fsocket

Future Reactive Application Protocol

info@netifi.com ■ www.netifi.com

netifi

Geekbang. 极客邦科技 InfoQ

极客邦科技
会议推荐2019

5月

QCon 北京
全球软件开发大会
大会：5月6–8日
培训：5月9–10日

QCon 广州
全球软件开发大会
培训：5月25–26日
大会：5月27–28日

6月

GTLC GLOBAL TECH LEADERSHIP CONFERENCE 上海
技术领导力峰会
时间：6月14–15日

GMTC 北京
全球大前端技术大会
大会：6月20–21日
培训：6月22–23日

7月

ArchSummit 深圳
全球架构师峰会
大会：7月12–13日
培训：7月14–15日

10月

QCon 上海
全球软件开发大会
大会：10月17–19日
培训：10月20–21日

11月

GMTC 深圳
全球大前端技术大会
大会：11月8–9日
培训：11月10–11日

AiCon 北京
全球人工智能与机器学习大会
大会：11月21–22日
培训：11月23–24日

12月

ArchSummit 北京
全球架构师峰会
大会：12月6–7日
培训：12月8–9日

About Us

- Ryland Degnan
  - Co-founder and CTO of Netifi
  - Prior to that, was a member of the Netflix Edge Platform team that created RSocket, Hystrix and RxJava
  - MSc Software Engineering from Oxford and BA Computer Science from Harvard

- Oleh Dokuka
  - Software Engineer focused on distributed systems development adopting Reactive Manifesto and Reactive Programming techniques
  - Open source geek, active contributor of Project Reactor
  - Author of the book "Reactive Programming in Spring 5"

netifi

Today's Outline

Microservice Challenges

Development and operational challenges when developing and deploying microservice architectures

Introducing RSocket

Open source communication protocol developed by Netifi in collaboration with Pivotal, Facebook, Netflix, Alibaba, etc.

Future of Microservices

How Netifi uses RSocket to deliver a fully featured, enterprise-ready, cloud-native application platform

netifi

Microservices are Growing at Record Pace

Record Growth of Microservices

91% are using or have plans to use microservices

Expansion to New Environments

88% are using multiple cloud environments

New Operational Challenges

99% report challenges in using microservices

Source: Business Insider - Global Microservices Trends Report - May 2018

netifi

Enterprises Have Strong Requirements

- Easy to deploy and operate
- Fast time-to-market
- Centralized security and access control
- Performant and scalable
- Low cost of ownership
- Works natively across multiple environments
- Real-time and streaming interactions

netifi

Old Technology is Not Sufficient

Microservices

Microservices are inherently complex and unpredictable in their performance characteristics

+

Old Techniques and Technology

Applying same techniques and technologies that were used before simply won't work

=

Dev and Ops Challenges

This has led to development and operational challenges when developing and deploying microservice architectures

netifi

Microservice Challenges

- Lyft
  - Spending $300,000,000+ over the next 3 years
  - “It may become increasingly difficult to maintain and improve our performance”

- Pinterest
  - Spending $440,000,000+ over the next 4 years
  - “Pinterest is spending upwards of 20 percent of its total revenue on cloud”

- Snapchat
  - Spending $500,000,000+ per year
  - “Google now has the company in handcuffs, there’s little Snap can do to change that without having to invest a tremendous amount of money”

netifi

Microservice Challenges

Google Cloud

netifi

Microservice Challenges

Google Cloud

Infrastructure

etcd | kube-apiserver | kube-scheduler | kube-controller | cloud-controller | Certificate authority

netifi

Microservice Challenges

Google Cloud

Nodes

kube-proxy | kubelet | kube-dns | dashboard | Network CNI | Container manager

Infrastructure

etcd | kube-apiserver | kube-scheduler | kube-controller | cloud-controller | Certificate authority

Kubernetes

netifi

Microservice Challenges

Google Cloud

Control

Pod galley

Pod mixer

Pod pilot

Pod security

Pod prometheus

Pod gateways

Nodes

kube-proxy

kubelet

kube-dns

dashboard

Network CNI

Container manager

Infrastructure

etcd

kube-apiserver

kube-scheduler

kube-controller

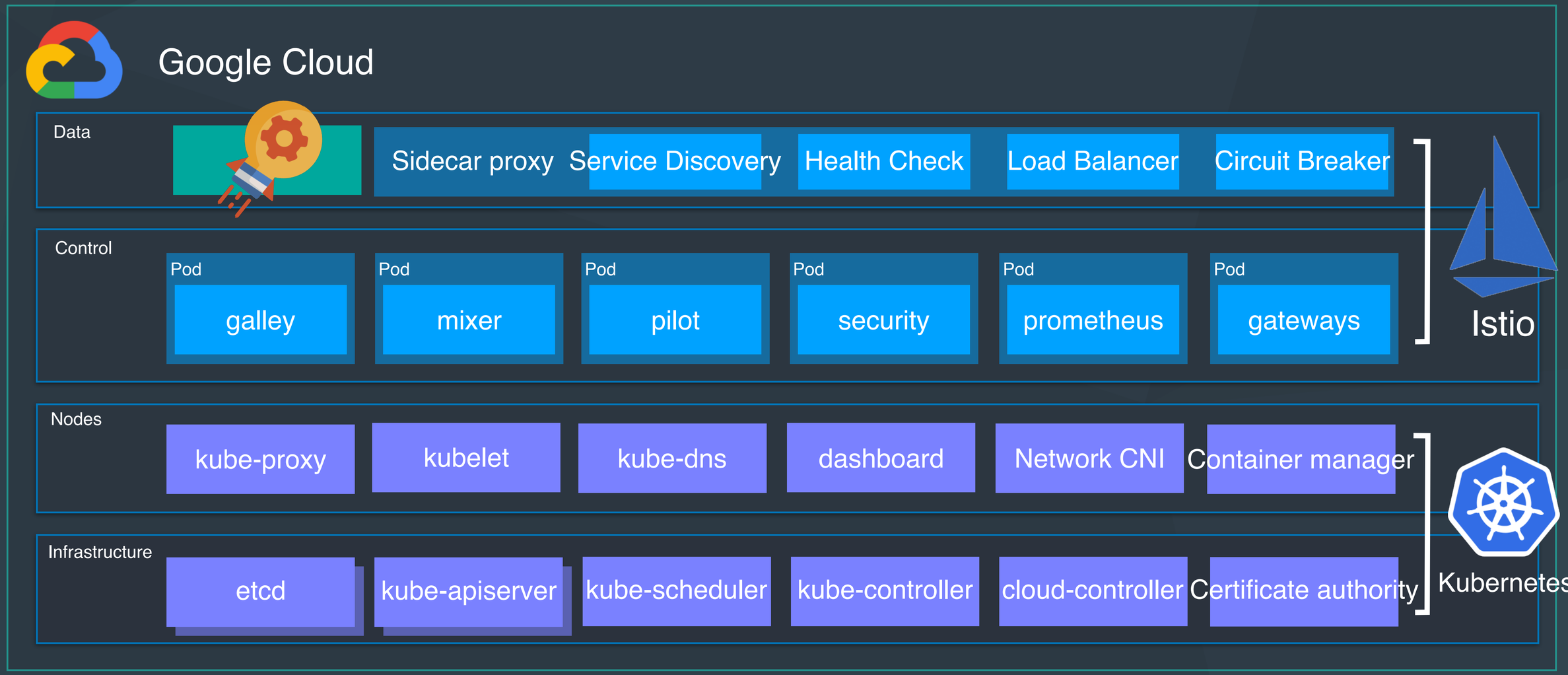
cloud-controller

Certificate authority

Kubernetes

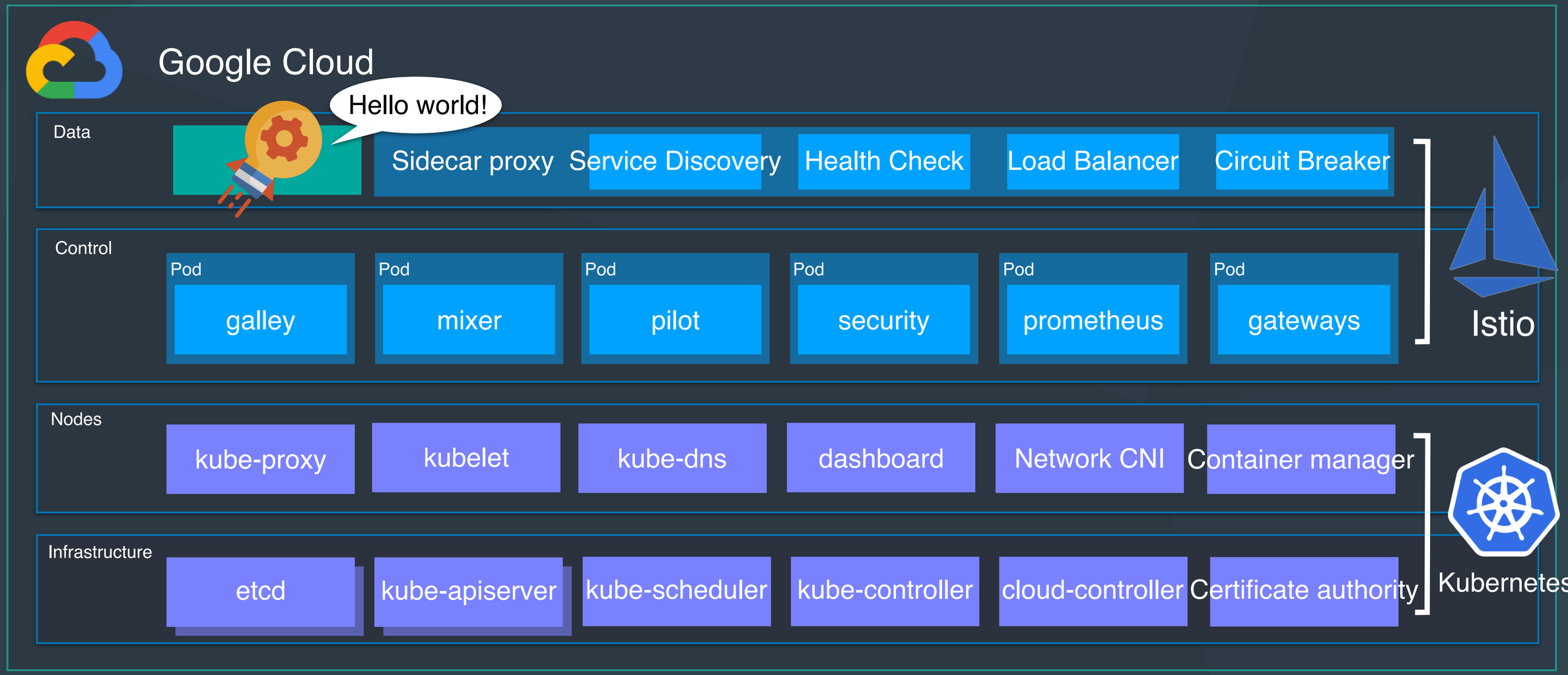
netifi

Microservice Challenges



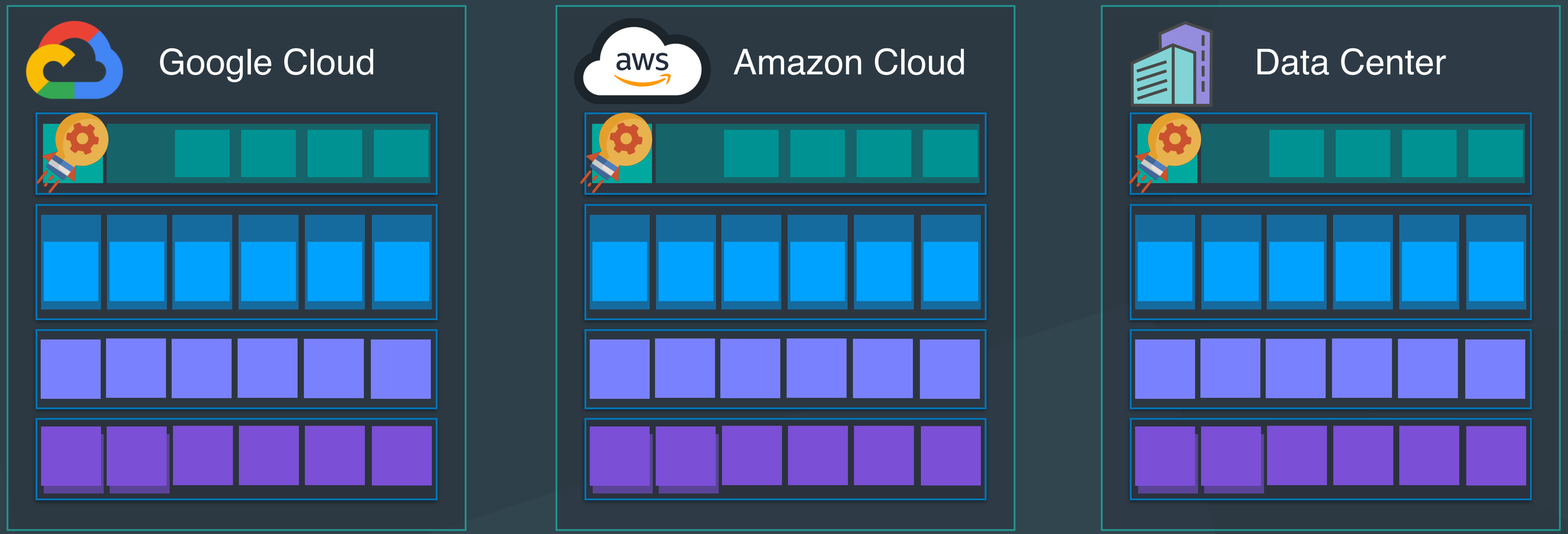
netifi

Microservice Challenges



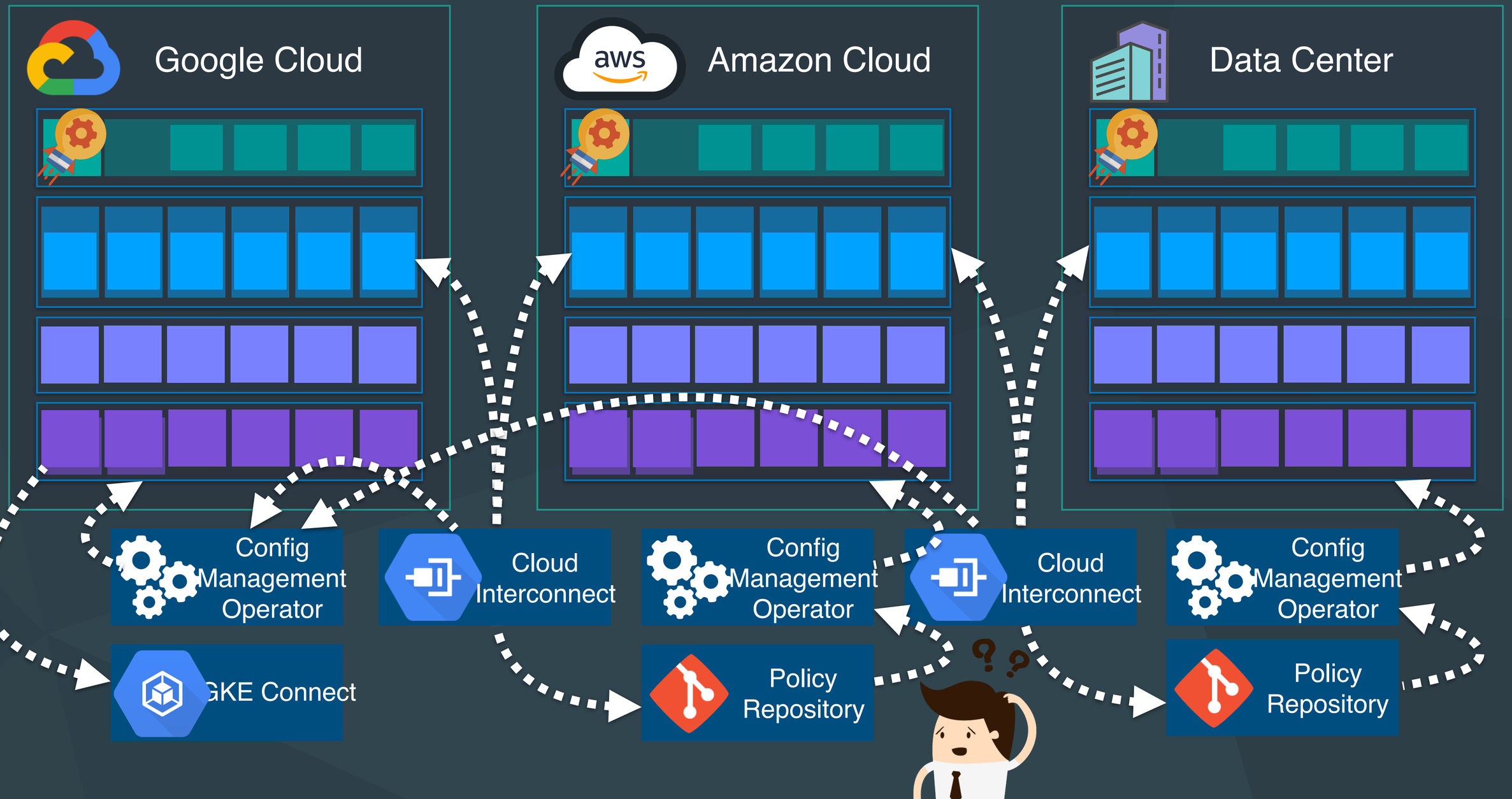
netifi

The Multi-Cloud Problem is Still Not Solved!



netifi

The Multi-Cloud Problem is Still Not Solved!



New Challenges for Everyone

- Developers
  - Complicated to use
  - Too much configuration
  - Too much code complexity
  - Can’t develop locally

- Ops/security engineers
  - Too many services
  - Ports open everywhere
  - Security vulnerabilities
  - Complex APIs

- Managers and architects
  - Too expensive
  - Requires a large team
  - Doesn’t scale well
  - Performance bottlenecks

netifi

Not Enterprise-Ready

- ~~Easy to deploy and operate~~
- ~~Fast time-to-market~~
- ~~Centralized security and access control~~
- ~~Performant and scalable~~
- ~~Low cost of ownership~~
- ~~Works natively across multiple environments~~
- ~~Real-time and streaming interactions~~

netifi

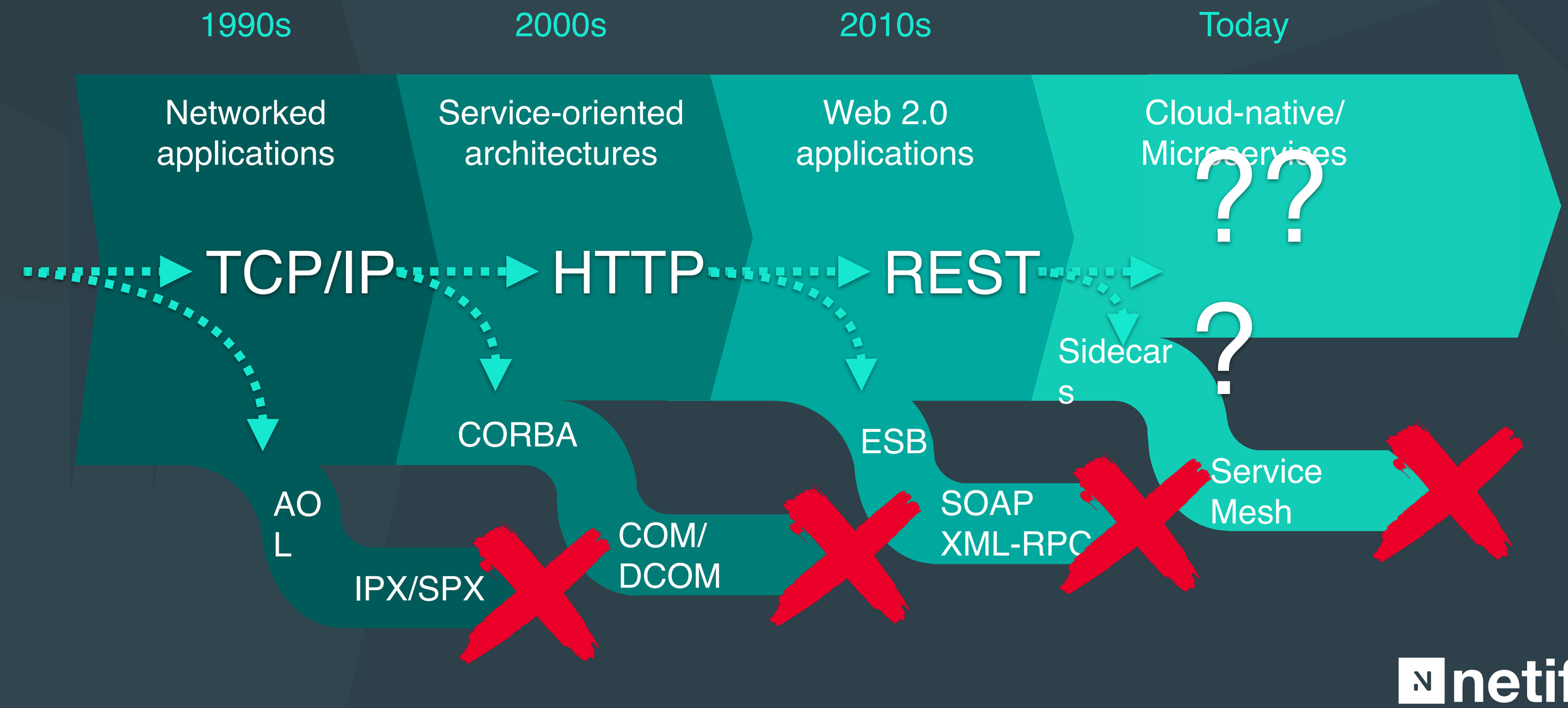
Introducing Netifi

Introducing Netifi

+

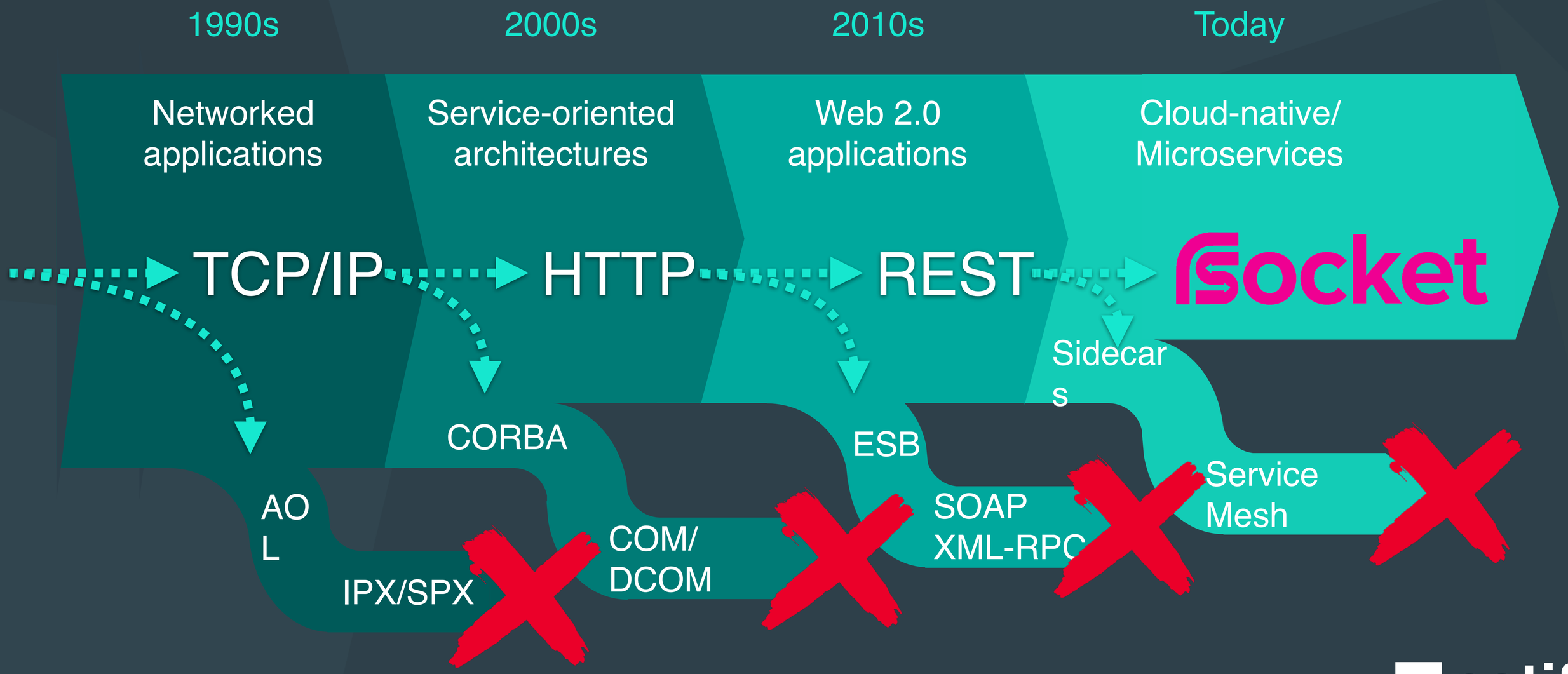
Enterprise

Real Innovation Is Needed



netifi

Real Innovation Is Needed



netifi

RSocket is a Breakthrough Technology

- Open Source Layer 5/6 communication protocol
- Reactive streams semantics
- Application-level flow control
- Supports both RPC and event-based messaging
- Up to 10x faster than HTTP, using 90% less resources

Built by leaders in microservices and cloud computing

Introducing RSocket

- RSocket is a bi-directional, multiplexed, message-based, binary protocol based on Reactive Streams (pull-push) flow control

- It provides out of the box support for four interaction models commonly seen in cross-application communication
  - Request-Response
  - Fire-and-Forget
  - Request-Stream
  - Channel

Message-Driven Binary Protocol

- Requester-Responder interaction is broken down into frames that encapsulate messages

- The framing is binary (not human readable like JSON or XML)
  - Massive efficiencies for machine-to-machine communication
  - Downsides only manifest rarely and can be mitigated with tooling

- Payload Agnostic
  - Protobuf, JSON, Custom Binary

Multiplexed

- Connections that are only used for a single request are massively inefficient (HTTP 1.0)
- Pipelining (ordering requests and responses sequentially) is a naive attempt solving the issue, but results in head-of-line blocking (HTTP 1.1)
- Multiplexing solves the issue by annotating each message on the connection with a stream id that partitions the connection into multiple "logical streams"

Bi-Directional

- Many protocols (notably not TCP) have a distinction between the client and server for the lifetime of a connection
- This division means that one side of the connection must initiate all requests, and the other side must initiate all responses
- Even more flexible protocols like HTTP/2 do not fully drop the distinction
  - Servers cannot start an unrequested stream of data to the client
- Once a client initiates a connection to a server, both parties can be requestors or responders to a logical stream
- Transport Agnostic
  - TCP, Websockets, HTTP/2, Aeron (UDP), Shared Memory

Reactive Streams Flow Control

- Network protocols generally send a single request, and receive an arbitrarily large response in return
- There is nothing to stop the responder (or even the requestor) from sending an arbitrarily large amount of data and overwhelming the receiver
- In cases where TCP flow control throttles the responder, queues fill with large amounts of un-transferred data
- Reactive Streams (pull-push) flow control ensures that data is only materialized and transferred when receiver is ready to process it

Reactive Streams Flow Control

HTTP

Socket

netifi

Reactive Streams Flow Control

HTTP

Socket

netifi

Advanced Features

- Cancellation
  - All streams (including request/response) support cancellation to allow efficient cleanup of server (responder) resources.
- Resumability
  - RSocket supports session resumption, allowing a simple handshake to resume a client/server session over a new transport connection.
- Leasing
  - Responder (typically a server) can issue leases to the requester based upon its knowledge of its capacity in order to control requests rates.

netifi

Introducing Netifi

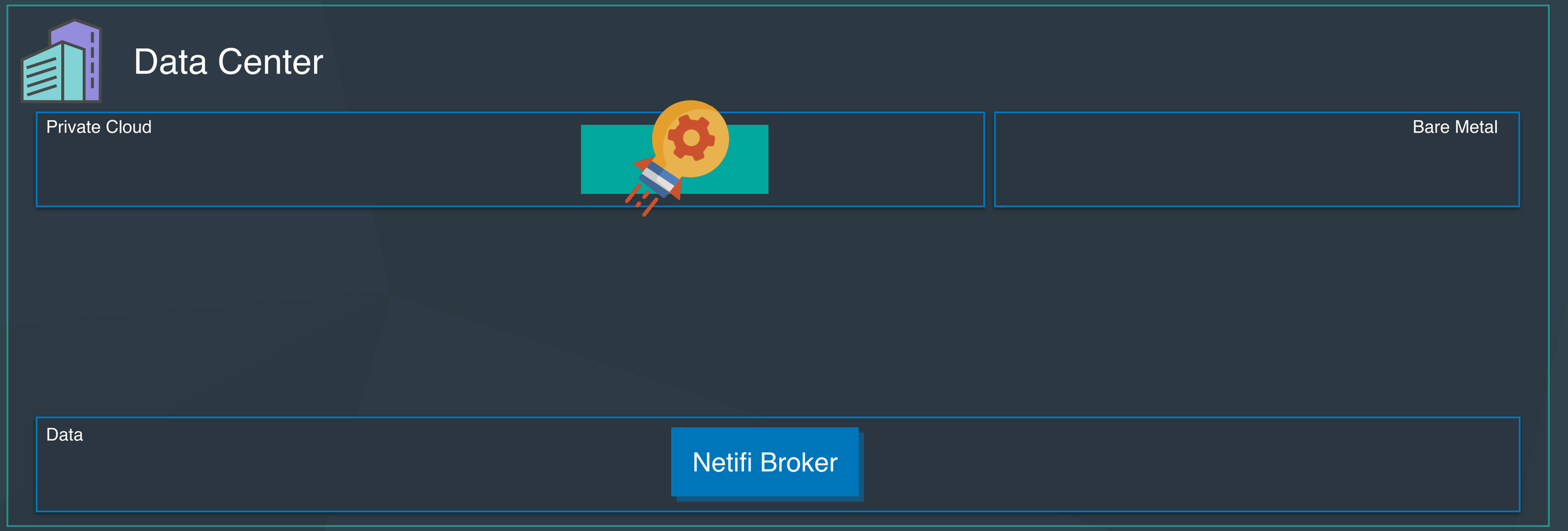
Introducing Netifi

Complete Microservices Platform

- Easy to deploy and operate
- Fast time-to-market
- Centralized security and access control
- Performant and scalable
- Low cost of ownership
- Works natively across multiple environments
- Real-time and streaming interactions

netifi

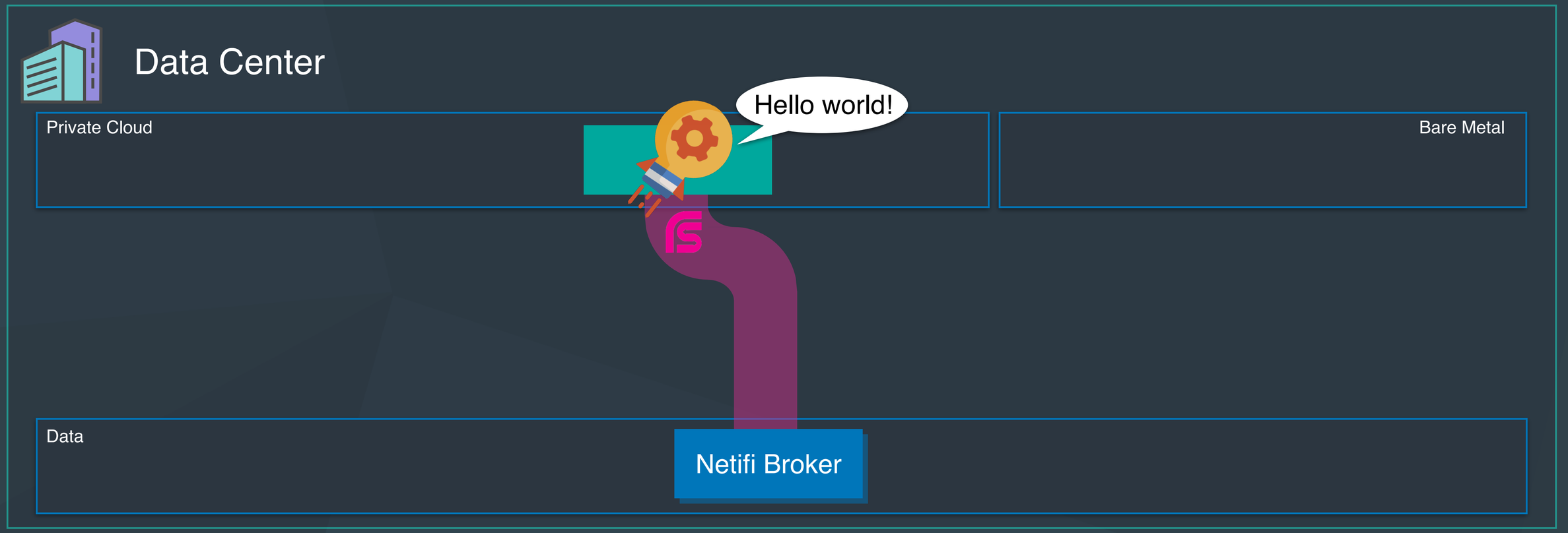
Microservices With Netifi



- Just a single deployable that can be scaled up and down

netifi

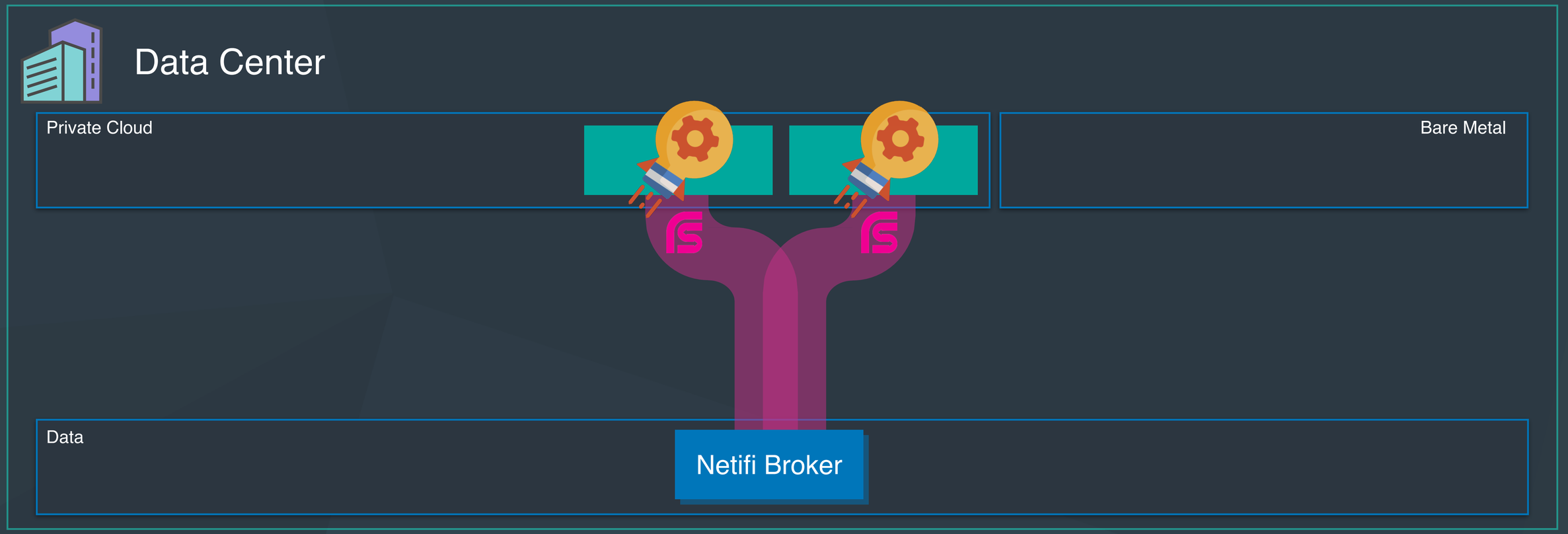
Microservices With Netifi



- Just a single deployable that can be scaled up and down

netifi

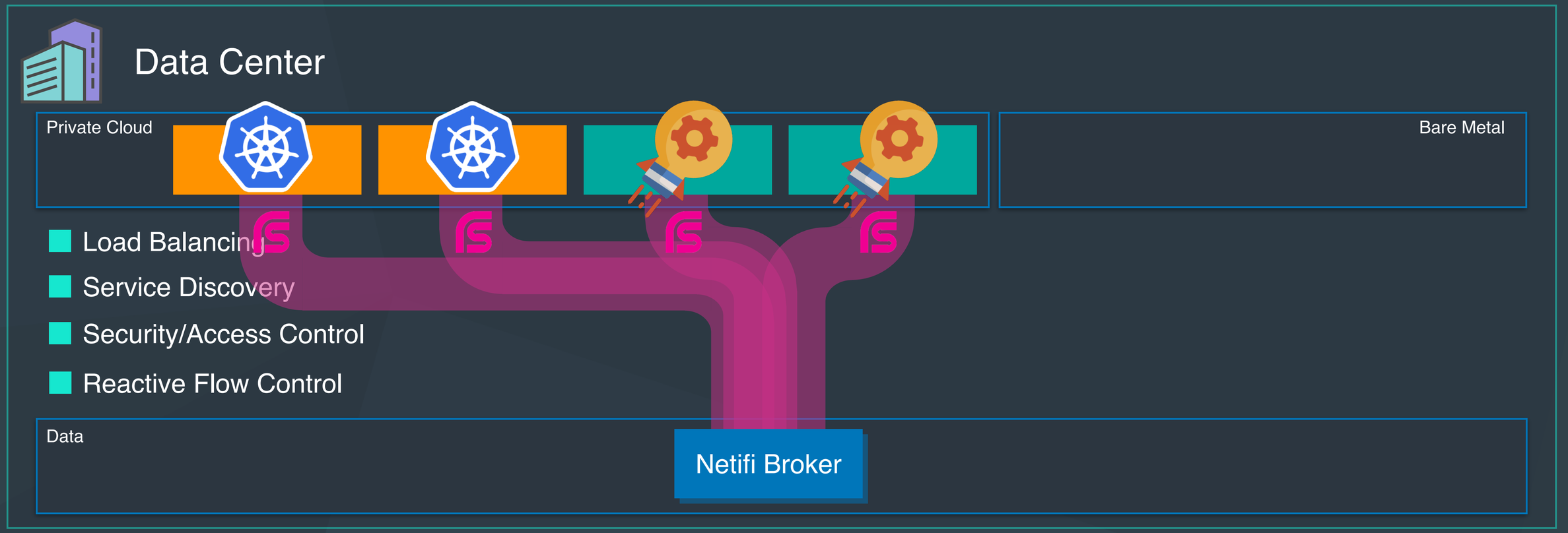
Microservices With Netifi



- Just a single deployable that can be scaled up and down

netifi

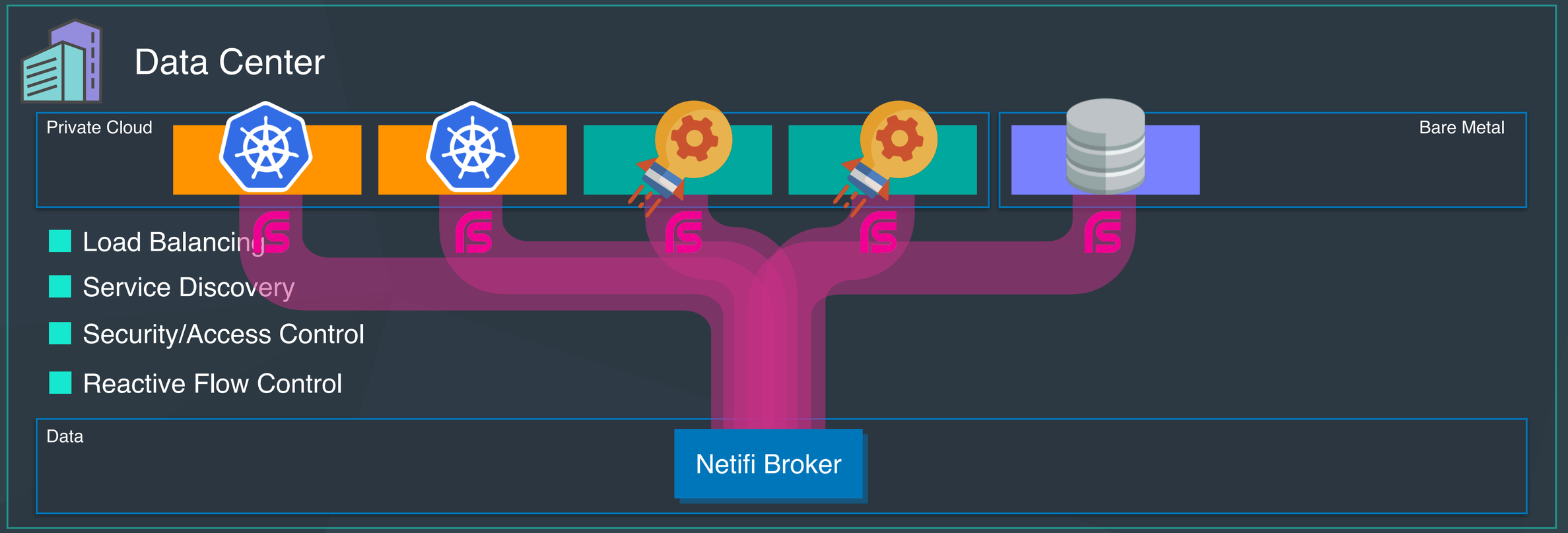
Microservices With Netifi



- Just a single deployable that can be scaled up and down
- Takes complexity out of your infrastructure, handles it in the network

netifi

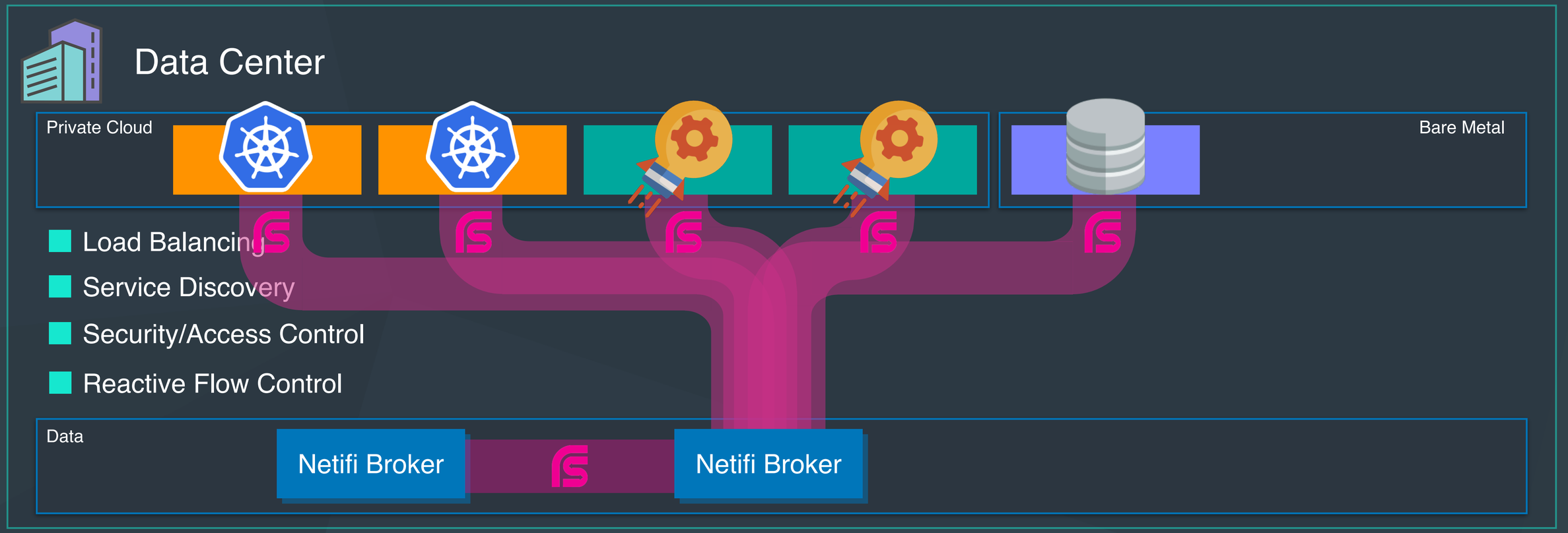
Microservices With Netifi



- Just a **single deployable** that can be scaled up and down
- Takes complexity out of your **infrastructure**, handles it in the **network**
- Runs **anywhere**: bridge your **datacenter** with public and private **cloud**

netifi

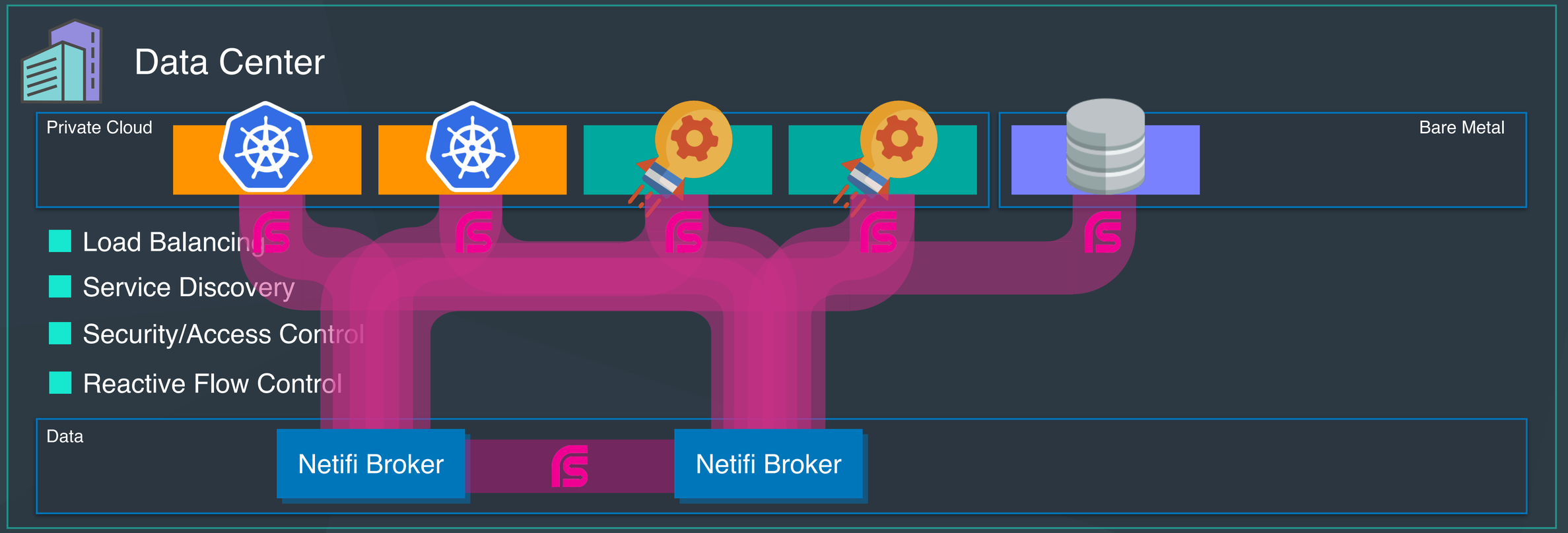
Microservices With Netifi



- Just a single deployable that can be scaled up and down
- Takes complexity out of your infrastructure, handles it in the network
- Runs anywhere: bridge your datacenter with public and private cloud

netifi

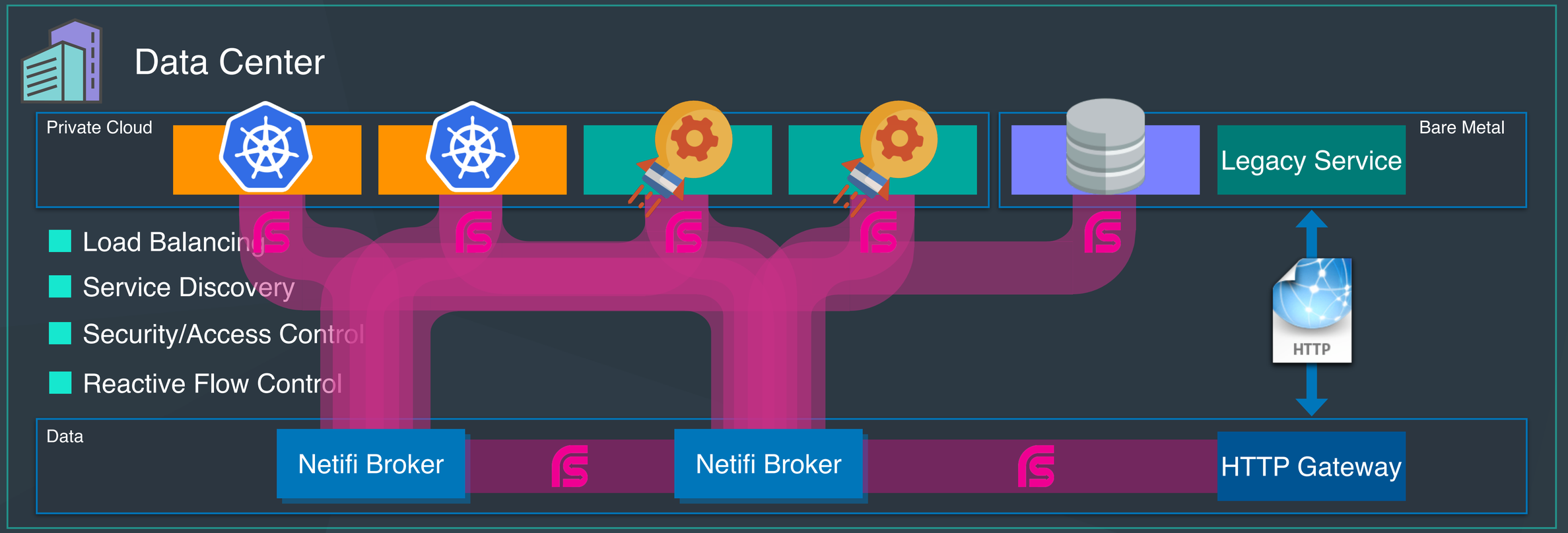
Microservices With Netifi



- Just a single deployable that can be scaled up and down
- Takes complexity out of your infrastructure, handles it in the network
- Runs anywhere: bridge your datacenter with public and private cloud

netifi

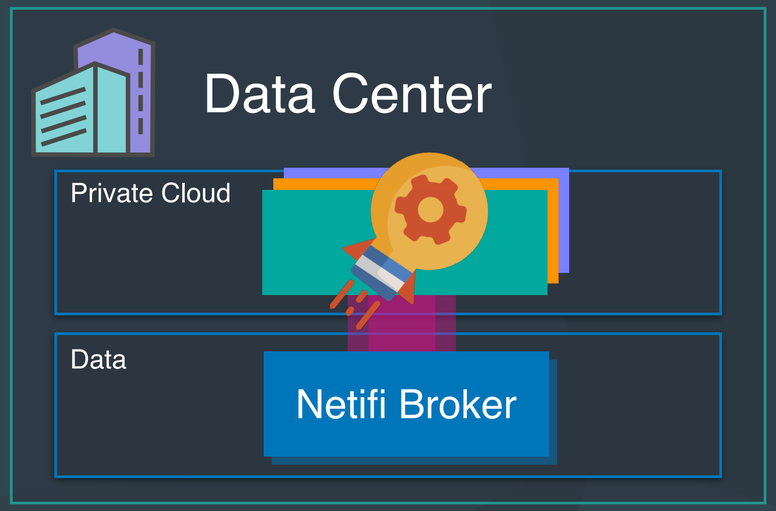
Microservices With Netifi



- Just a **single deployable** that can be scaled up and down
- Takes complexity out of your **infrastructure**, handles it in the **network**
- Runs **anywhere**: bridge your **datacenter** with public and private **cloud**
- Enterprise **connectors** allow **legacy** services to migrate **seamlessly**

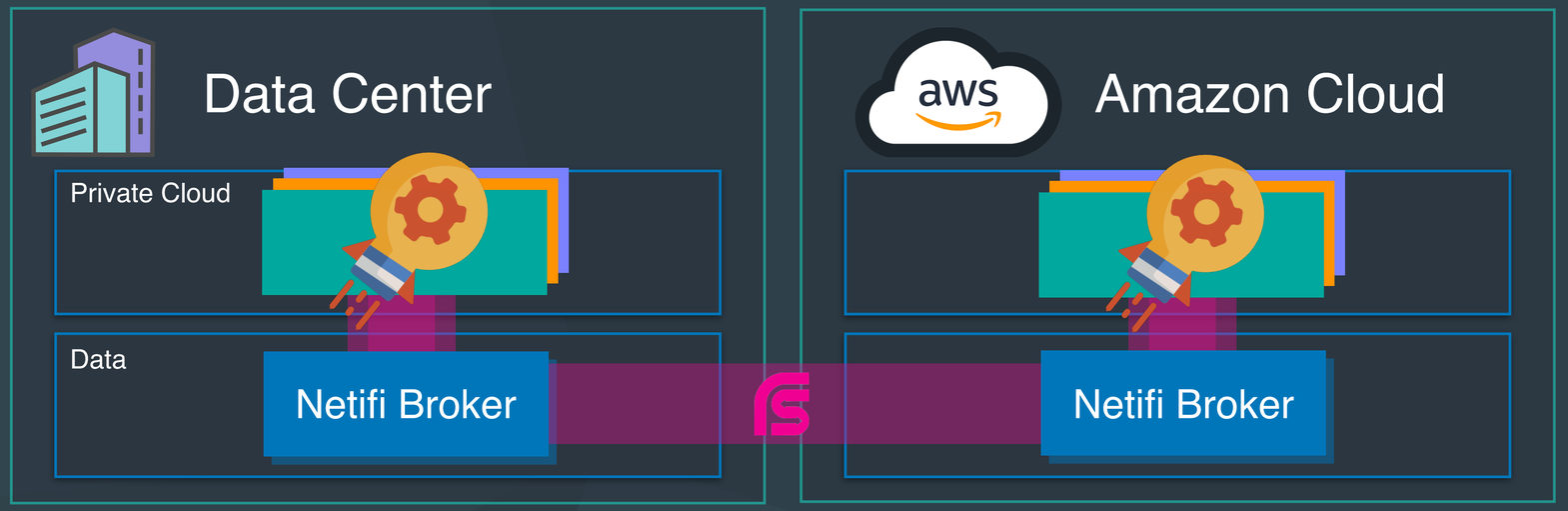
netifi

Microservices With Netifi



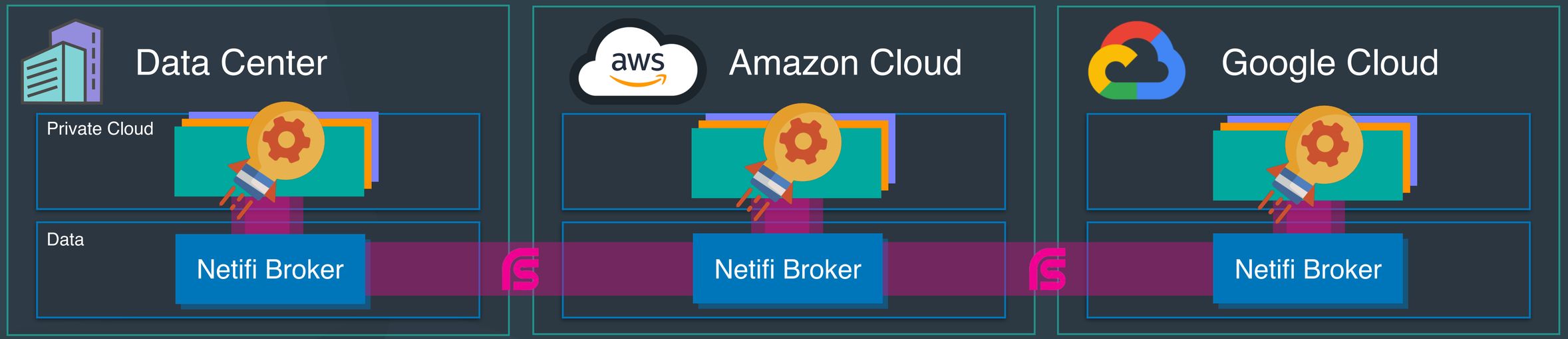
netifi

Microservices With Netifi



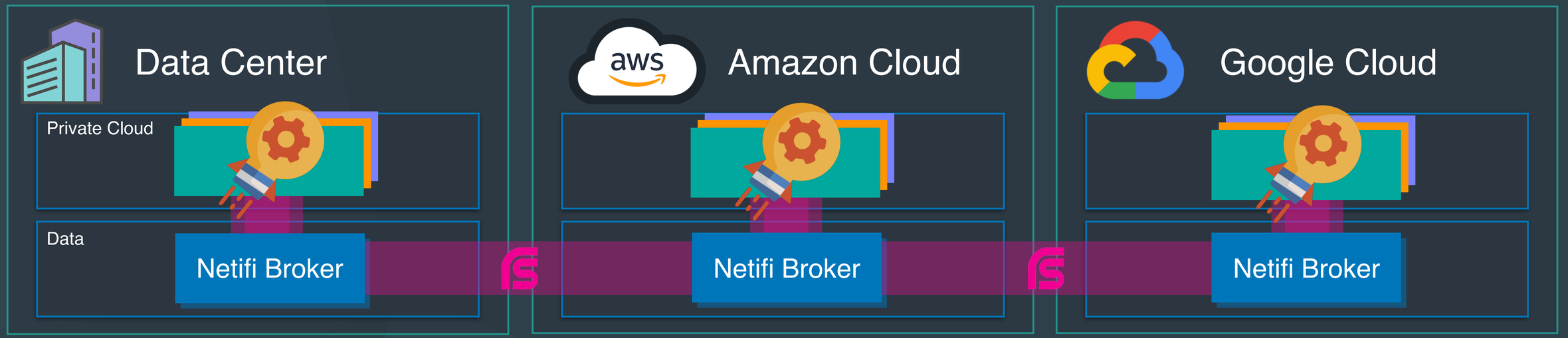
netifi

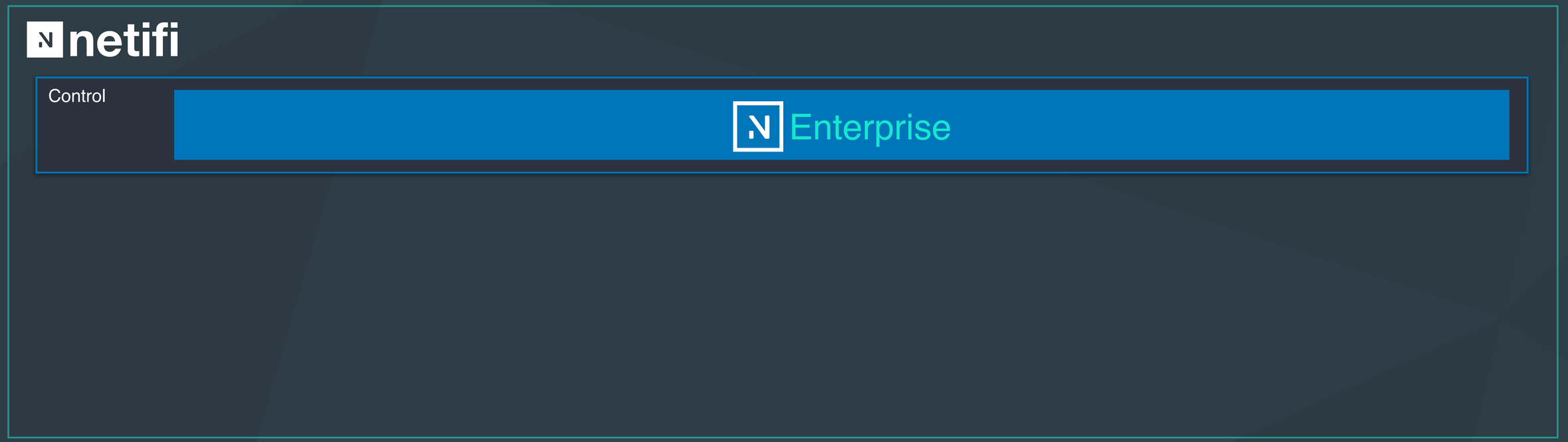
Microservices With Netifi



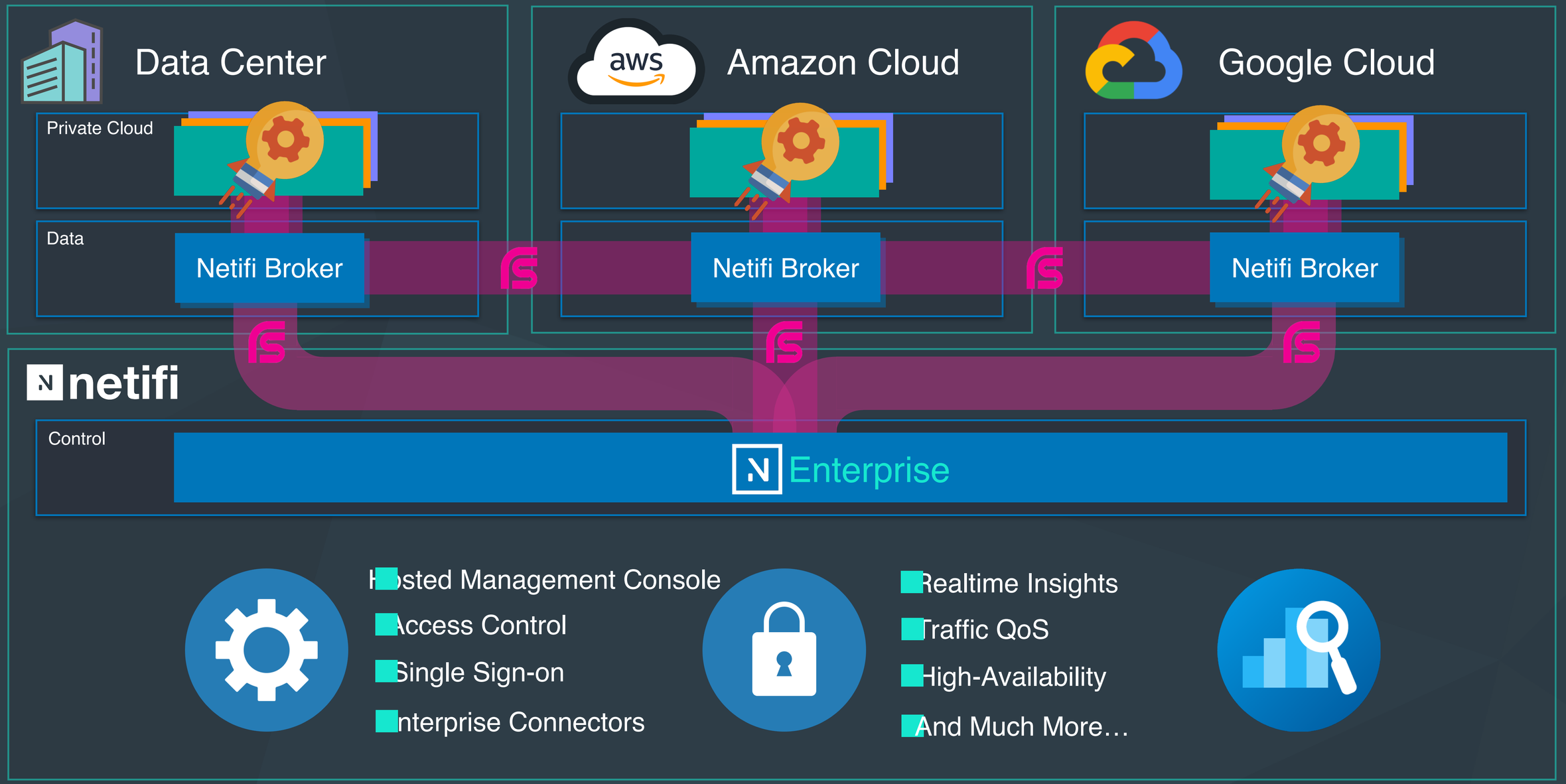
netifi

Microservices With Netifi

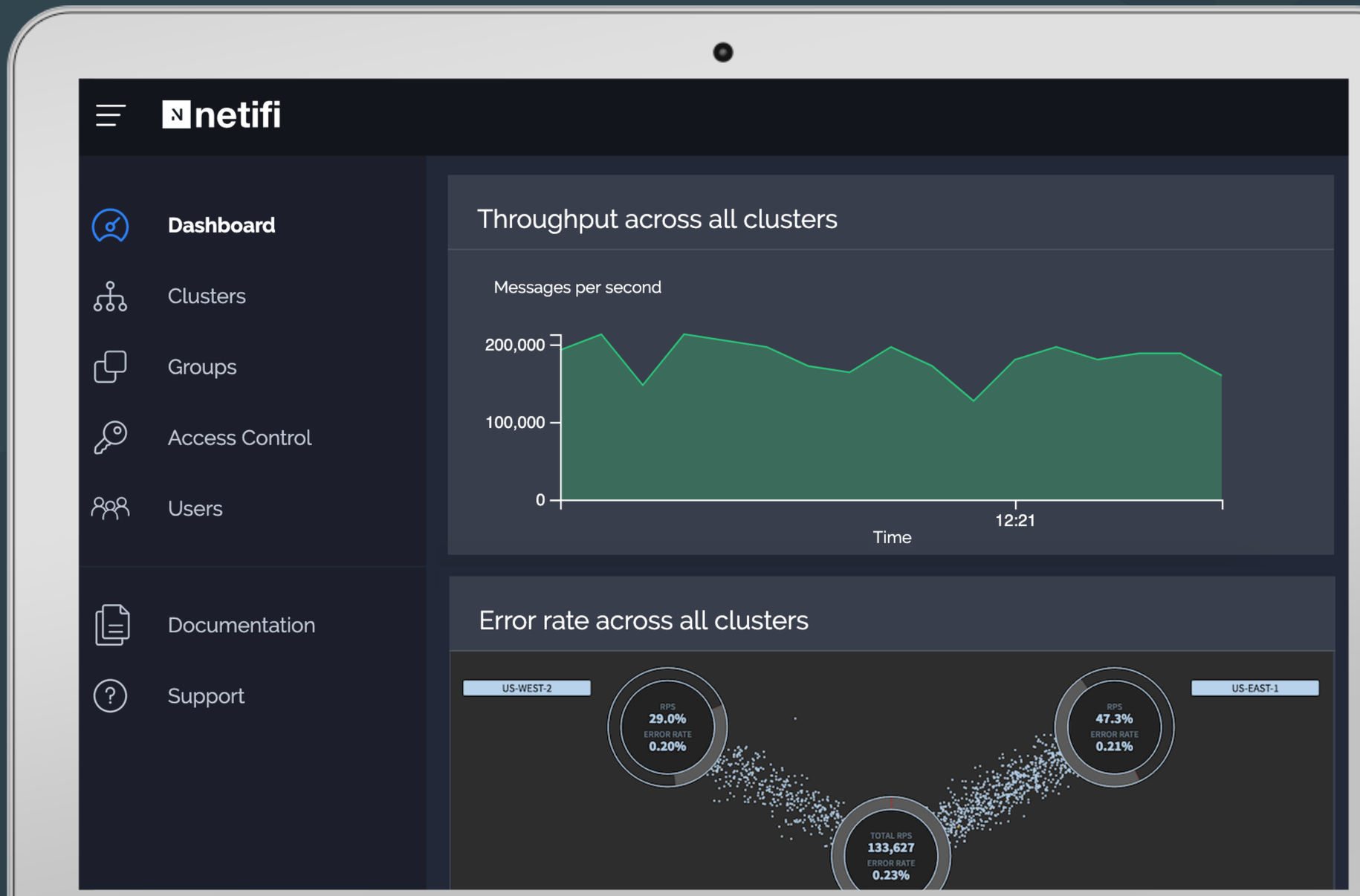




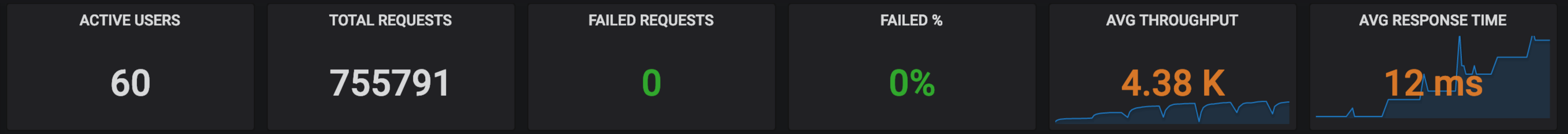
Microservices With Netifi

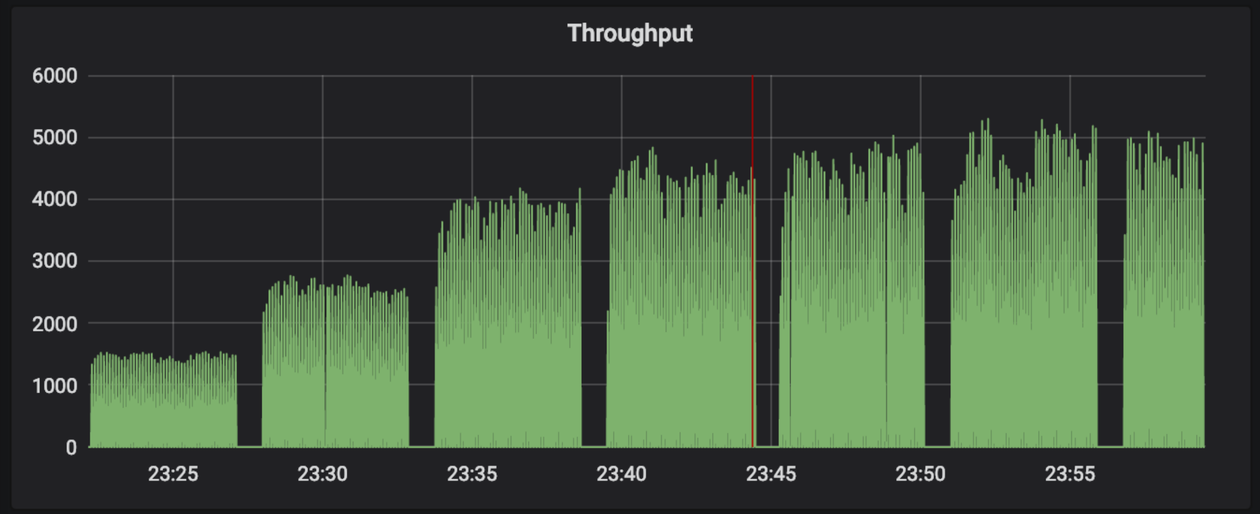


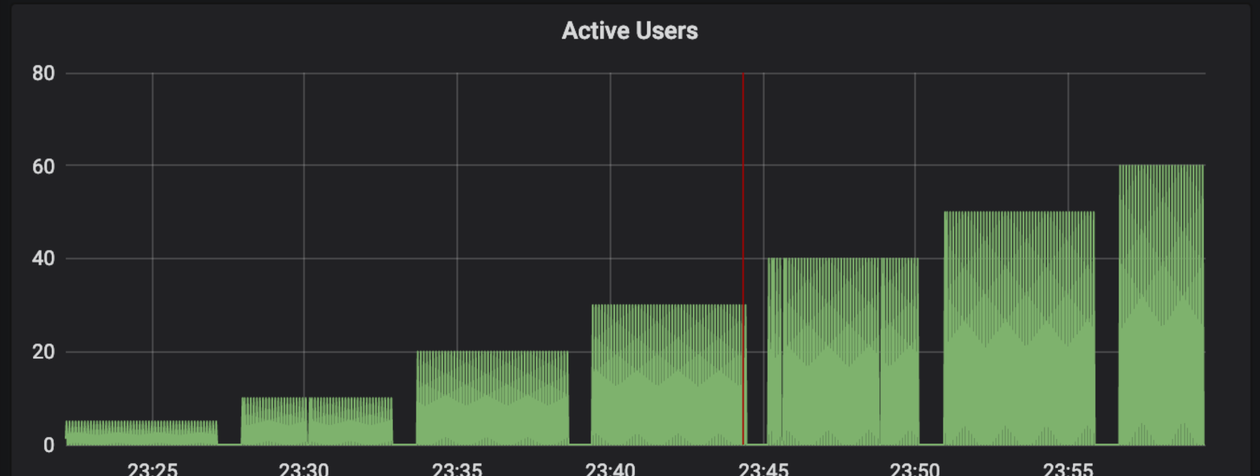
Microservices Operations In Real Time

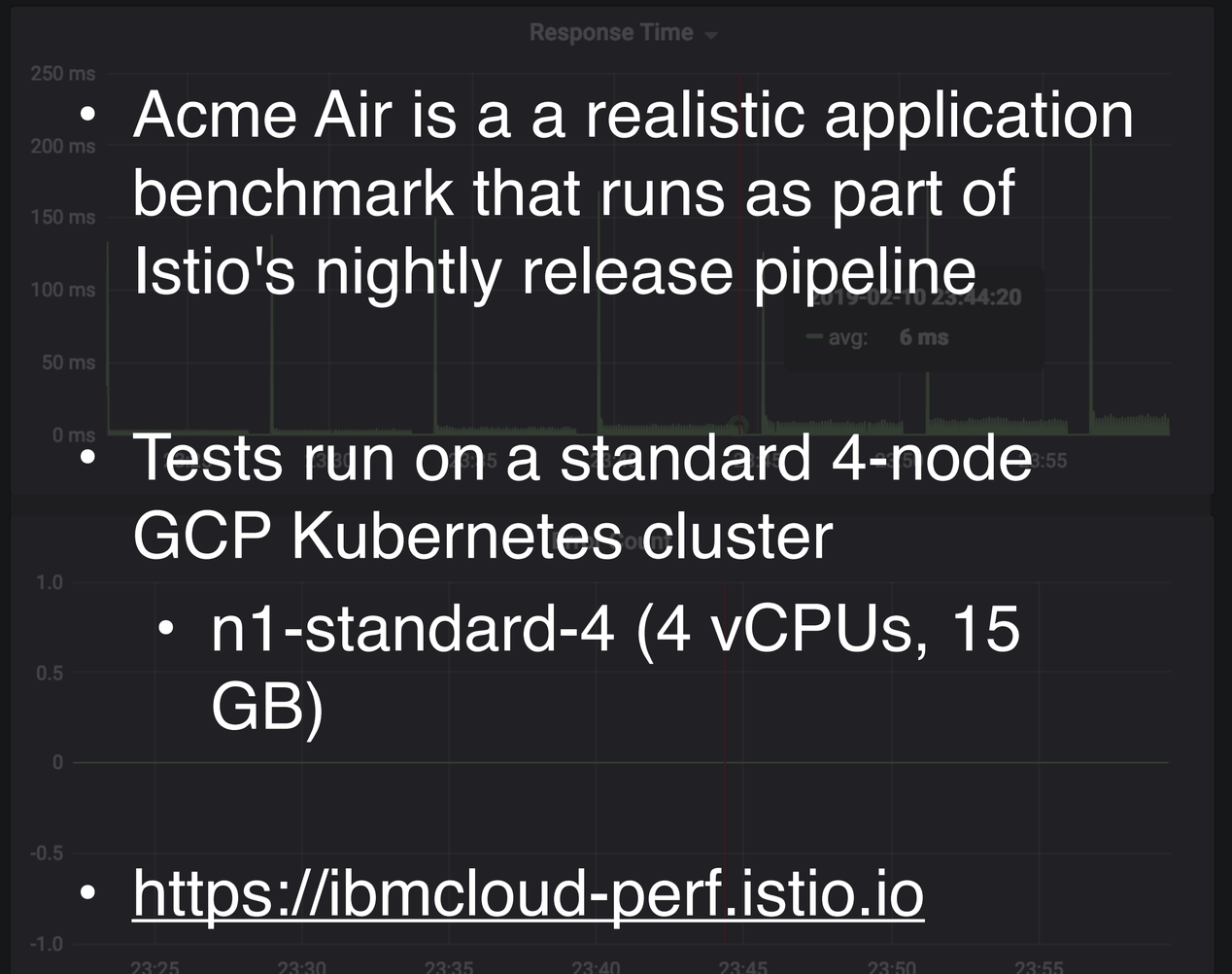


Performance Benchmark



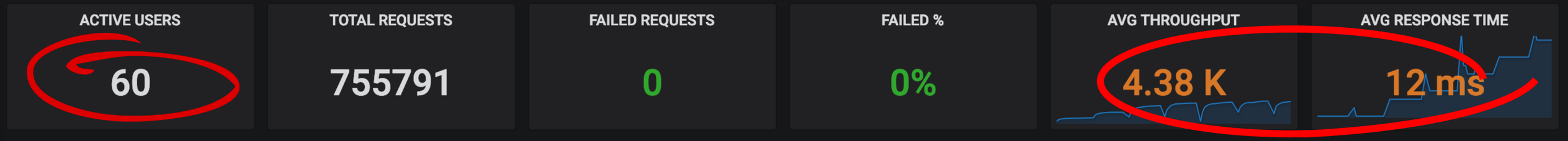


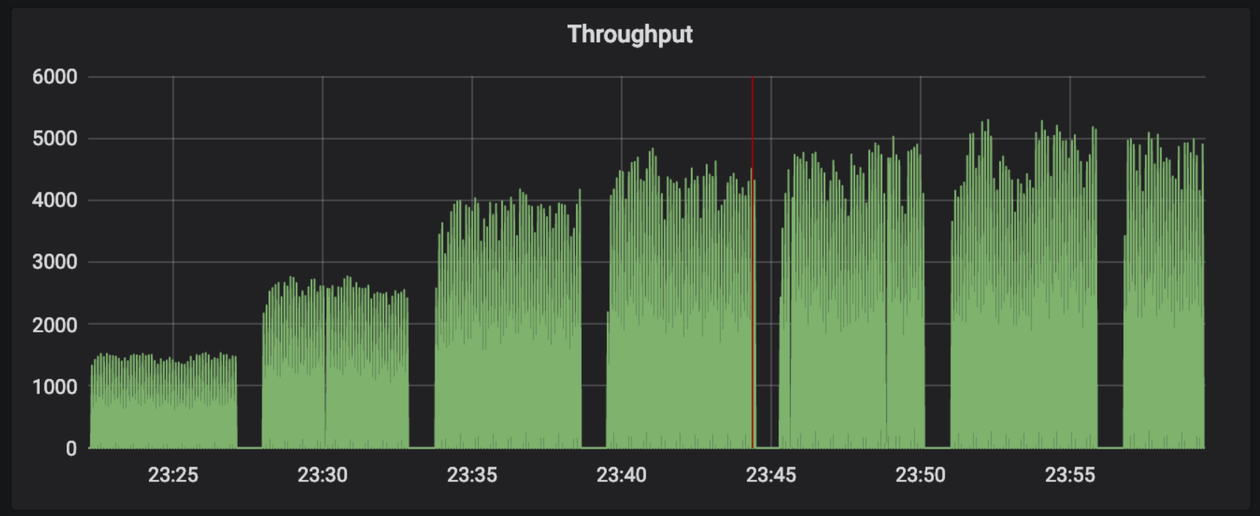


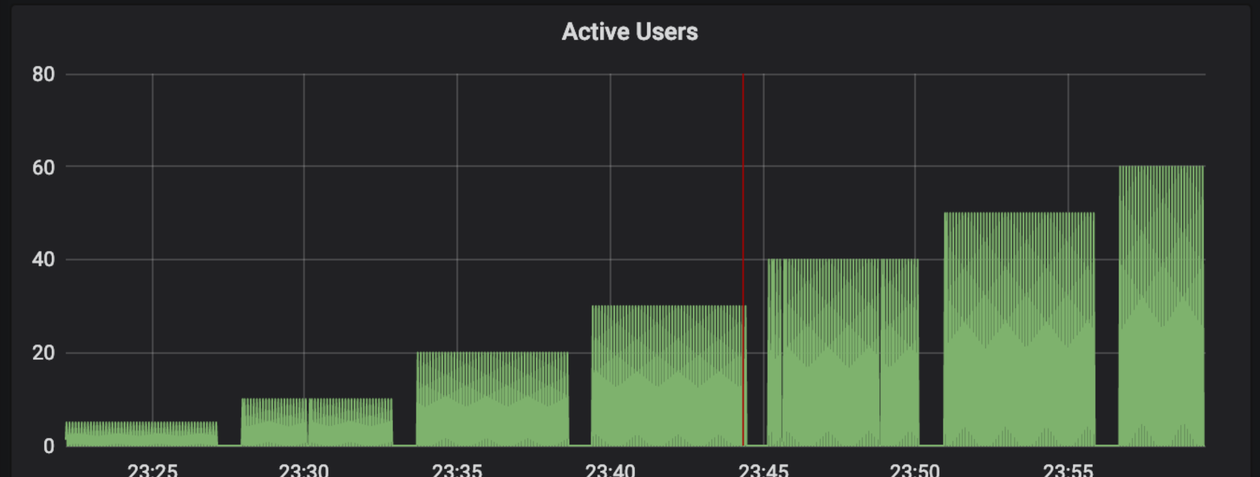


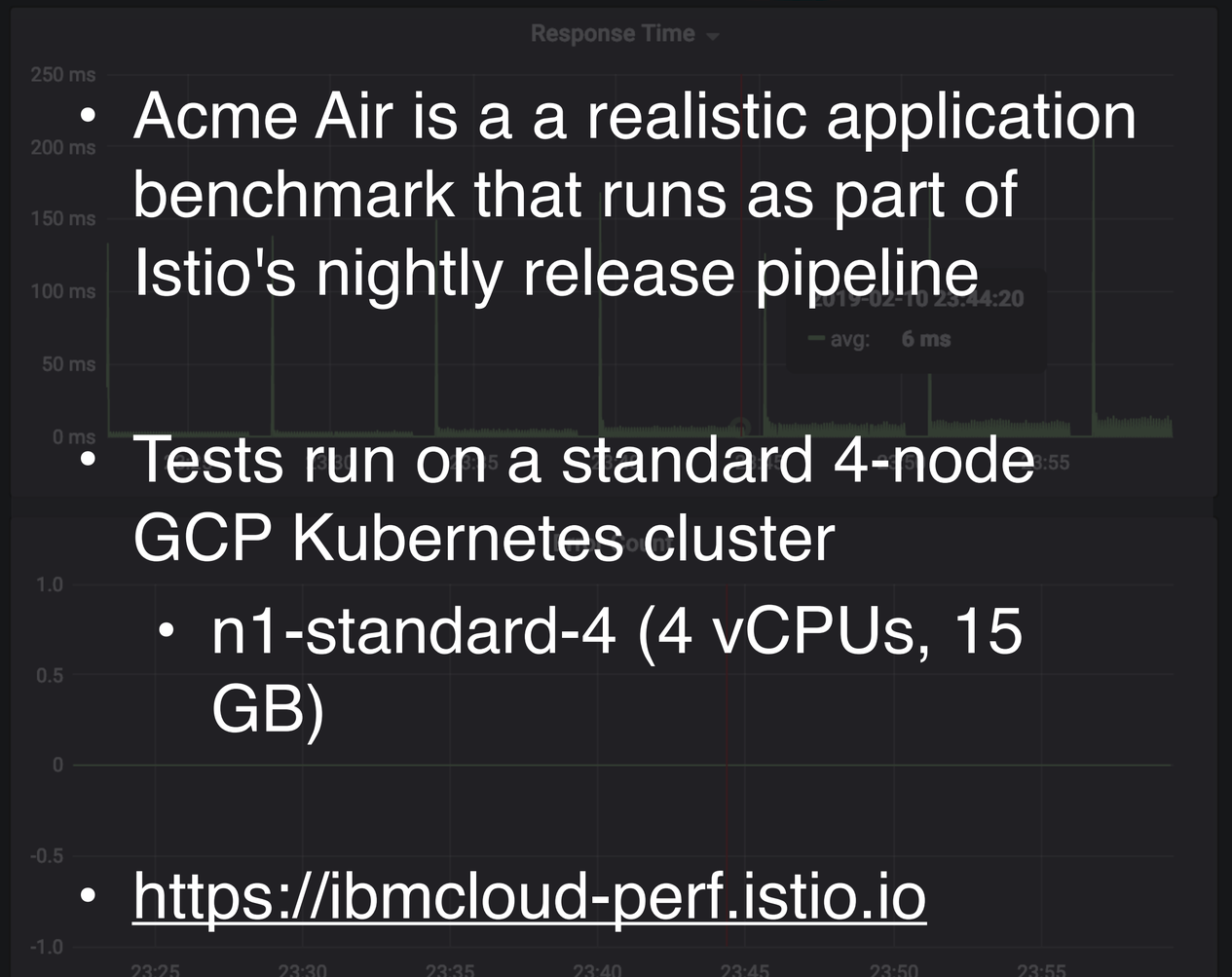
- Acme Air is a a realistic application benchmark that runs as part of Istio's nightly release pipeline
- Tests run on a standard 4-node GCP Kubernetes cluster
  - n1-standard-4 (4 vCPUs, 15 GB)
- https://ibmcloud-perf.istio.io

Performance Benchmark



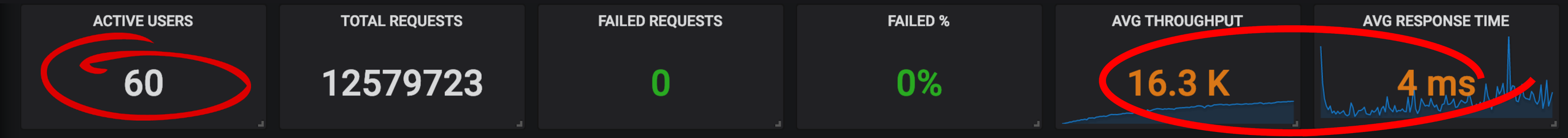


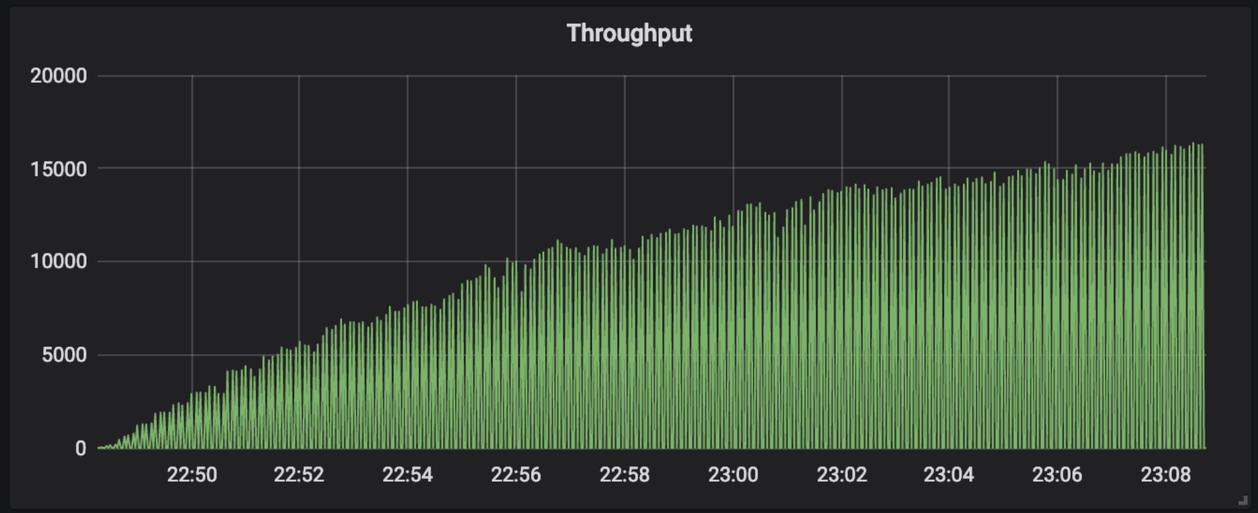


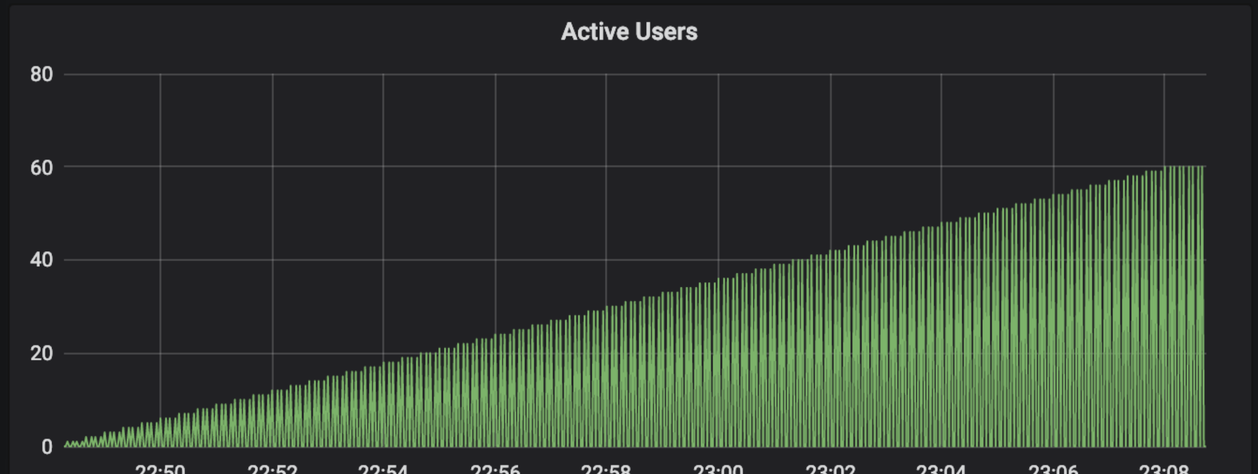


- Acme Air is a a realistic application benchmark that runs as part of Istio's nightly release pipeline
- Tests run on a standard 4-node GCP Kubernetes cluster
  - n1-standard-4 (4 vCPUs, 15 GB)
- https://ibmcloud-perf.istio.io

Netifi is 4x Faster



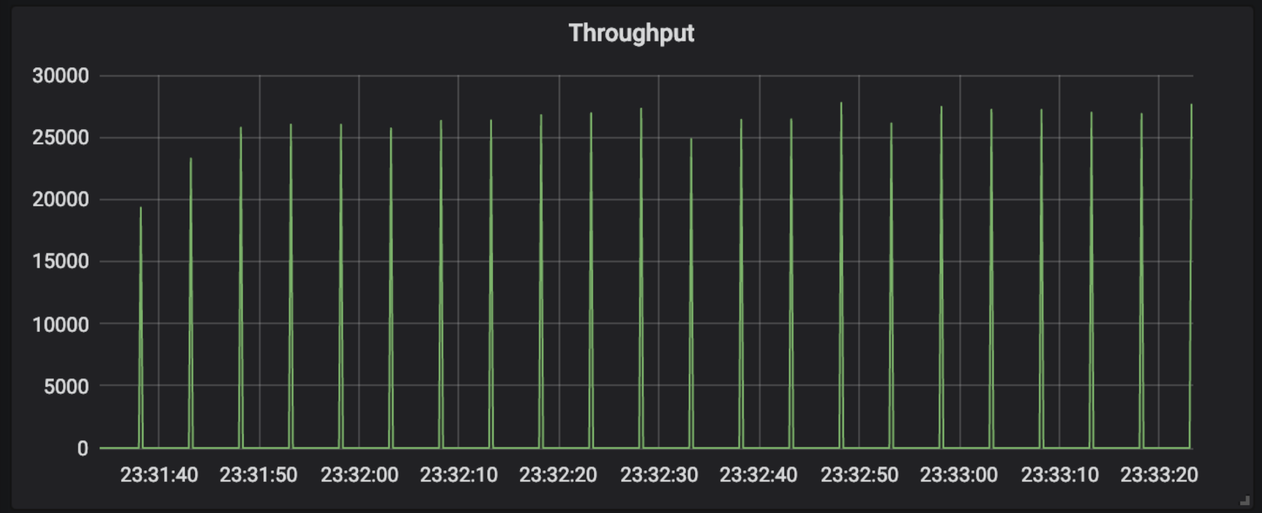


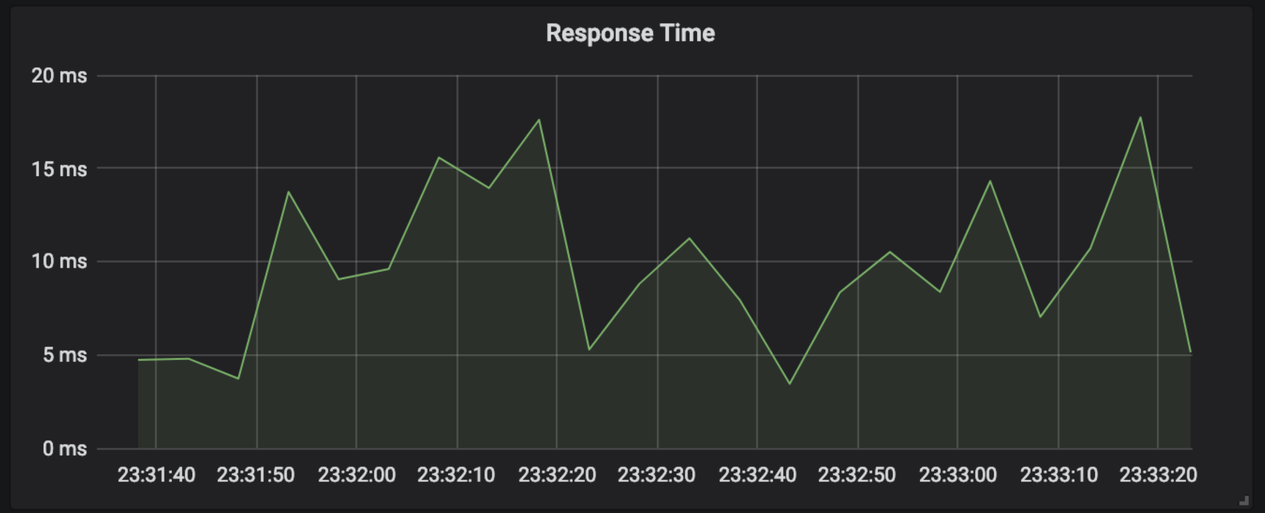


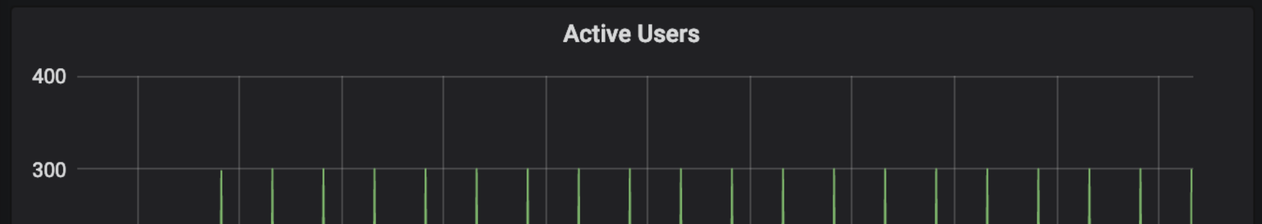
Netifi is 4x Faster

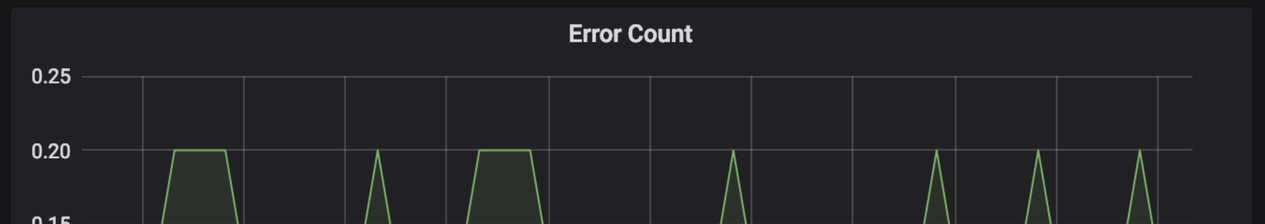
ACTIVE USERS	TOTAL REQUESTS	FAILED REQUESTS	FAILED %	AVG THROUGHPUT	AVG RESPONSE TIME
60	12579723	0	0%	16.3 K	4 ms

ACTIVE USERS	TOTAL REQUESTS	FAILED REQUESTS	FAILED %	AVG THROUGHPUT	AVG RESPONSE TIME
300	2873562	9	0.0003%	27.7 K	5.1 ms



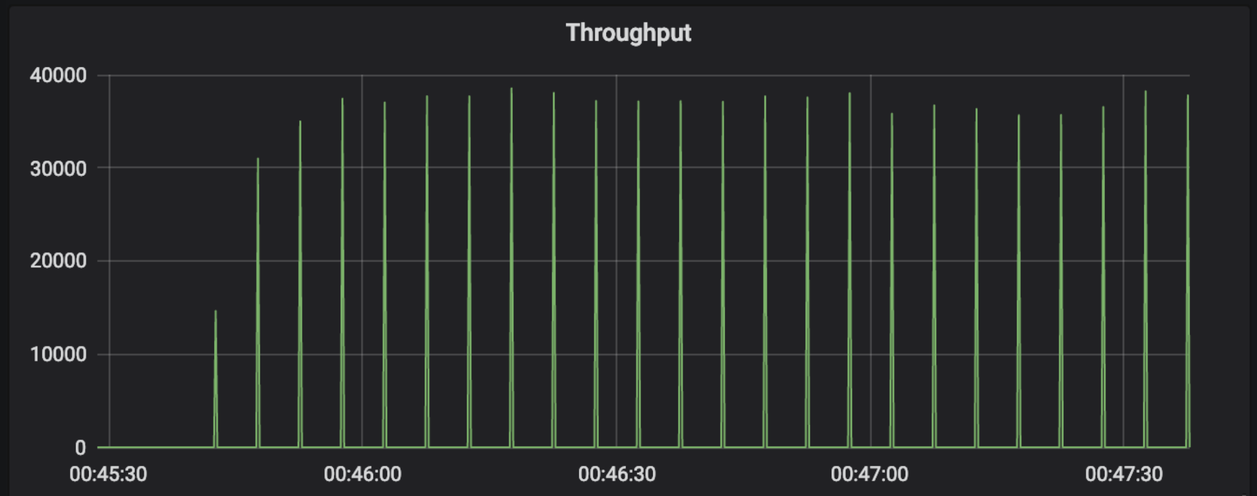


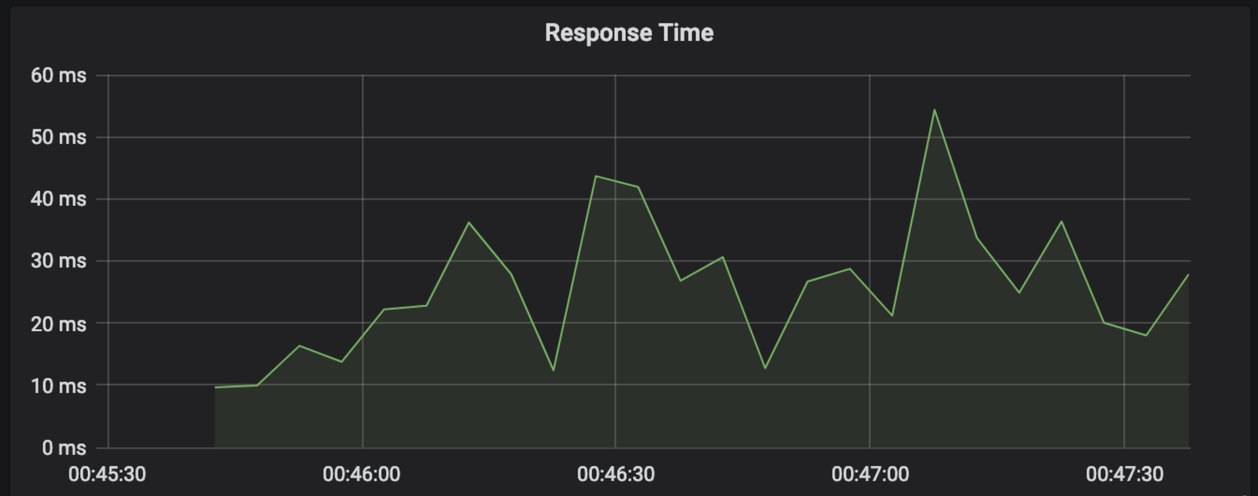




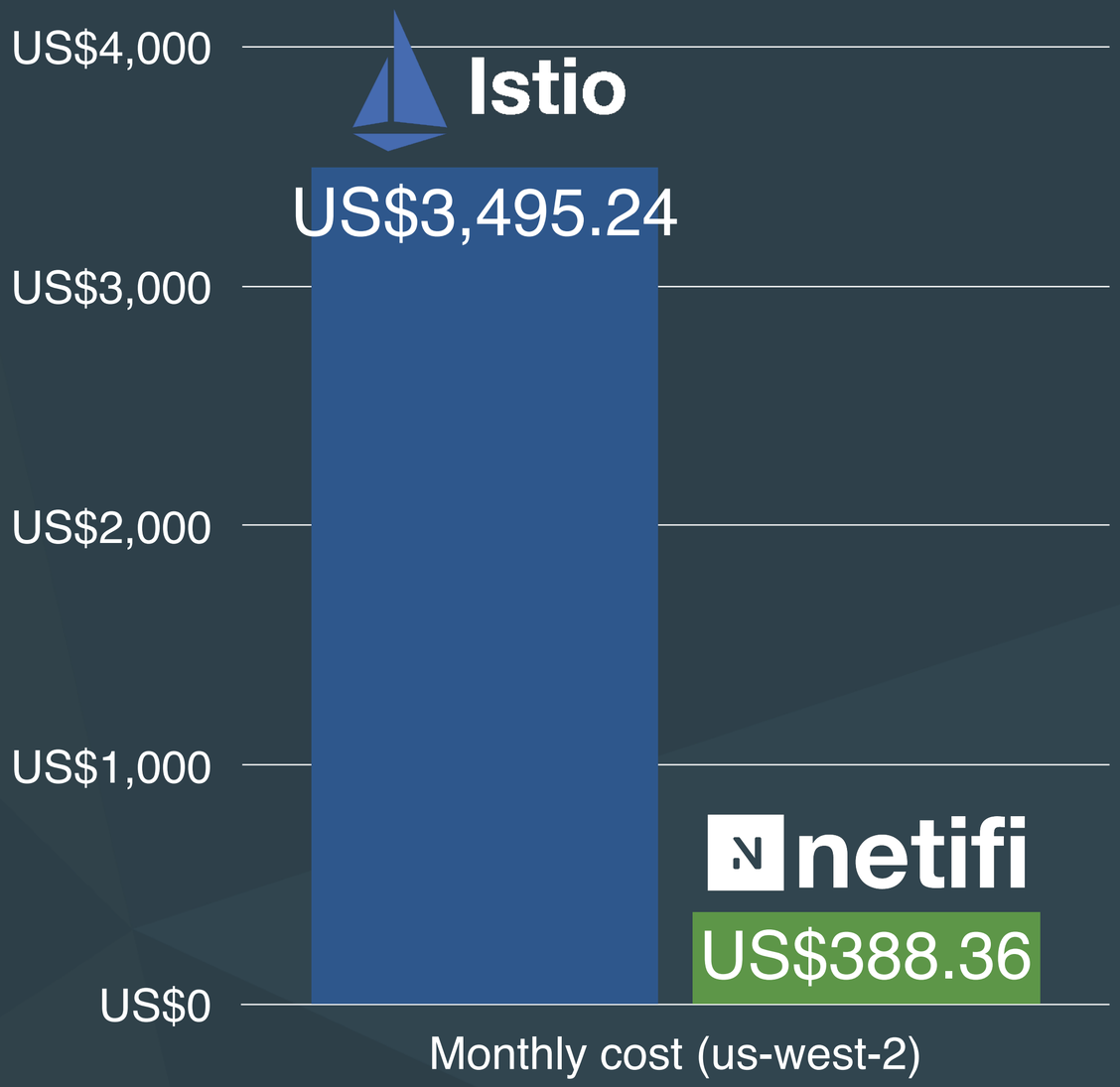
Netifi is 4x Faster

ACTIVE USERS	TOTAL REQUESTS	FAILED REQUESTS	FAILED %	AVG THROUGHPUT	AVG RESPONSE TIME
60	12579723	0	0%	16.3 K	4 ms
300	2873562	9	0.0003%	27.7 K	5.1 ms
1000	4307204	0	0%	37.8 K	28 ms





Netifi is 10x Less Expensive



- Based on n1-standard-4 instances, at a throughput of 40,000 RPS
- Istio requires a total of 36 instances ($3,495.24/month)
- Netifi requires only 4 instances ($388.36/month), an 89% reduction
- Based on RSocket TCP transport. Shared memory transport can boost performance an additional 10x

netifi

Removes Complexity For Everyone

- Developers
  - Simple to use
  - Zero configuration
  - Reduces code complexity
  - Can develop locally

- Ops/security engineers
  - Single deployable
  - No ports to open
  - Application-layer security
  - No complex APIs

- Managers and architects
  - Scales with my business
  - Reduces costs by up to 90%
  - Doesn’t require team to run
  - Decreases time-to-market

netifi

The Future of Microservices

- Easy to deploy and operate
- Fast time-to-market
- Centralized security and access control
- Performant and scalable
- Low cost of ownership
- Works natively across multiple environments
- Real-time and streaming interactions

netifi

Next: Netifi + RSocket in Action!

MULTIPLAYER PAC-MAN With RSocket

@netifi_inc
@rjdegnan
@OlehDokuka

Geekbang 极客邦科技 | TGO 鲲鹏會

TGO鲲鹏会

汇聚全球科技领导者的高端社群

全球12大城市

850+高端科技领导者

使命 Mission

为社会输送更多优秀的科技领导者

愿景 Vision

构建全球领先的有技术背景优秀人才的学习成长平台

扫描二维码，了解更多内容

Bringing Microservices to the Enterprise

info@netifi.com ■ www.netifi.com