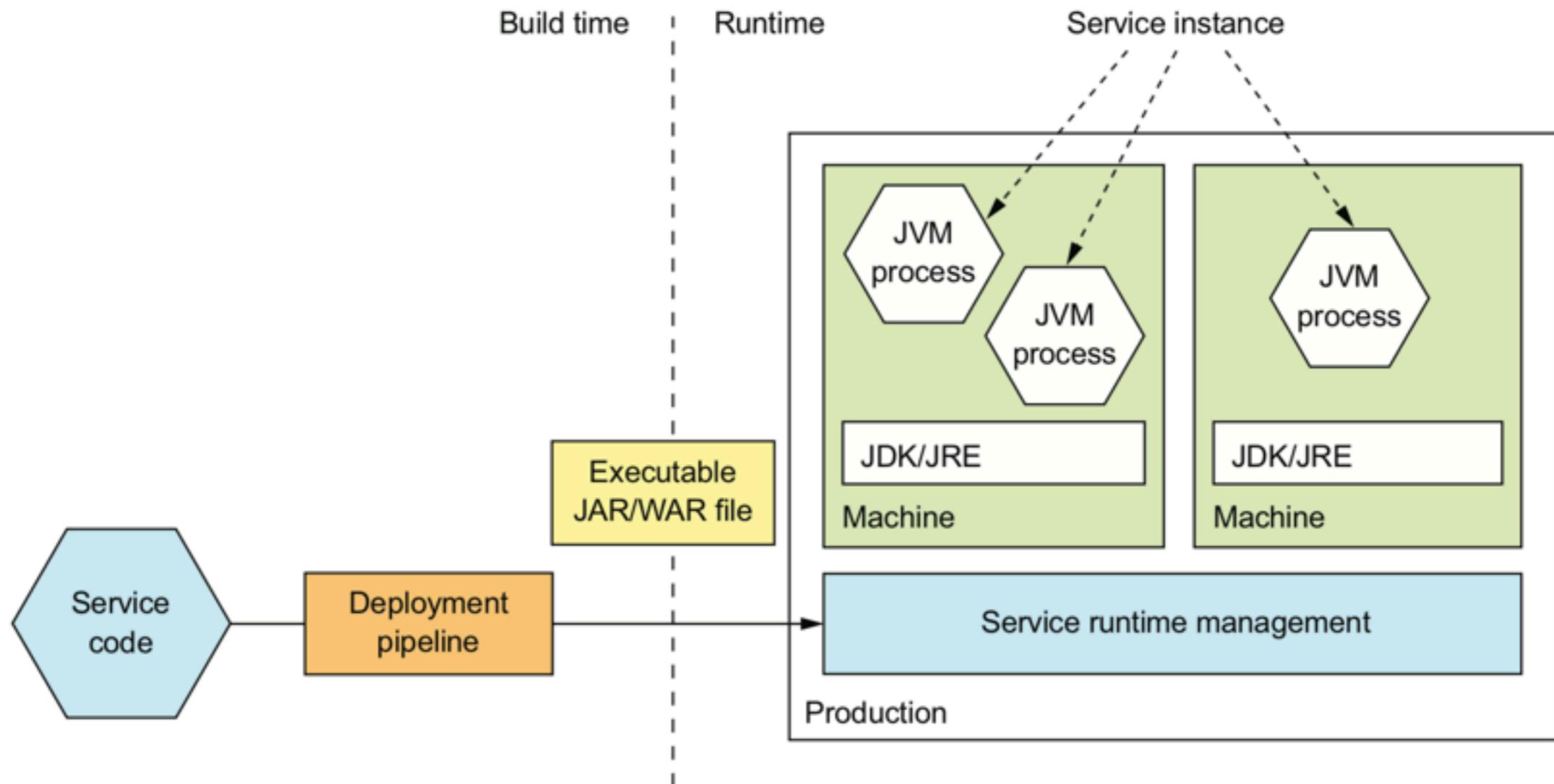
Container

Kubernetes

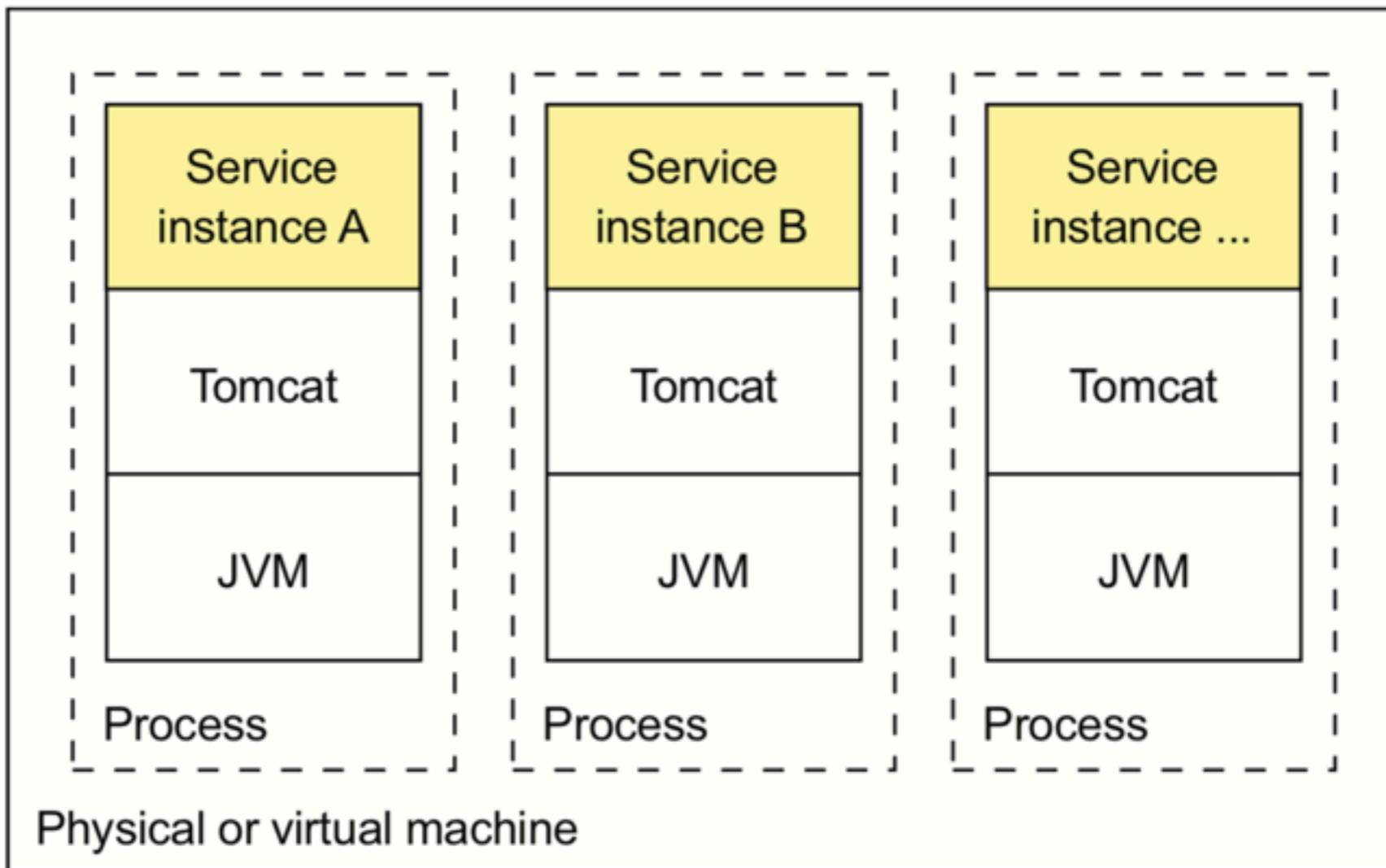
Serverless/FaaS



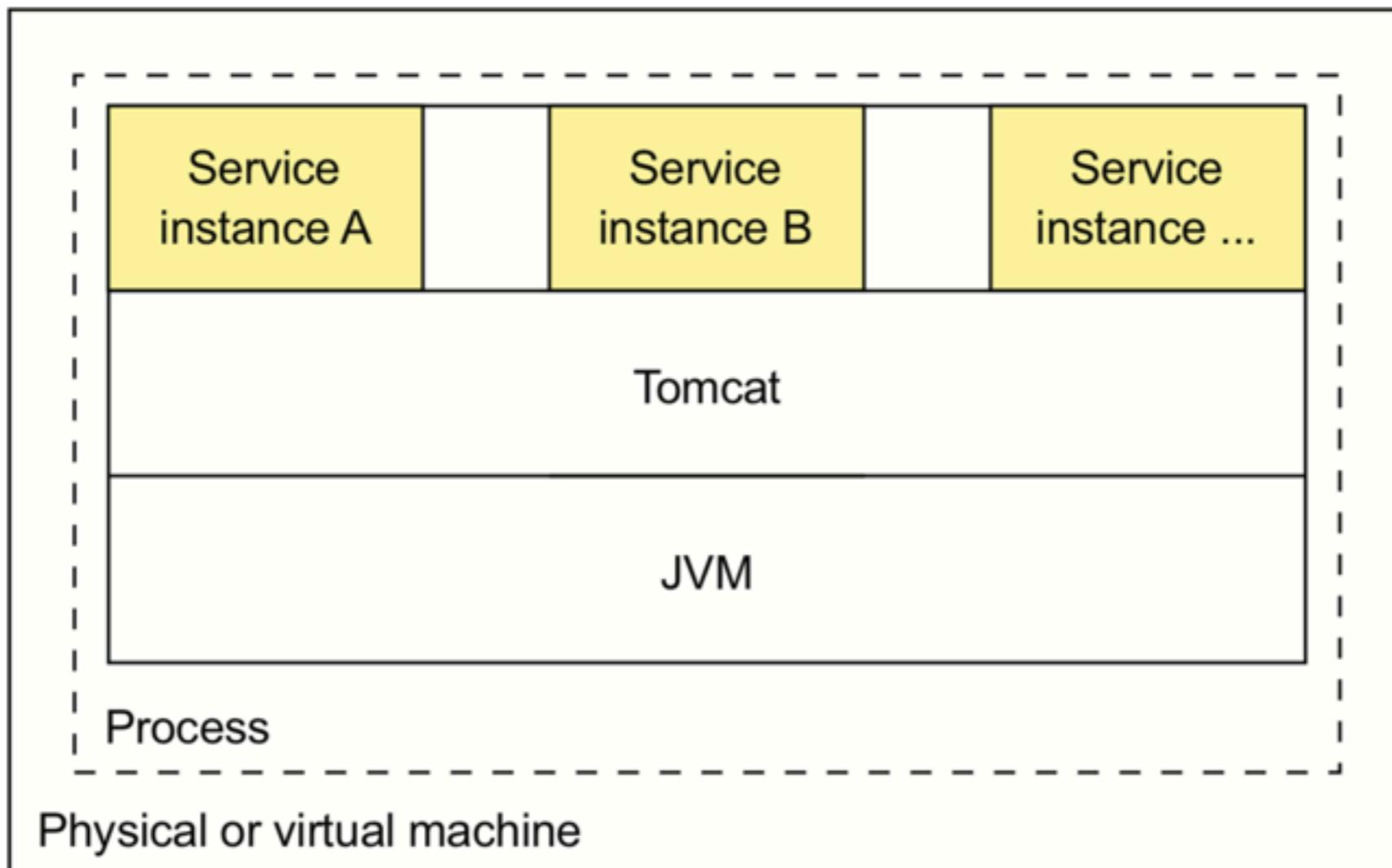
1. Language-specific packaging



Multiple services on same machine



Multiple services on same process



Benefits

Fast deployment

Efficient resource utilization

Service instances's resources are constrained



Drawbacks

Lack of encapsulation of technology stack

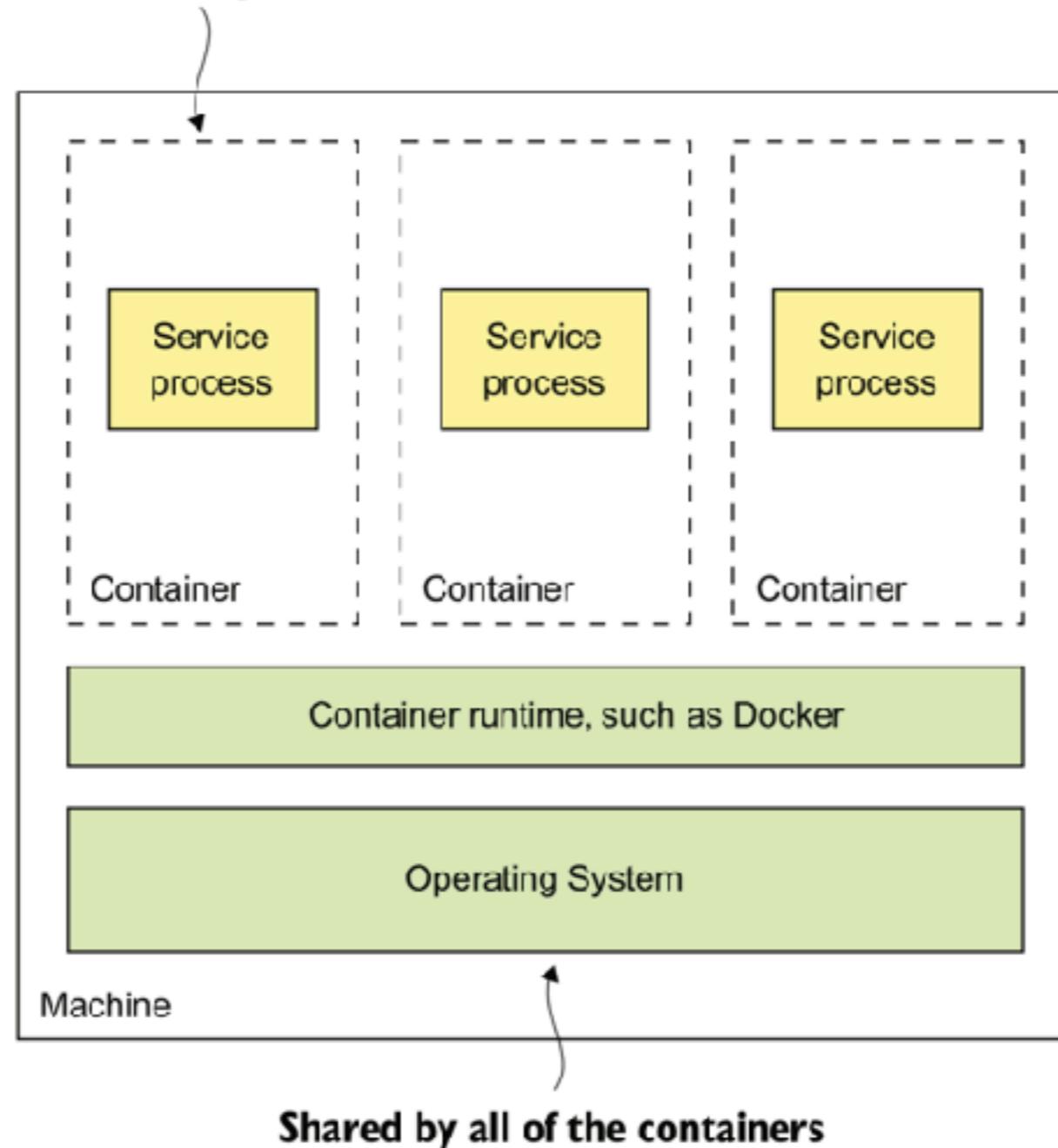
No ability to constrain resources of service

Lack of isolation

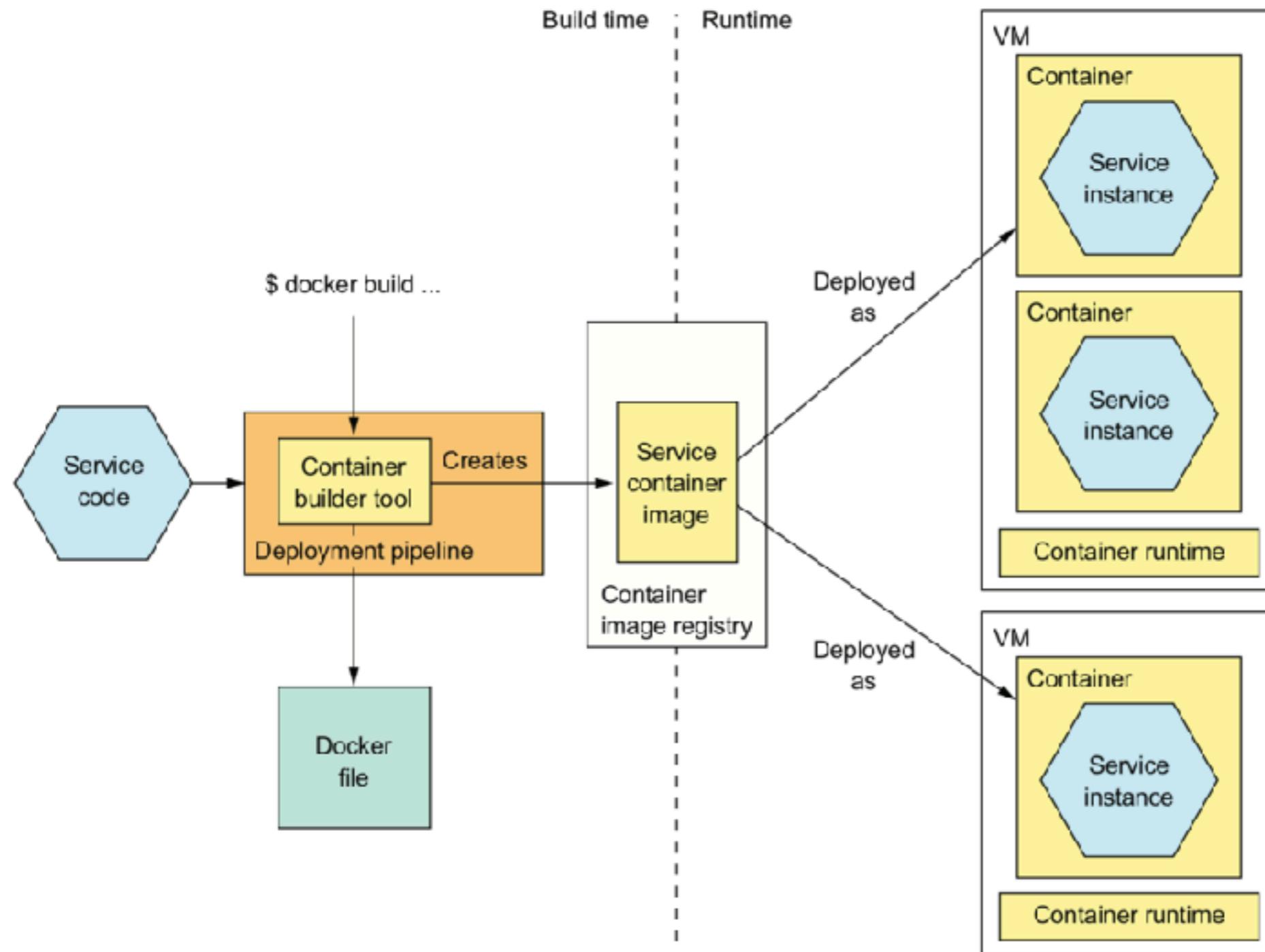


2. Working with container

**Each container is a sandbox
that isolates the processes.**



Deployment with container



Deploy services with Docker

Build a docker image

Push docker image to a registry

Run docker container

Working with docker-compose



Benefits

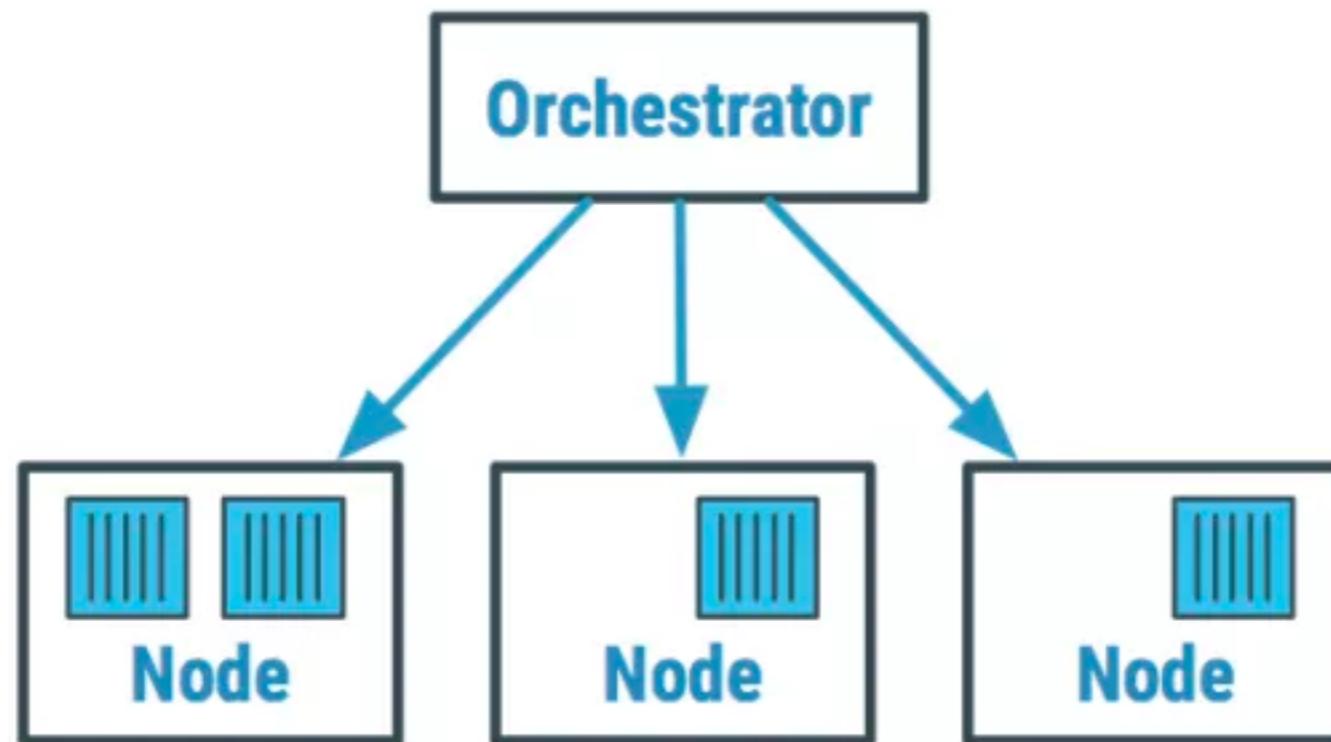
Encapsulate technology stack

Service instances are isolated

Service instances's resources are constrained



Container Orchestration ?



Orchestration tools

Configuration Management



CI/CD orchestration



Container orchestration



Cloud-specific orchestration



PaaS orchestration



Microservices

© 2022 - 2023 Siam Chamnankit Company Limited. All rights reserved.

Application Deployment Strategies



Strategies to deploy

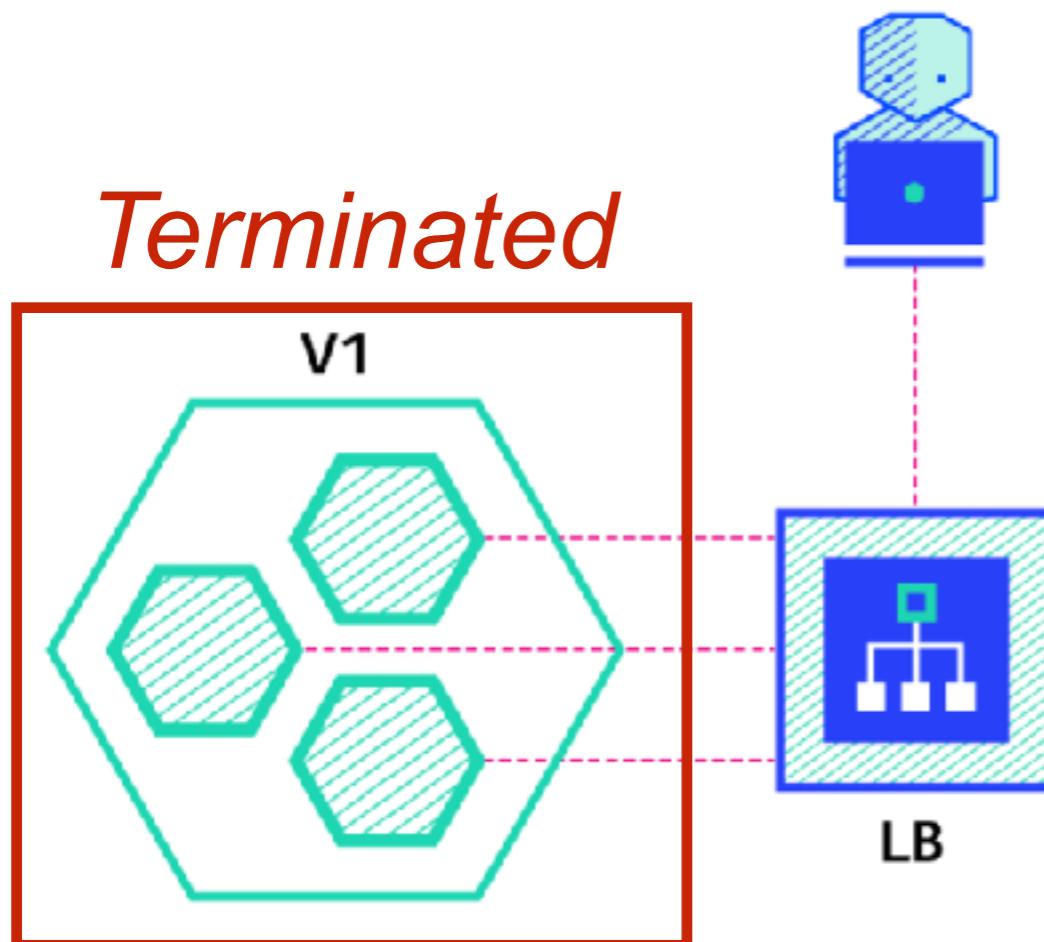
Recreate
Ramped
Blue/Green
Canary
A/B testing
Shadow

<https://thenewstack.io/deployment-strategies/>



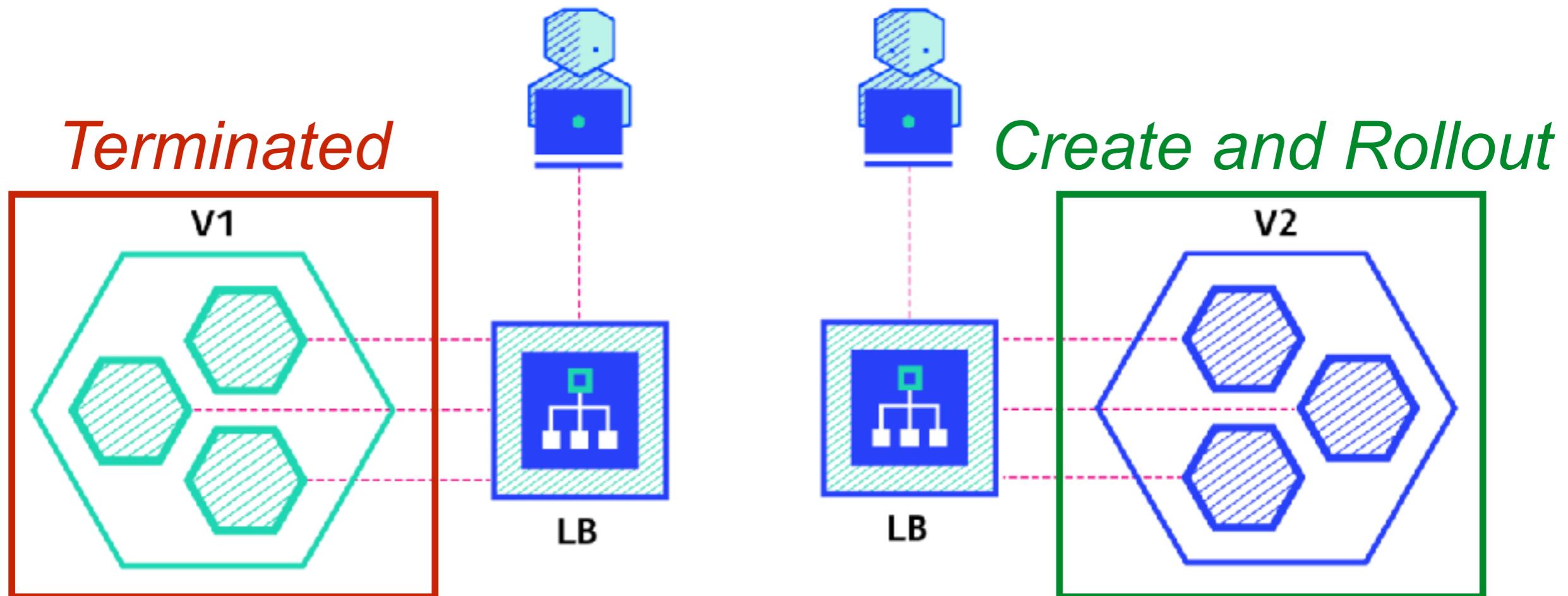
1. Recreate

Version A is terminated then version B is rollout



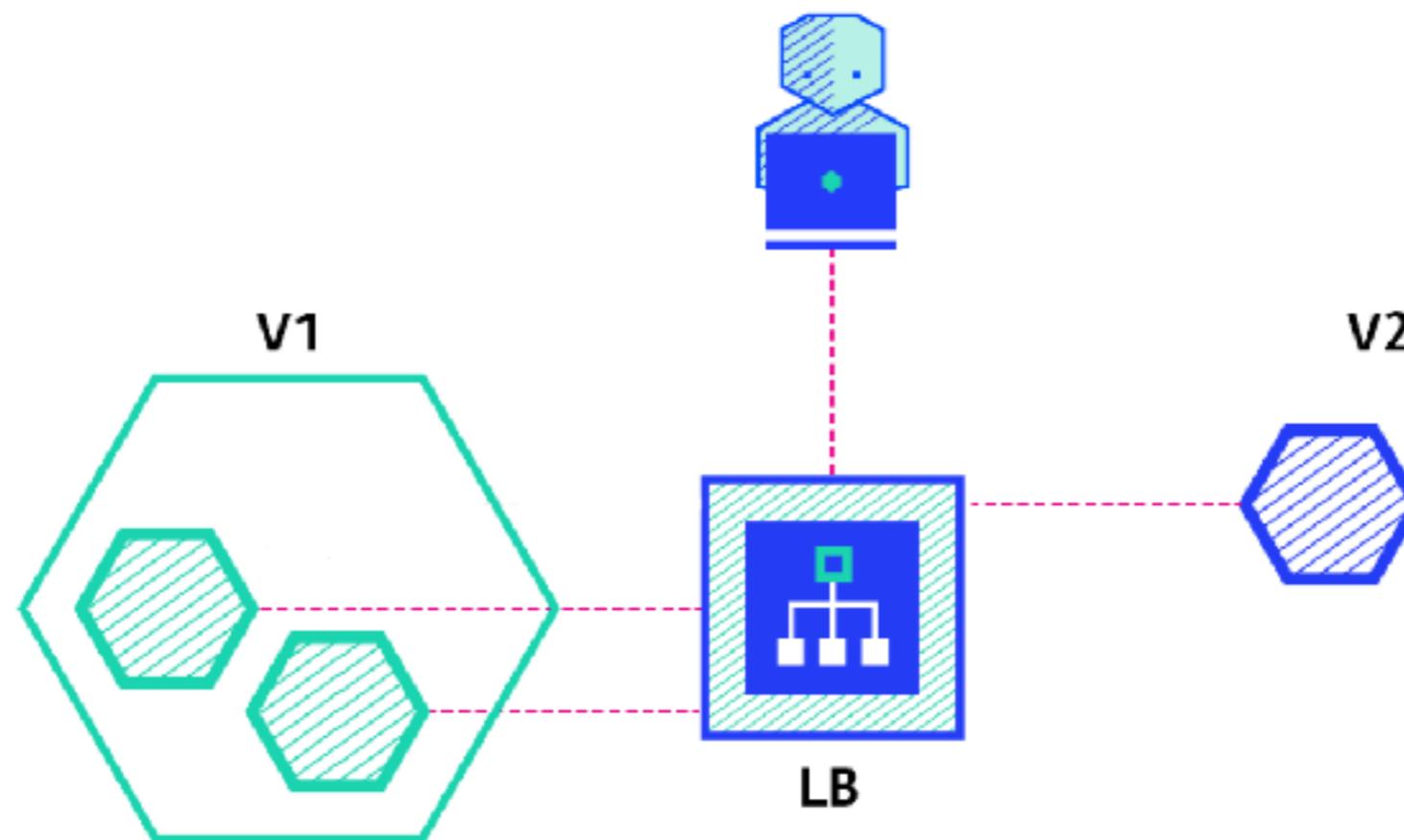
1. Recreate

Version A is terminated then version B is rollout



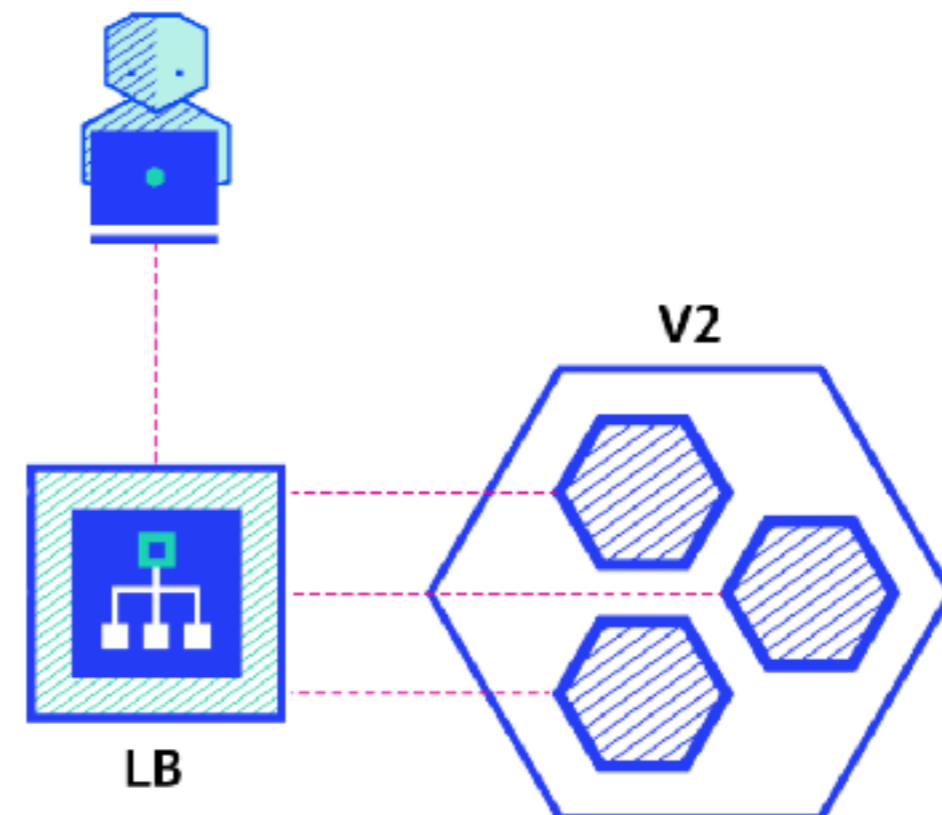
2. Ramped

Slow roll out by replace instance one-by-one



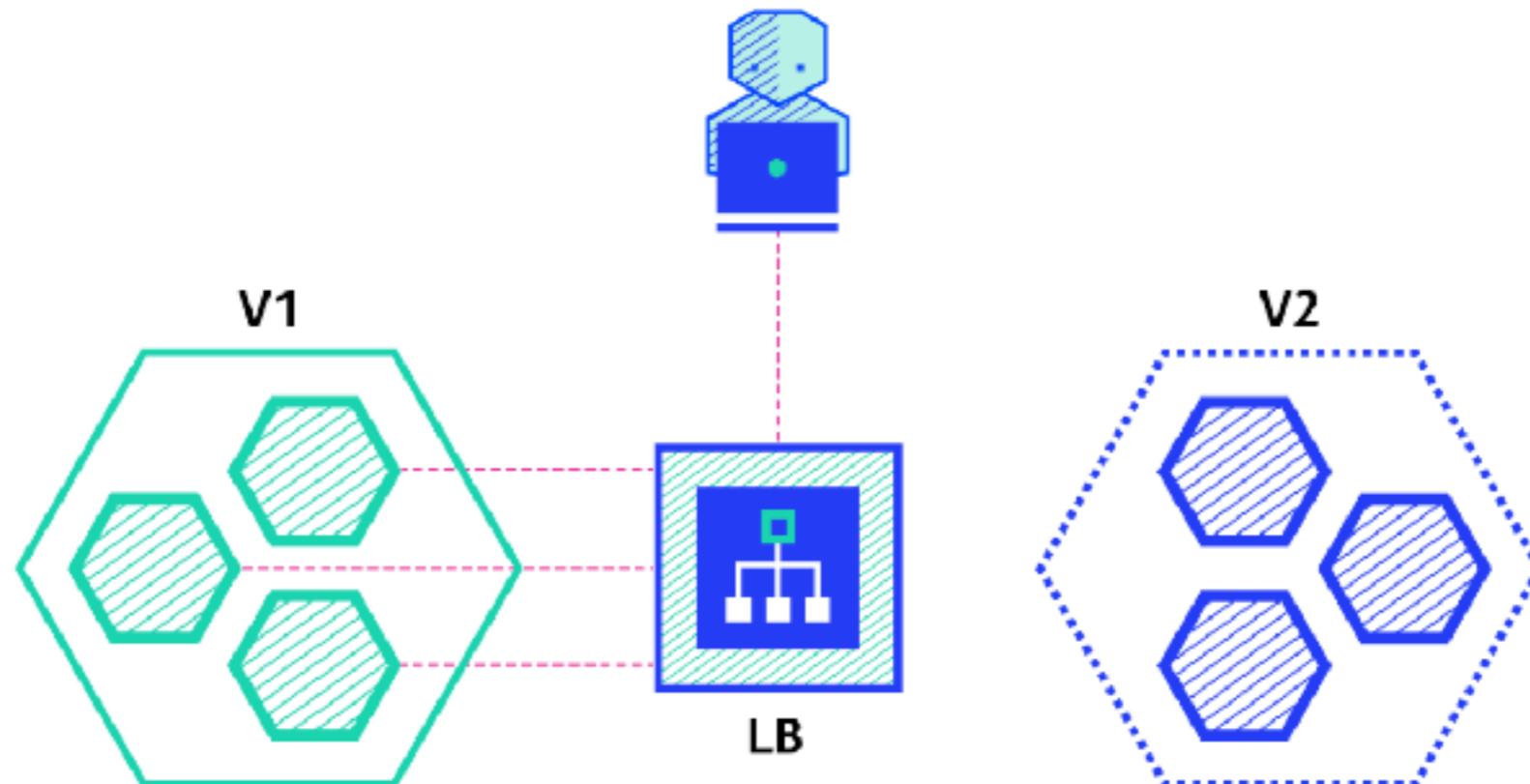
2. Ramped

Slow roll out by replace instance one-by-one

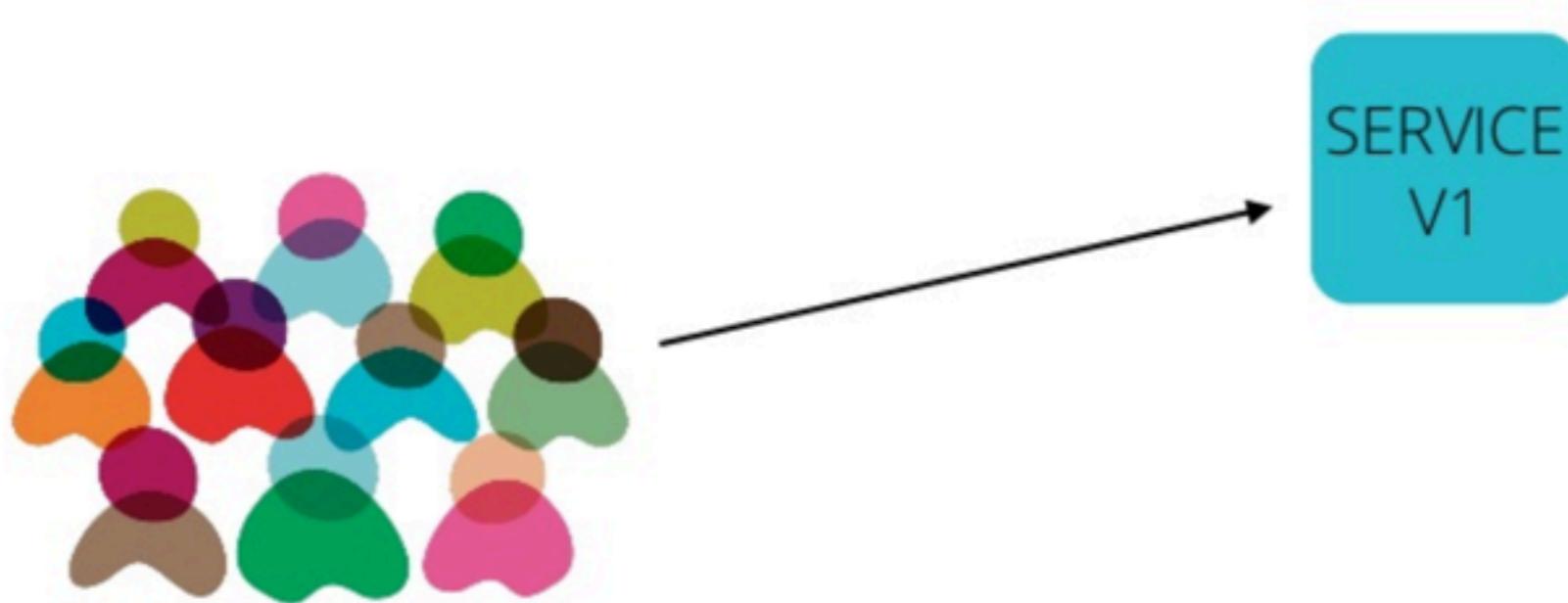


3. Blue/Green

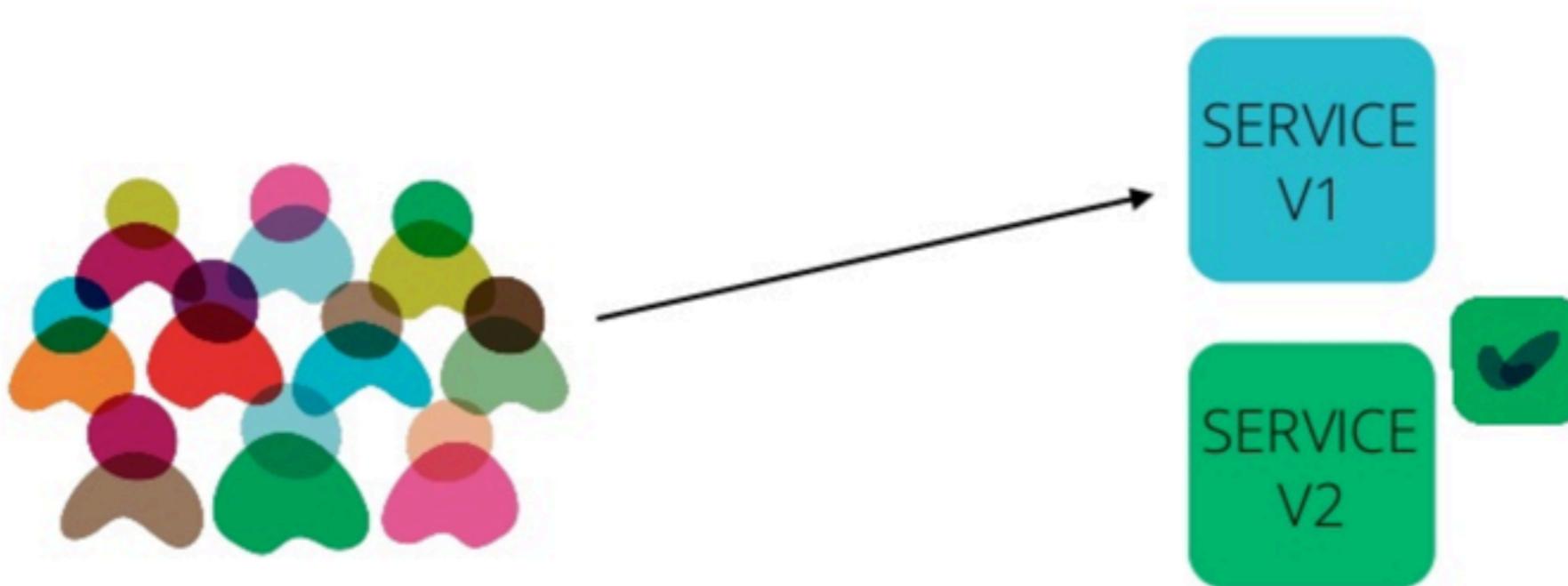
Current version is called **Blue**
New version is called **Green**



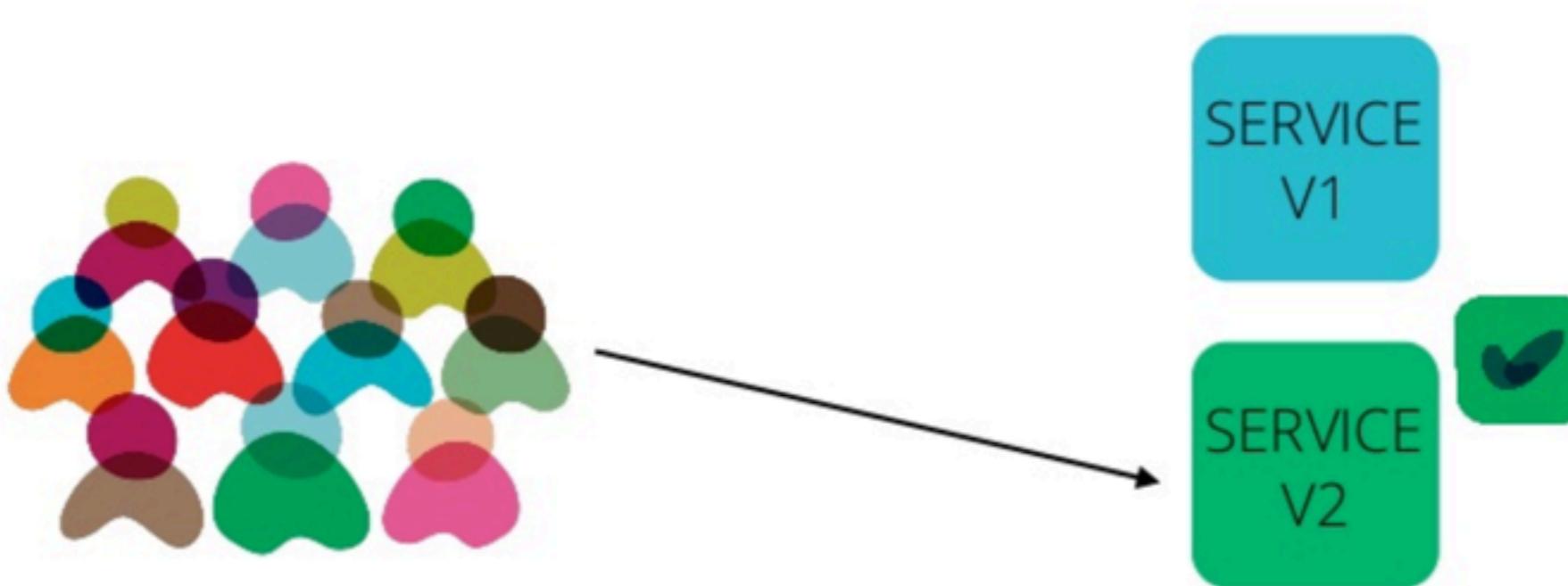
Blue Green Deployment



Blue Green Deployment



Blue Green Deployment

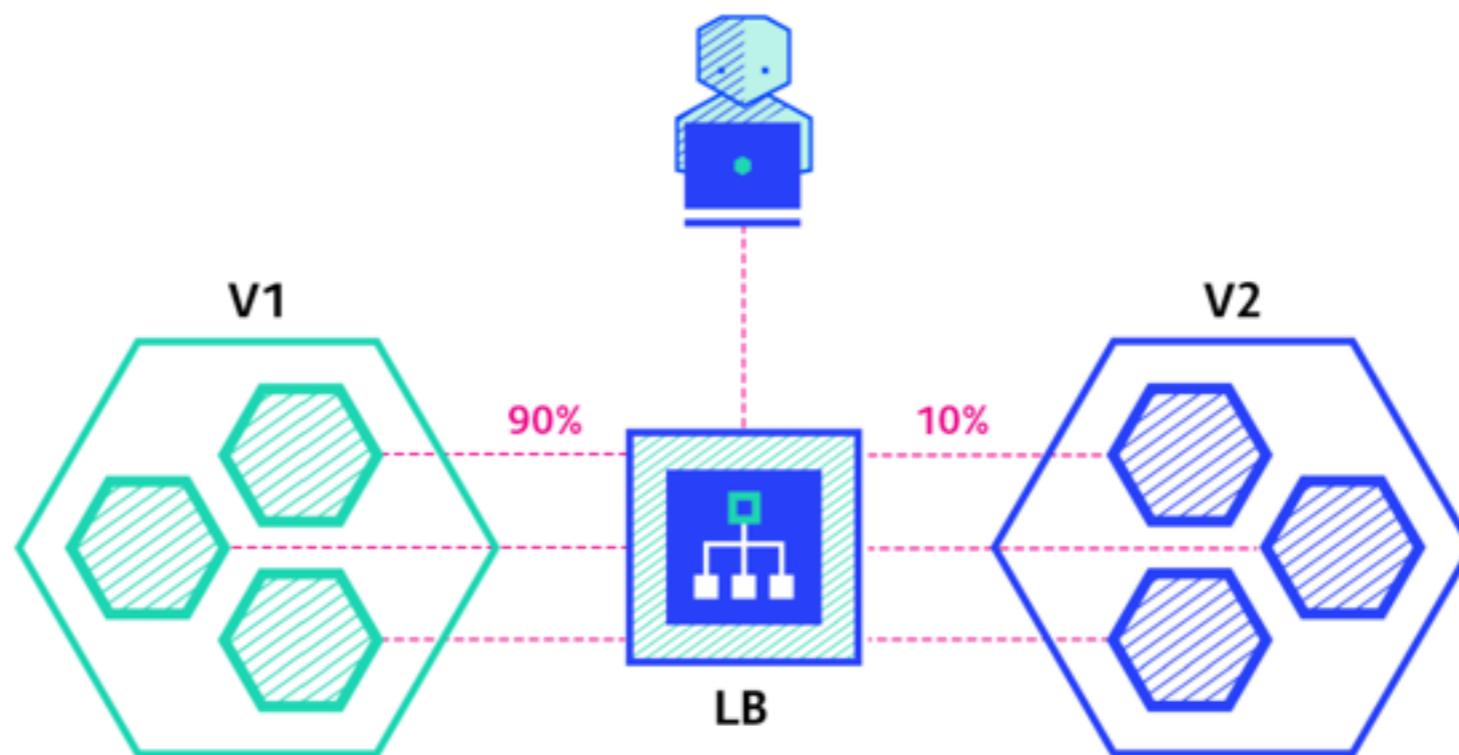


4. Canary

Shift production traffic from version A to B

Traffic is split based-on weight

Use when tests are lacking/not reliable and less confident in system



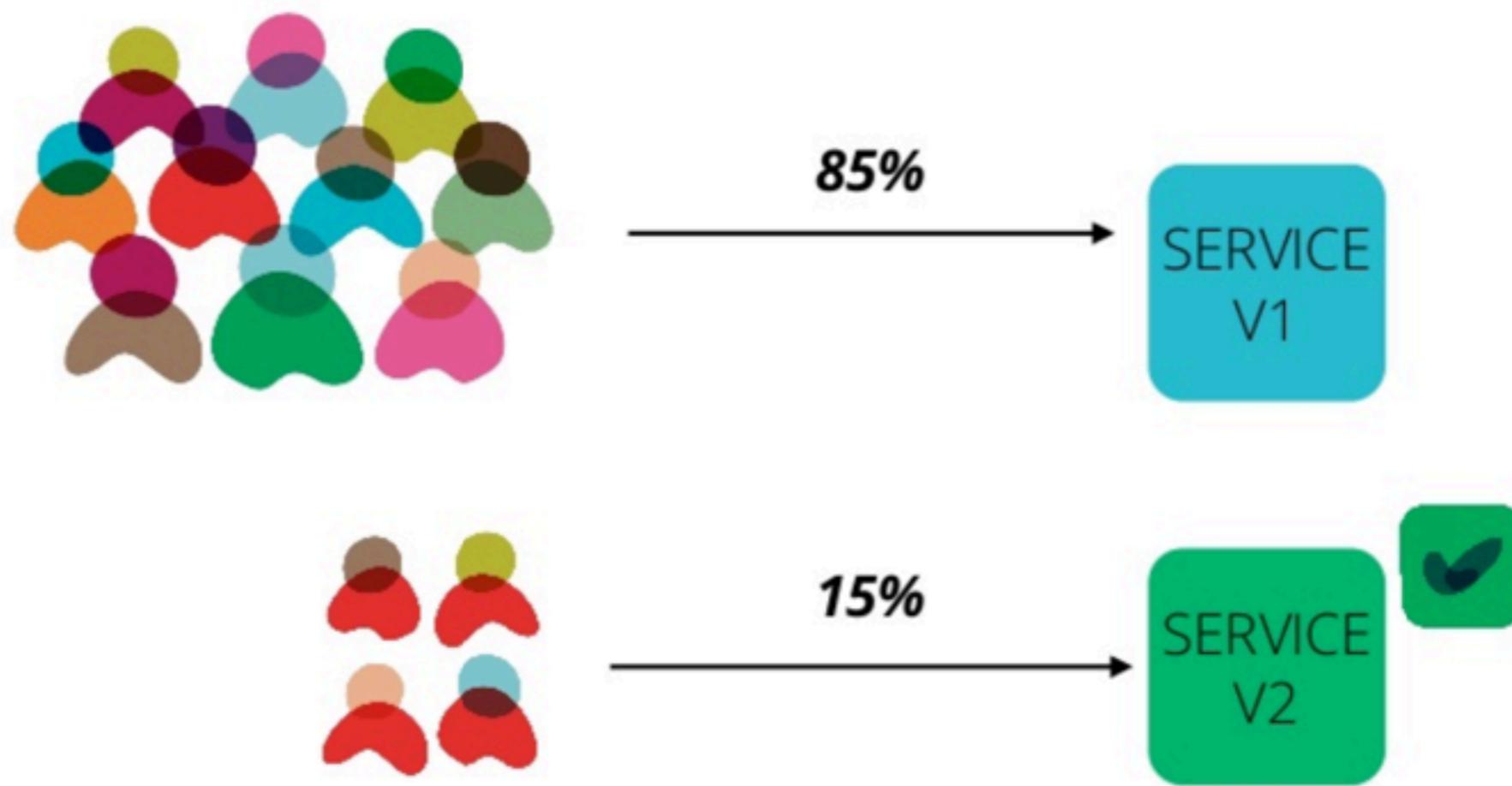
Canary Release



Canary Release

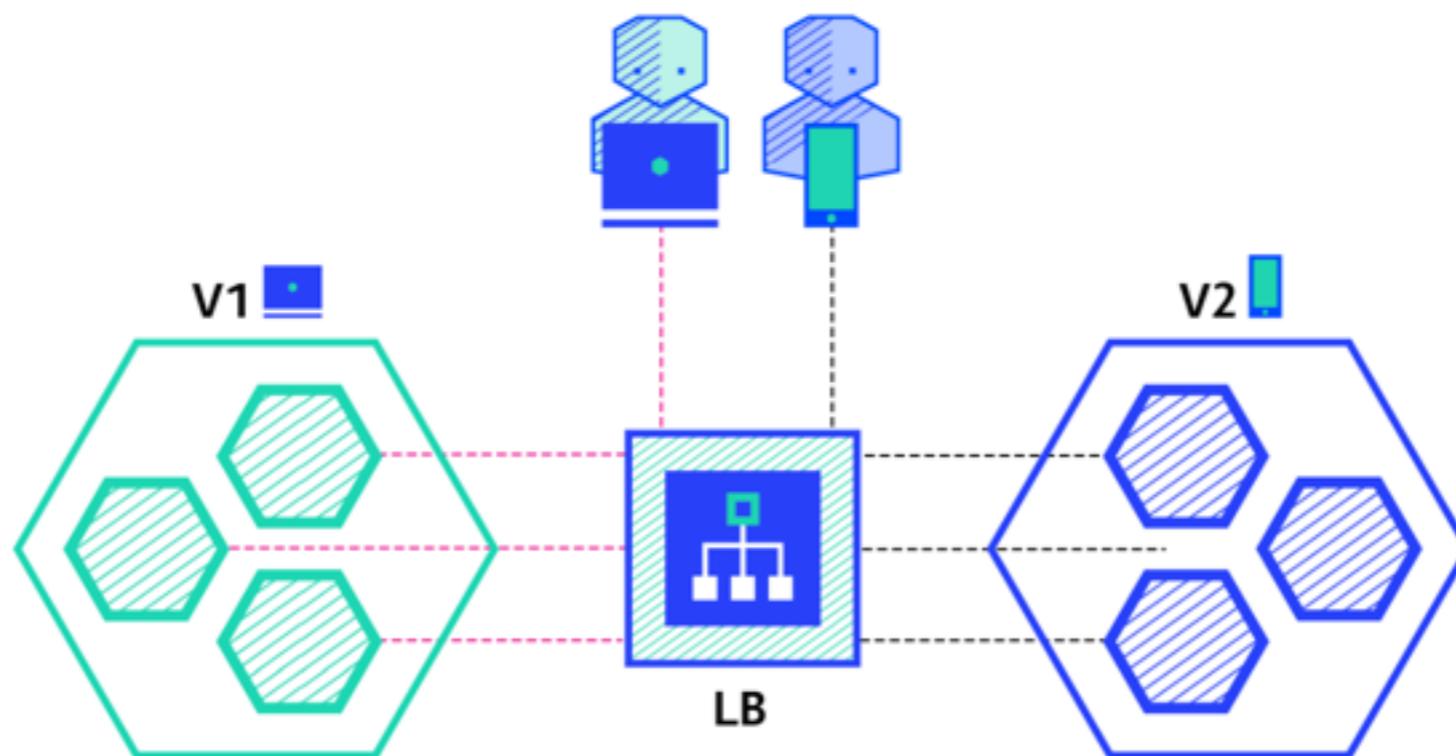


Canary Release



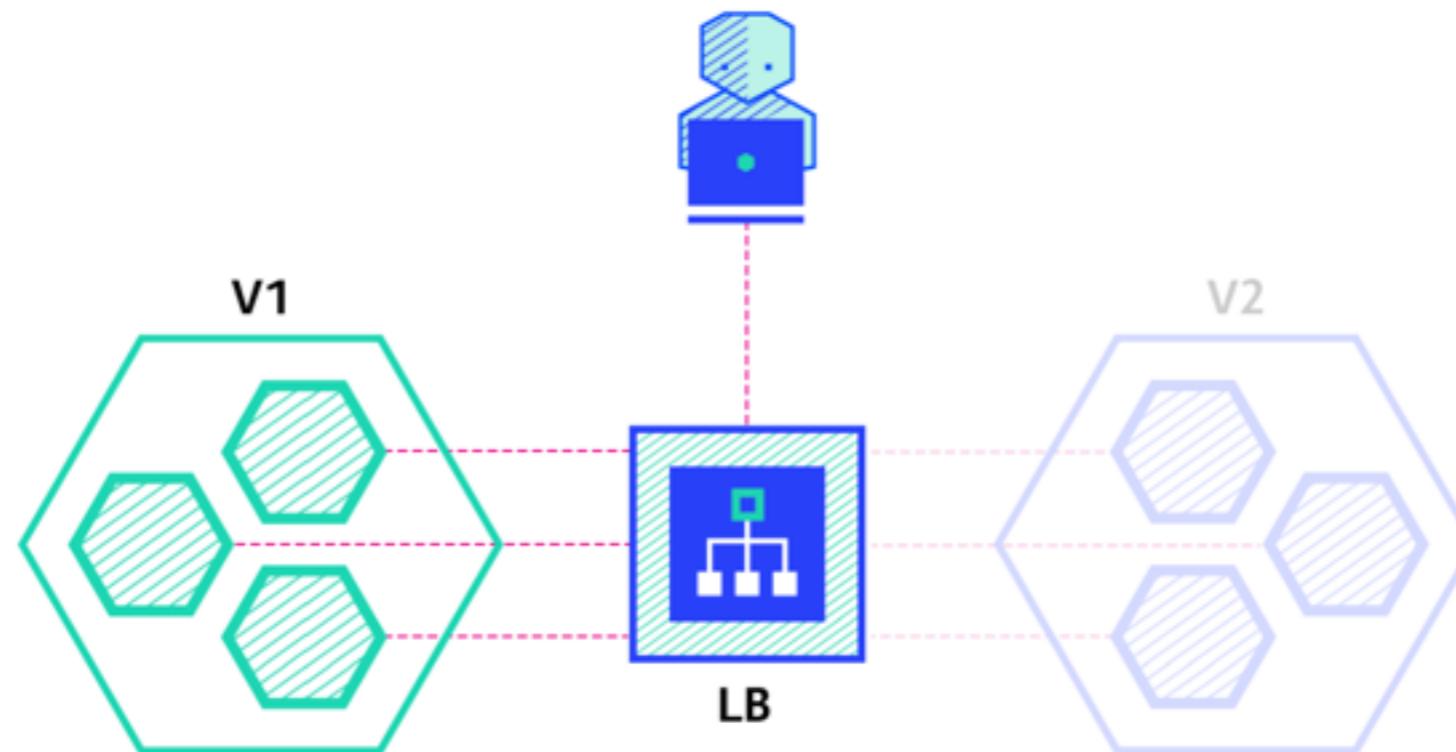
5. A/B testing

Routing the subset of users to new services under the specific condition



6. Shadow

Release version B alongside version A
Send request's A to B without production impact



DEPLOYMENT STRATEGIES

When it comes to production, a ramped or blue/green deployment is usually a good fit, but proper testing of the new platform is necessary.

Blue/green and shadow strategies have more impact on the budget as it requires double resource capacity. If the application lacks in tests or if there is little confidence about the impact/stability of the software, then a canary, a/b testing or shadow release can be used.

If your business requires testing of a new feature amongst a specific pool of users that can be filtered depending on some parameters like geolocation, language, operating system or browser features, then you may want to use the a/b testing technique.



Strategy	ZERO DOWNTIME	REAL TRAFFIC TESTING	TARGETED USERS	CLOUD COST	ROLLBACK DURATION	NEGATIVE IMPACT ON USER	COMPLEXITY OF SETUP
RECREATE version A is terminated then version B is rolled out	✗	✗	✗	■ ■ ■	■ ■ ■	■ ■ ■	□ □ □
RAMPED version B is slowly rolled out and replacing version A	✓	✗	✗	■ ■ ■	■ ■ ■	■ □ □	■ □ □
BLUE/GREEN version B is released alongside version A, then the traffic is switched to version B	✓	✗	✗	■ ■ ■	□ □ □	■ ■ □	■ ■ □
CANARY version B is released to a subset of users, then proceed to a full rollout	✓	✓	✗	■ ■ ■	□ □ □	■ □ □	■ ■ □
A/B TESTING version B is released to a subset of users under specific condition	✓	✓	✓	■ ■ ■	□ □ □	■ □ □	■ ■ ■
SHADOW version B receives real world traffic alongside version A and doesn't impact the response	✓	✓	✗	■ ■ ■	□ □ □	□ □ □	■ ■ ■



Deployment workshop with



<https://github.com/up1/workshop-develop-microservices-2023>



Design CI/CD process

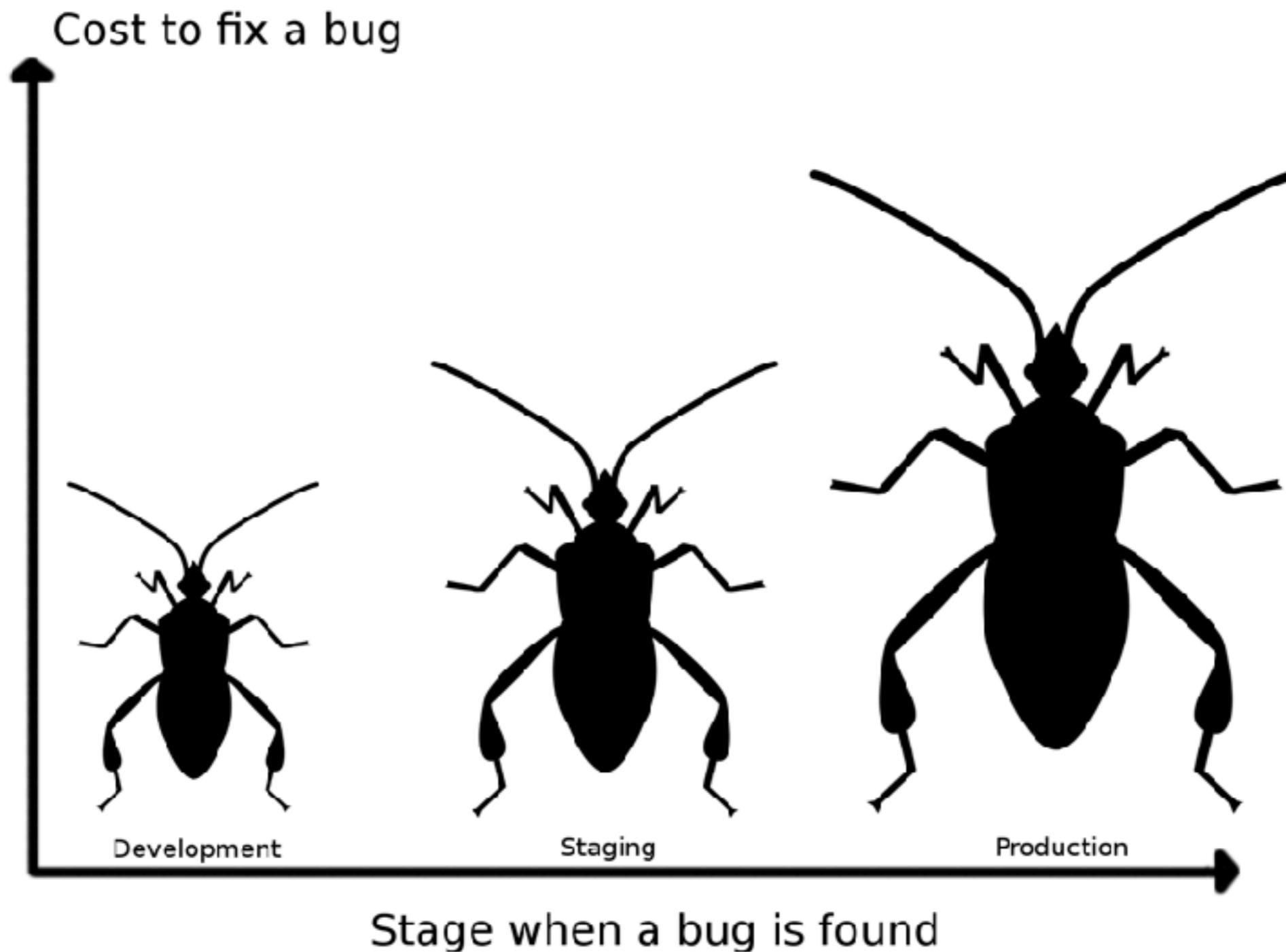


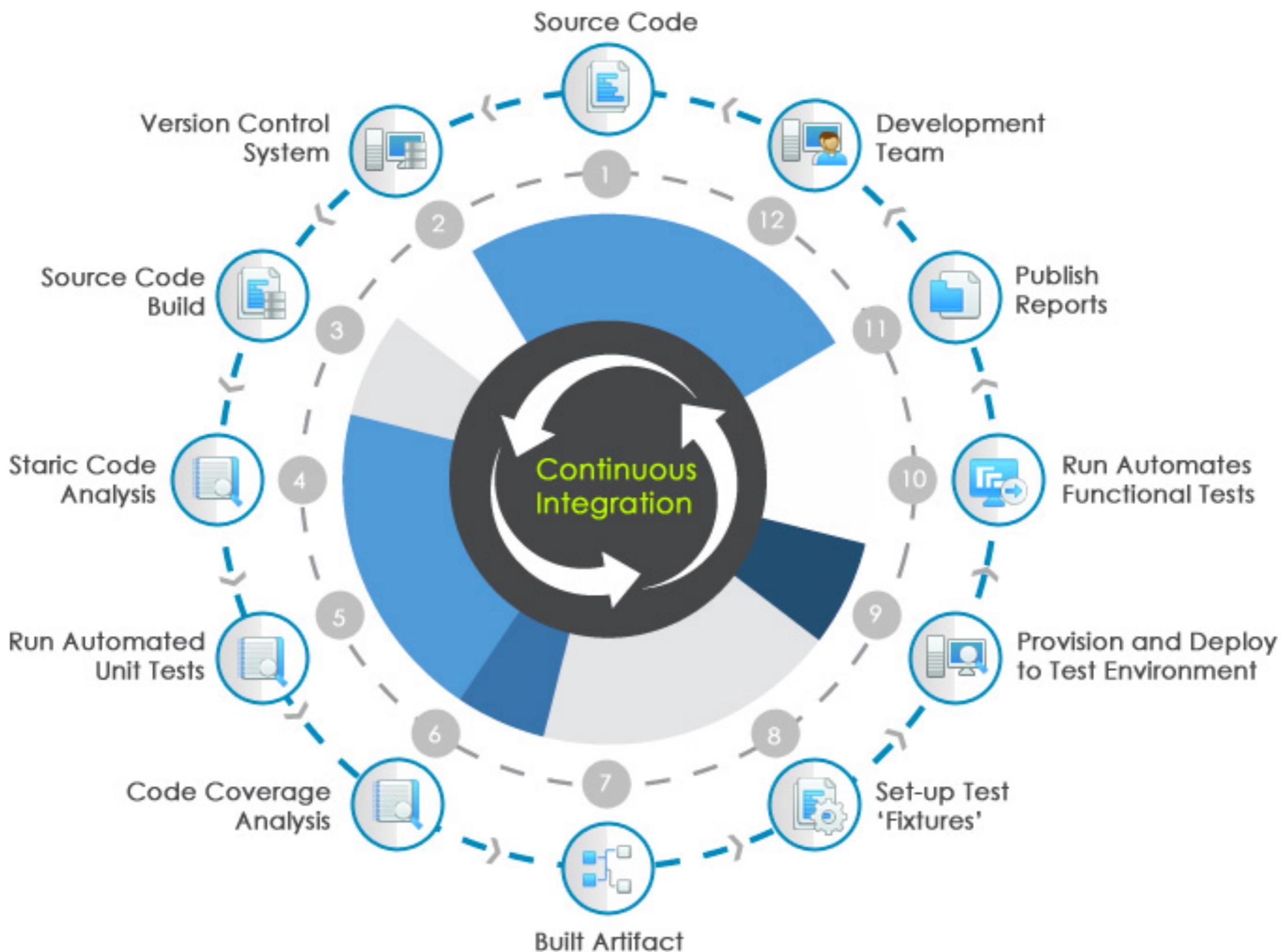
The cost of integration

1. Merging the code
2. Duplicate changes
3. Test again again !!
4. Fixing bugs
5. Impact on stability



The cost of integration







Jenkins

Bamboo

CI is about **what people do**
not about **what tools they use**



Hudson



Continuous Integration

Discipline to integrate frequently



Continuous Integration

Strive to make **small change**

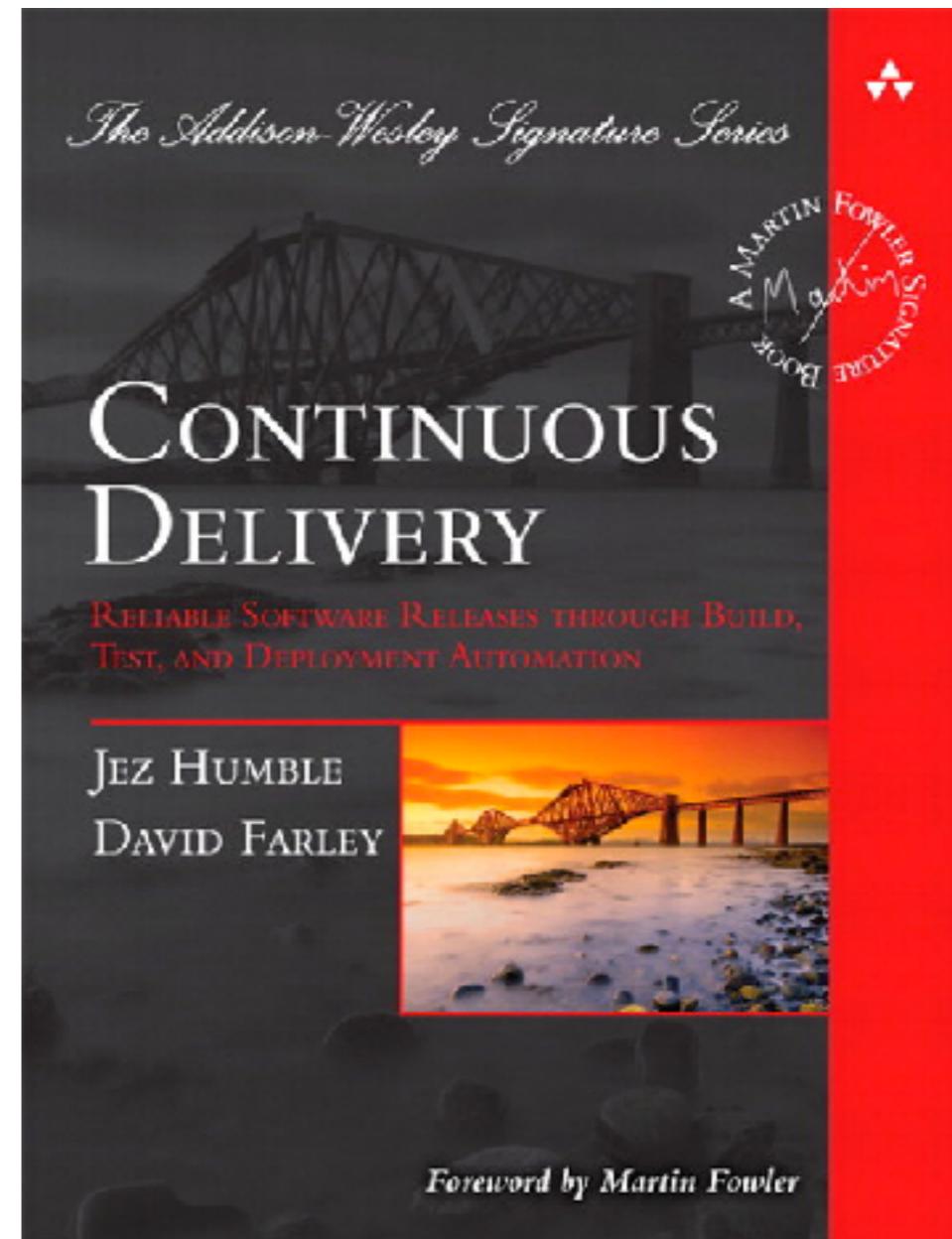
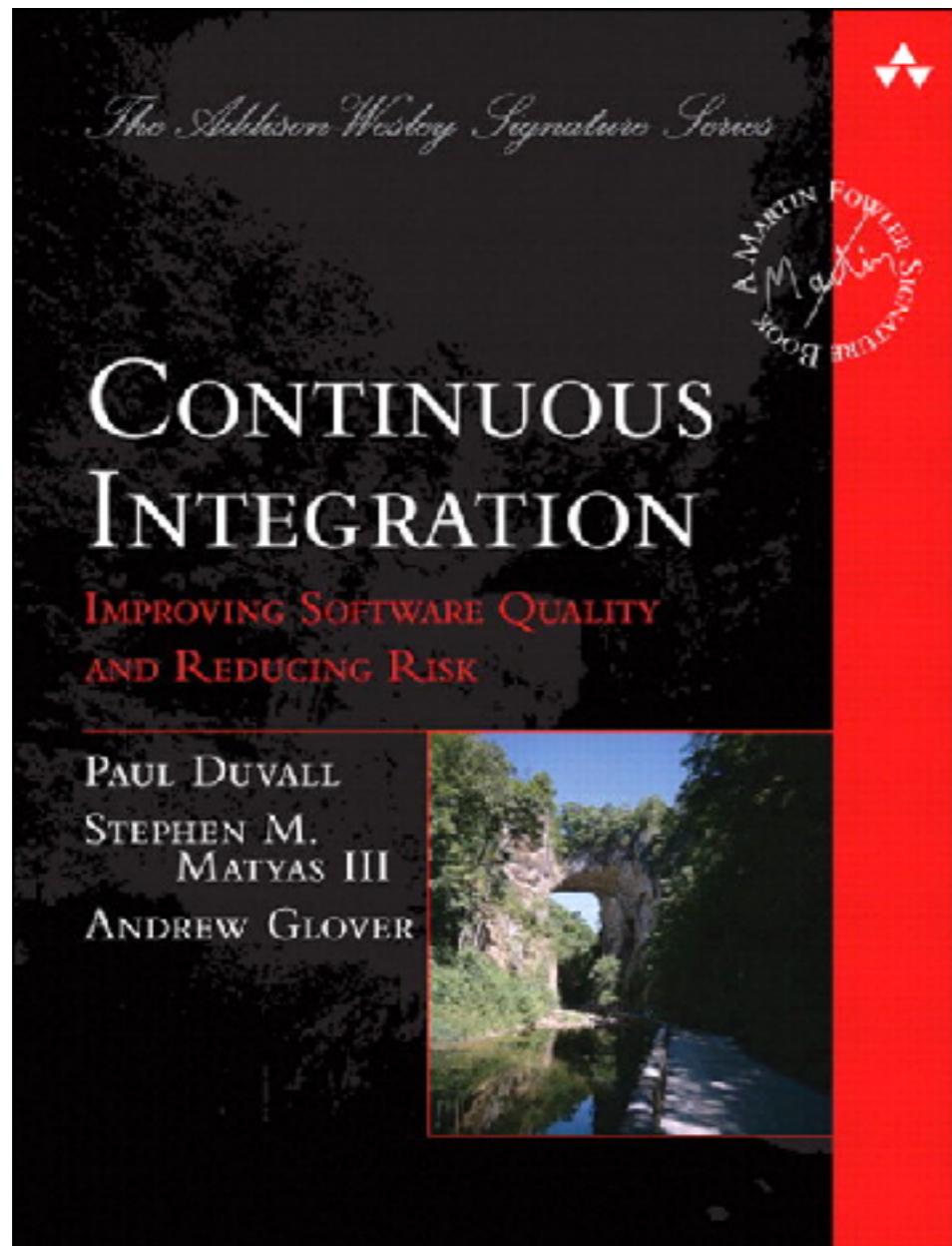


Continuous Integration

Strive for **fast feedback**



Improve quality and reduce risk



Microservices

© 2022 - 2023 Siam Chamnankit Company Limited. All rights reserved.

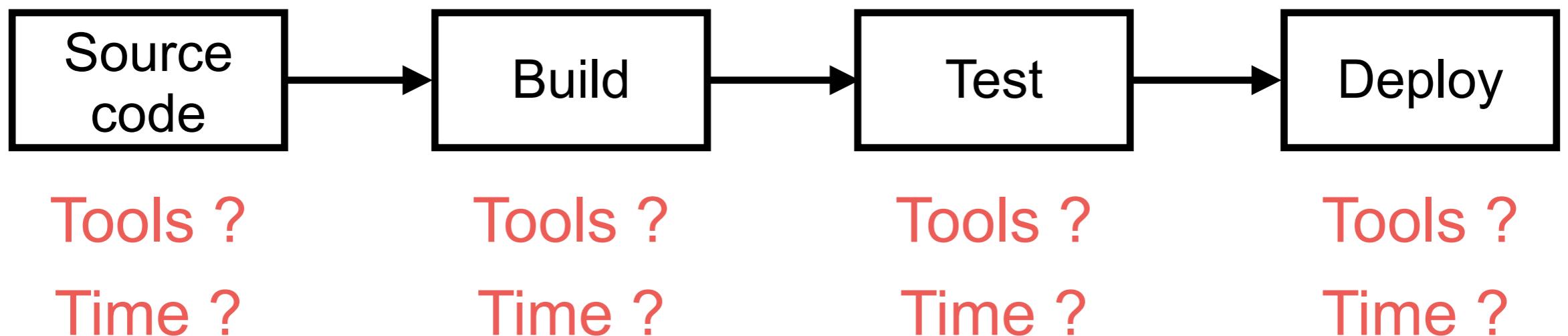
Design pipeline

Path to Production !!



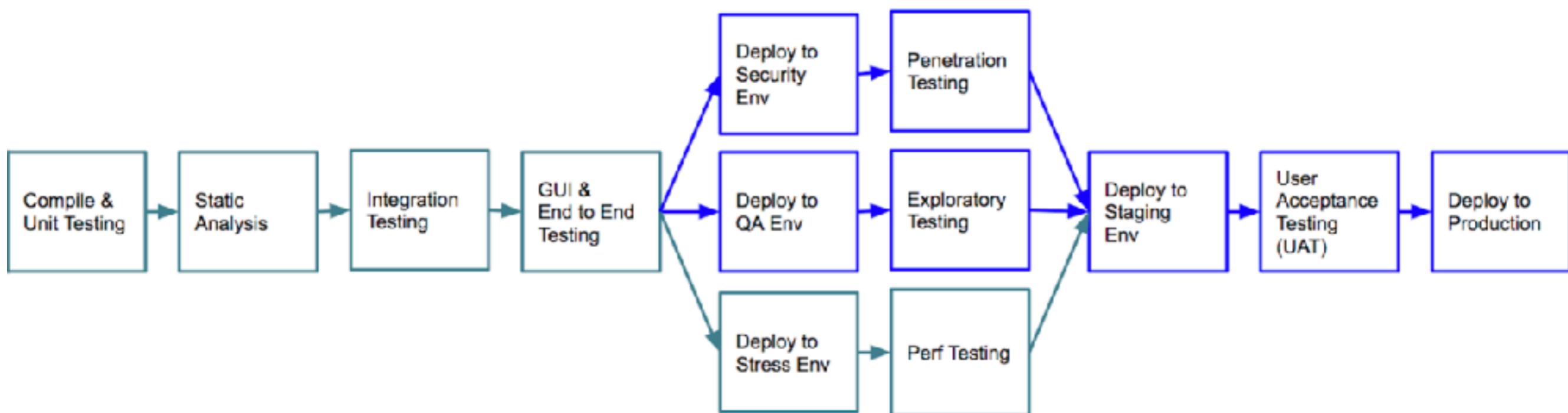
Pipeline of services

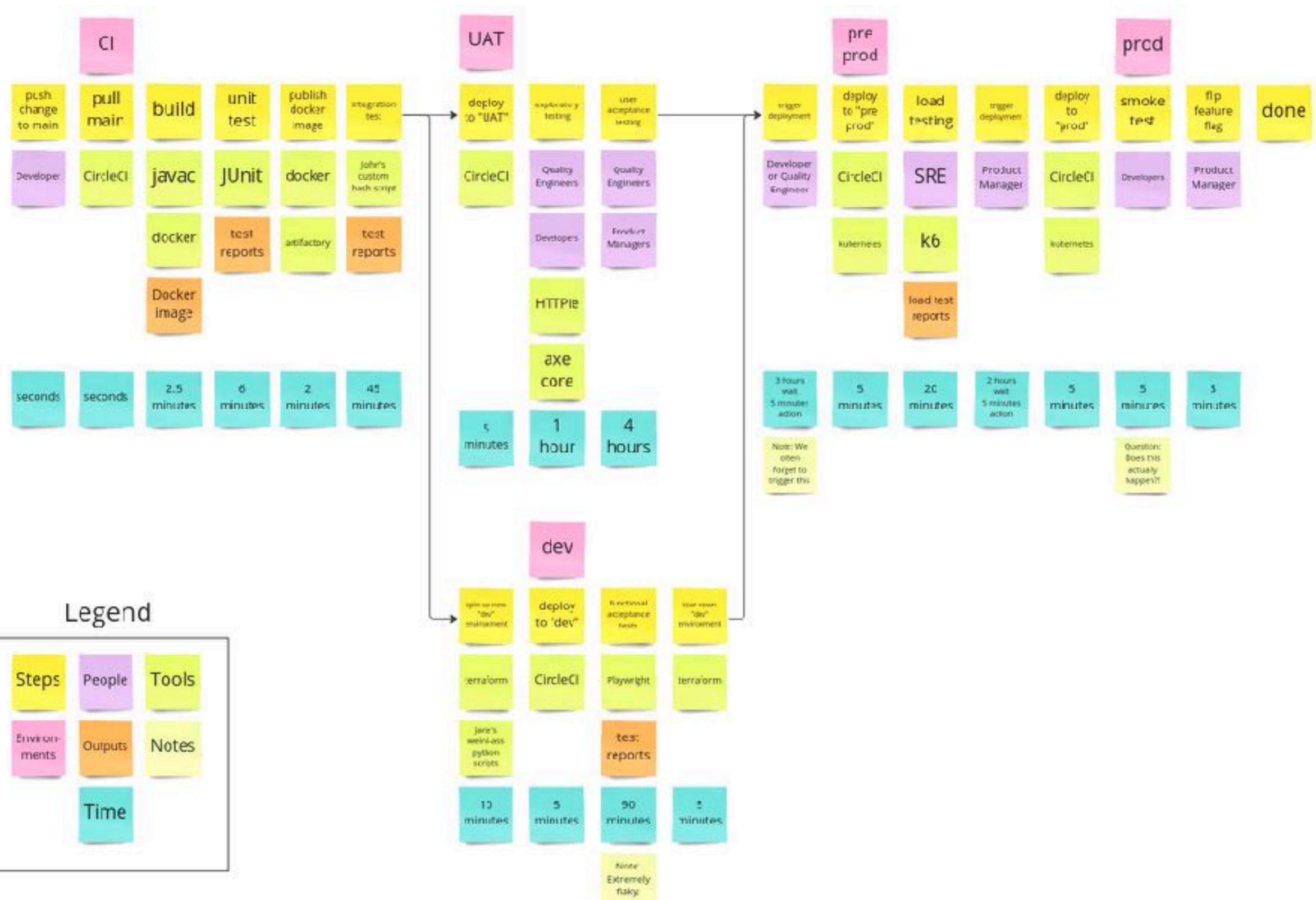
Fast feedback and high quality
Automation process



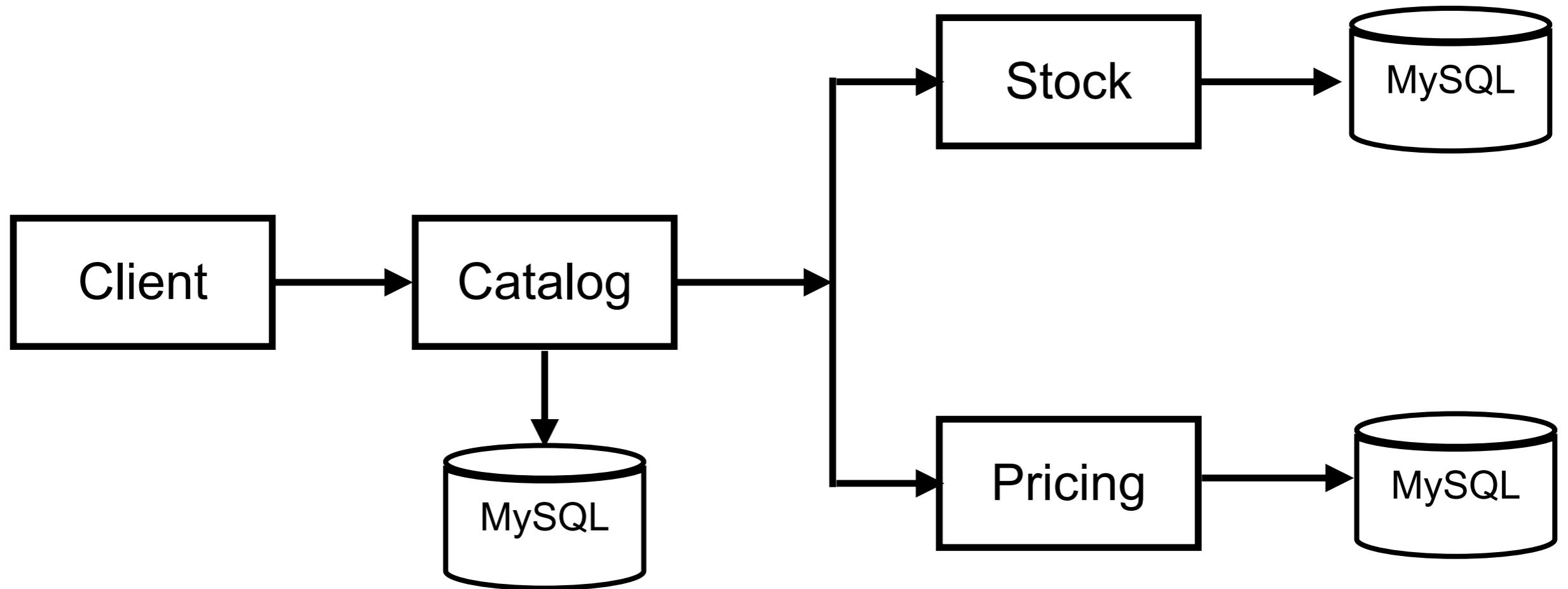
Path to Production ?

Improve your deployment/delivery pipeline





How to design pipeline ?



CI/CD pipeline



Create pipeline workshop



Jenkins

<https://github.com/up1/workshop-develop-microservices-2023>



Q/A

