



KubeCon



CloudNativeCon

Europe 2020

Easy, Secure, and Fast: Using NATS.io for Streams and Services

Colin Sullivan

Virtual

About Me



Colin Sullivan / [@ColinSullivan1](https://twitter.com/ColinSullivan1)

Product Management @ [Synadia.com](https://synadia.com)

NATS Core maintainer since 2015

Building distributed systems for over 20 years

Agenda



- ✓ NATS Overview
- ✓ Streams and Services
- ✓ Topology
- ✓ Security
- ✓ Additional Features & Roadmap

What is NATS?



KubeCon



CloudNativeCon

Europe 2020

Virtual

NATS is a ten year old production proven cloud-native distributed communications system made for developers and operators who want to spend more time doing their work and less time worrying about how to do messaging.

- ✓ DNA: Performance, simplicity, security, and availability
- ✓ Built from the ground up to be cloud native
- ✓ Multiple qualities of service
- ✓ Support for multiple communication patterns
- ✓ Over 40 types of clients

- Cloud Messaging
 - ✓ Services (microservices)
 - ✓ Event/Data Streaming (observability, analytics)
 - ✓ Command and Control
- IoT and Edge
 - ✓ Telemetry / Sensor Data / Command and Control
- Augmenting or Replacing Legacy Messaging

CNCF Landscape



KubeCon



CloudNativeCon

Europe 2020



The CNCF Landscape is a comprehensive map of cloud native technologies, organized into several key categories:

- App Definition and Development:** Includes Database, Streaming & Messaging, Application Definition & Image Build, and Continuous Integration & Delivery.
- Orchestration & Management:** Includes Scheduling & Orchestration, Coordination & Service Discovery, Remote Procedure Call, Service Proxy, API Gateway, and Service Mesh.
- Runtime:** Includes Cloud Native Storage, Container Runtime, and Cloud Native Network.
- Provisioning:** Includes Automation & Configuration, Container Registry, Security & Compliance, and Key Management.
- Platform:** Includes Certified Kubernetes - Distribution, Certified Kubernetes - Hosted, Certified Kubernetes - Installer, and PaaS/Container Service.
- Observability and Analysis:** Includes Monitoring, Logging, Tracing, and Chaos Engineering.
- Serverless:** A dedicated section for serverless technologies.
- Members:** A section listing various members of the CNCF community.
- Special:** A section for special projects and partners.

A large red arrow points from the 'Database' category towards the 'Streaming & Messaging' category.

Cloud Native Landscape
This landscape is intended as a map through the previously uncharted terrain of cloud native technologies. There are many routes to deploying a cloud native application, with CNCF Projects representing a particularly well-traveled path.
i.cncf.io

Joined CNCF as an incubation project in 2018

<https://landscape.cncf.io>

Contribution Statistics



KubeCon



CloudNativeCon

Europe 2020

Virtual

- Over 1000 contributors, over 100 with more than 10 commits
- Over 90 public repos
 - 18,600+ GitHub stars across repos
- ~79M NATS Server Docker Hub pulls
- ~50M NATS Streaming Server Docker Hub pulls
- 2300+ Slack members
- 35 releases of the NATS server since June 2014, ~ = 5/year

<https://nats.devstats.cncf.io/d/9/developers-summary>



Derek Collison

Founder and CEO at Synadia

Founder and former CEO at Apcera
CTO, Chief Architect at VMware
Architected CloudFoundry
Technical Director at Google
SVP and Chief Architect at TIBCO

Created by Derek Collison

Derek has been building messaging systems and solutions > 30 yrs

Maintained by a highly experienced messaging team

Engaged User Community

End Users



Virtual



clarifai



cruise



htc



Pivotal



SCHAEFFLER

SIEMENS



TESLA



VIACOMCBS

vmware



- The server is a single binary deployable anywhere
- 10.2 MB docker image with no external dependencies
- “Text-based” protocol with a handful of verbs

PUB | HPUB | SUB | UNSUB | CONNECT | INFO | MSG | HMSG | -ERR | +OK | PING | PONG

- Low Configuration
 - ✓ Clients only need a url and credentials
 - ✓ Servers auto-discover
 - ✓ You can share configuration files amongst servers
- Simple and Straightforward API

NATS Clients



nats.go

Golang client for NATS, the cloud native messaging system.



go golang microservices nats cloud-native

Go ★ 2,265 🍴 303 📄 Apache-2.0 3 issues need help Updated a day ago

nats.rb

Ruby client for NATS, the cloud native messaging system.



ruby client messaging cncf pubsub nats eventmachine

Ruby ★ 823 🍴 131 📄 Apache-2.0 Updated a day ago

nats.java

Java client for NATS



java client middleware messaging nats messaging-library

Java ★ 194 🍴 68 📄 Apache-2.0 Updated a day ago

nats.ex

Elixir client for NATS, the cloud native messaging system. <https://nats.io>



client elixir nats nats-io

Elixir ★ 33 🍴 11 📄 MIT 1 issue needs help Updated 6 days ago

nats.js

Node.js client for NATS, the cloud native messaging system.



JavaScript ★ 672 🍴 96 📄 Apache-2.0 Updated 8 days ago

nats.net

The official C# Client for NATS



client visual-studio csharp messaging message-bus pubsub

C# ★ 232 🍴 63 📄 Apache-2.0 3 issues need help Updated 2 days ago

nats.c

A C client for NATS



c messaging message-bus message-queue messaging-library

C ★ 139 🍴 45 📄 Apache-2.0 Updated 7 days ago

nats.py

An asyncio based Python 3 client for NATS



aiο nats python3 asyncio cloud-native aiο-nats

Python ★ 187 🍴 34 📄 Apache-2.0 Updated 4 days ago



KubeCon



CloudNativeCon

Europe 2020

Virtual

Messaging Patterns: Streams and Services

Mess

Streams and Services



- Streams
 - ✓ A flow of data
 - ✓ Fan out
- Services
 - ✓ Do some work and return a result
 - ✓ Load balanced

Application Level Patterns



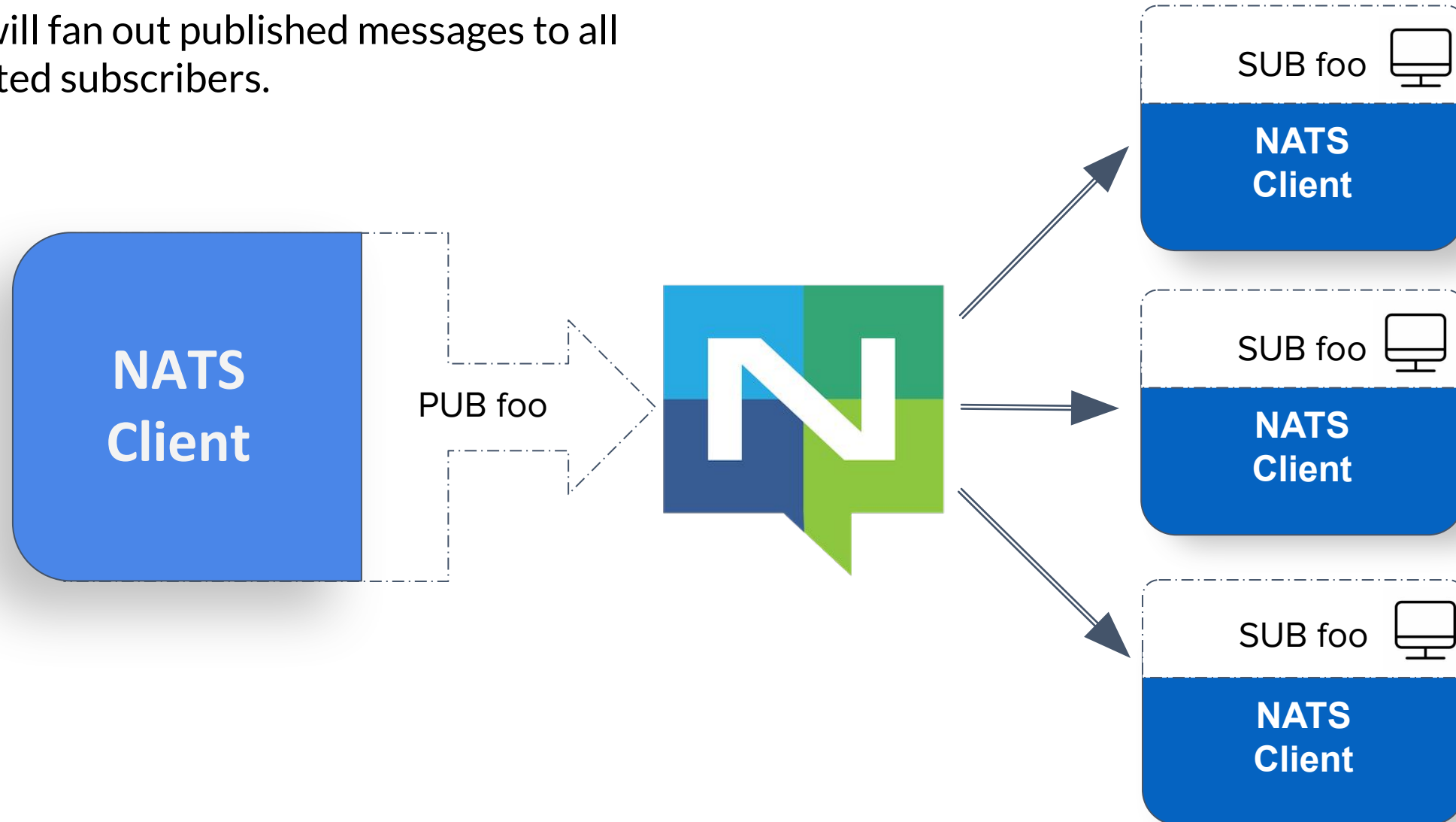
- ✓ Request/Reply
- ✓ Publish/Subscribe
- ✓ Load Balanced Queue Subscribers
- ✓ New high level API Coming!

A subject is simply a string representing an interest in data.

- Simple subject: **foo** or **weather**
- Hierarchically Tokenized: **foo.bar**, **weather.us.co.denver**
- Wildcard subscriptions
 - ✓ **foo.*** matches **foo.bar** and **foo.baz**.
 - ✓ **foo*.bar** matches **foo.a.bar** and **foo.b.bar**.
 - ✓ **foo.>** matches any of the above
 - ✓ **>** matches everything in NATS
- Unique subjects for 1:1 addressability

Streams 1:N

NATS will fan out published messages to all interested subscribers.



Streaming Code



```
nc, err := nats.Connect("demo.nats.io")
if err != nil {
    log.Fatal(err)
}

nc.Subscribe("foo", func(m *nats.Msg) {
    log.Println(" [Received]", string(m.Data))
})

nc.Publish("foo", []byte("Hello"))
```

Services 1:1

Using unique reply subjects, clients can make requests to services that respond only to the requestor, creating a 1 to 1 relationship.



Service Code



```
nc, err := nats.Connect("127.0.0.1")
if err != nil {
    log.Fatal(err)
}
nc.Subscribe("help", func(m *nats.Msg) {
    m.Respond([]byte("I can help!"))
})
msg, err := nc.Request("help", []byte("Need help!"), 1*time.Second)
if err != nil {
    log.Fatal(err)
}
log.Println("[Received]", string(msg.Data))
```

Load Balancing



KubeCon

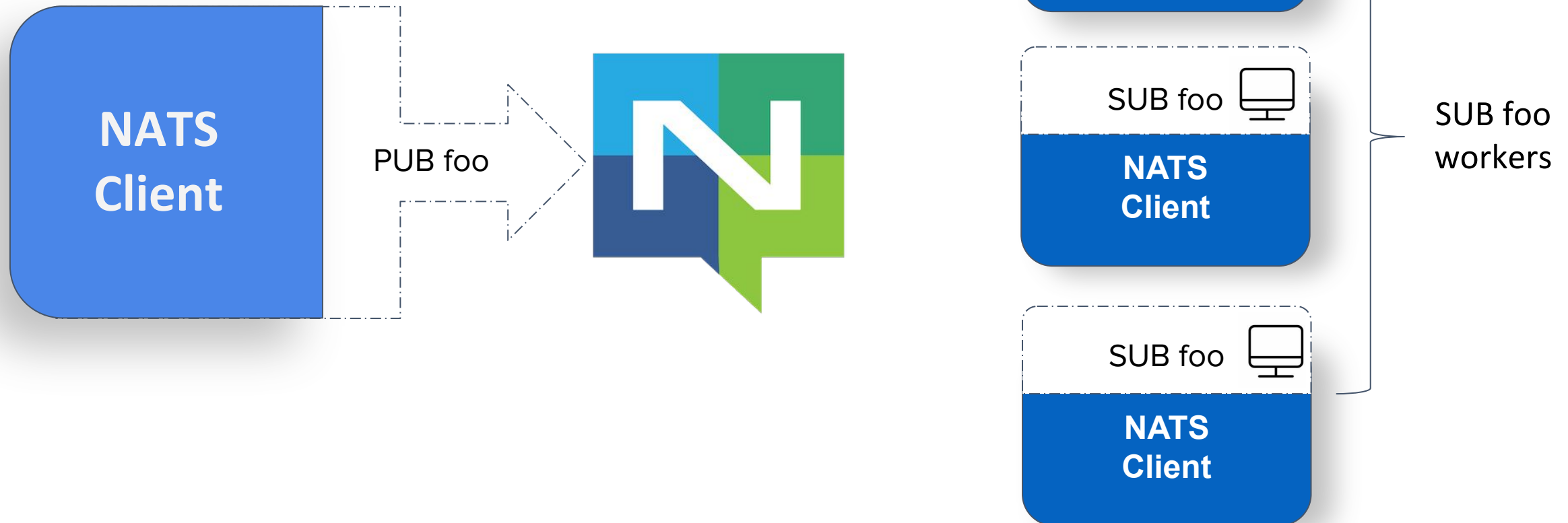


CloudNativeCon

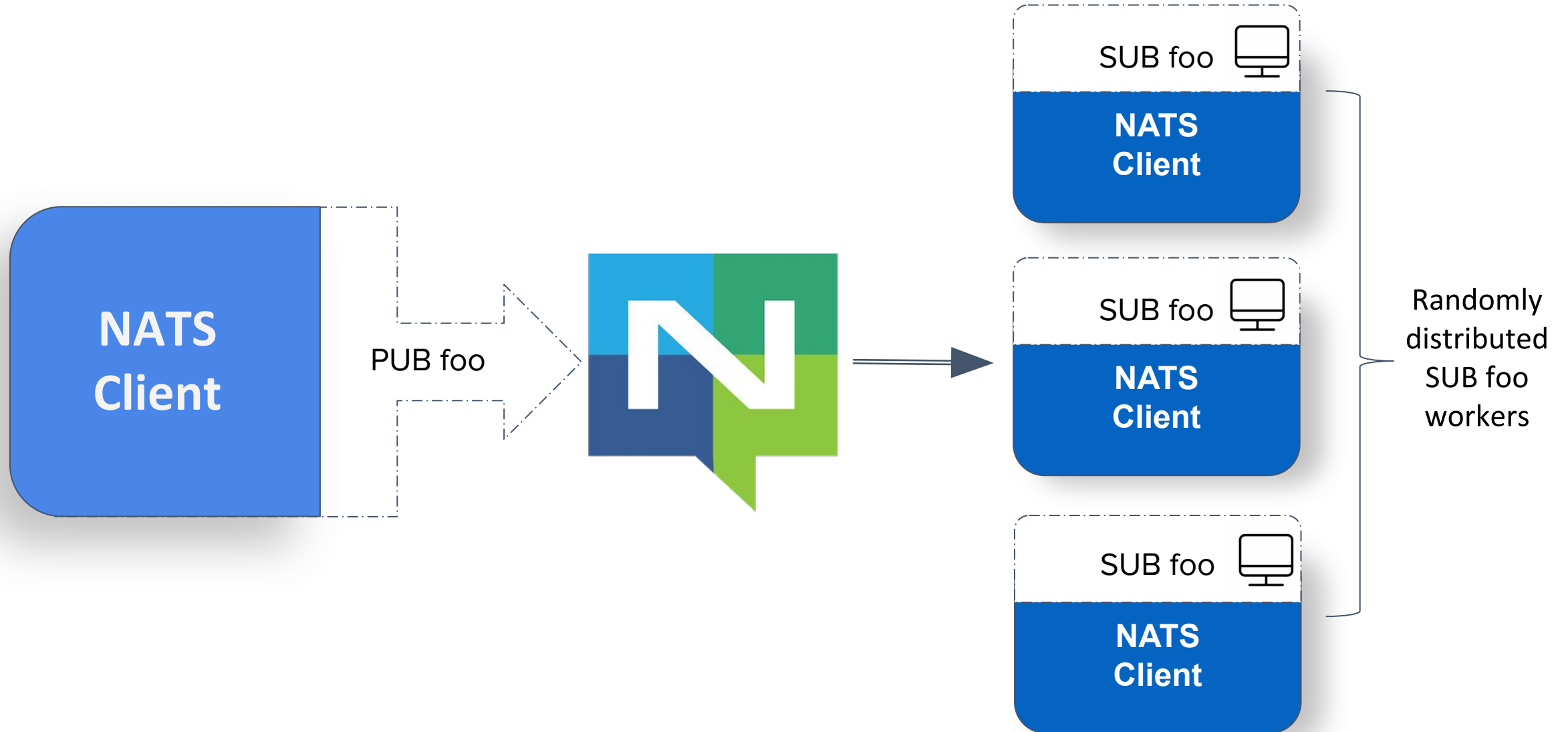
Europe 2020

Virtual

When subscribers are grouped together in a named queue group, NATS will randomly distribute messages to the subscribers, allowing NATS to act as a layer 7 load balancer for services.



Load Balancing



Load Balancing



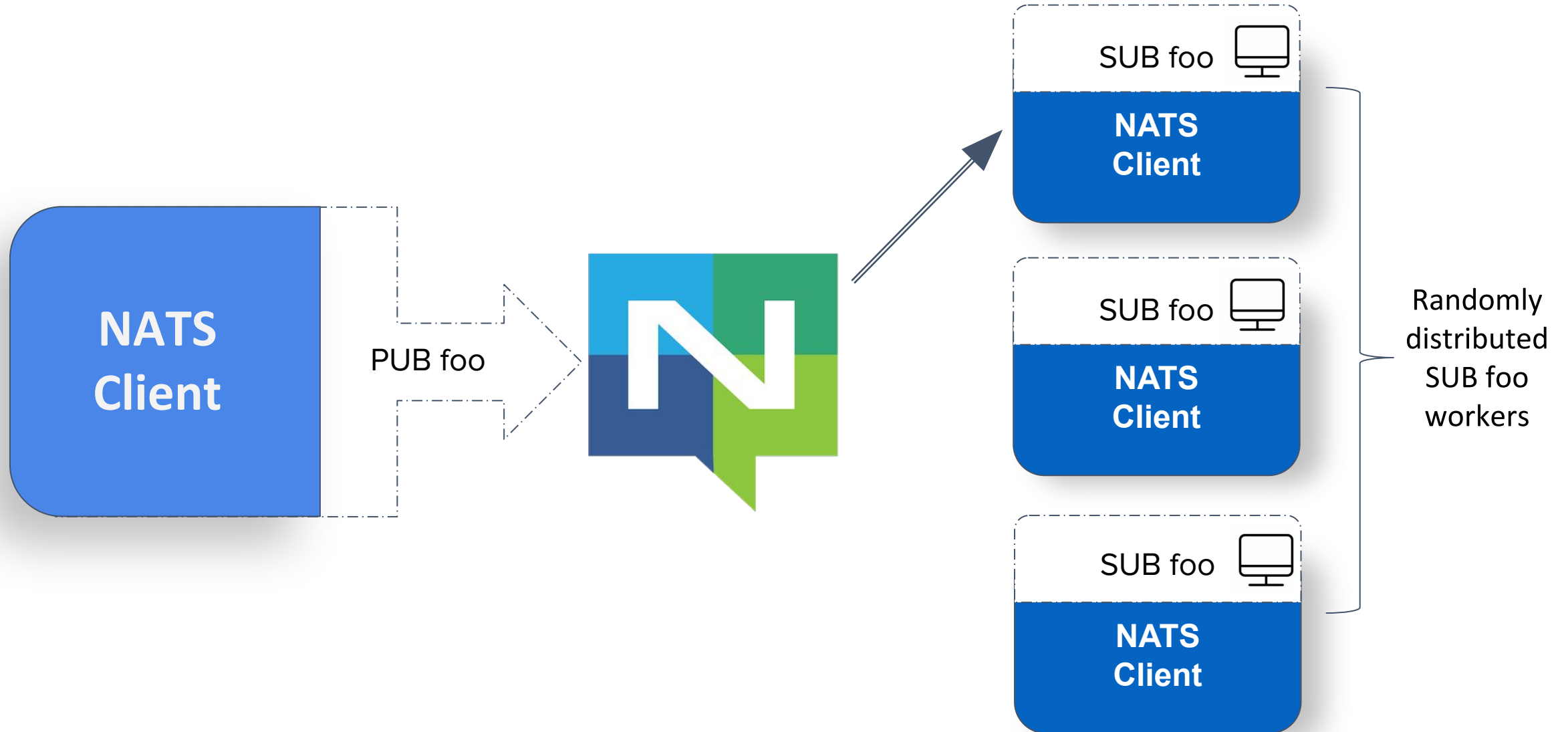
KubeCon



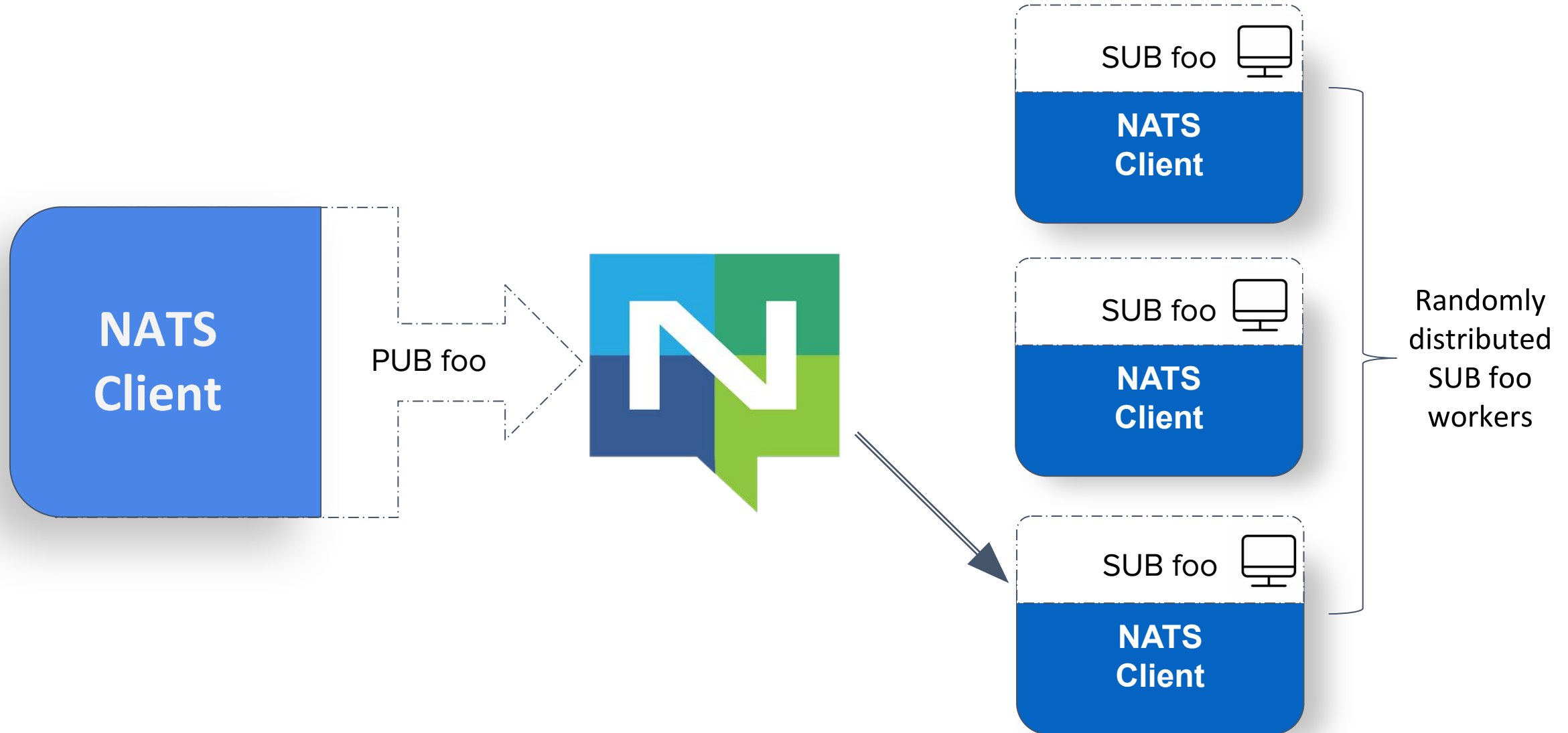
CloudNativeCon

Europe 2020

Virtual



Load Balancing



Topology



KubeCon

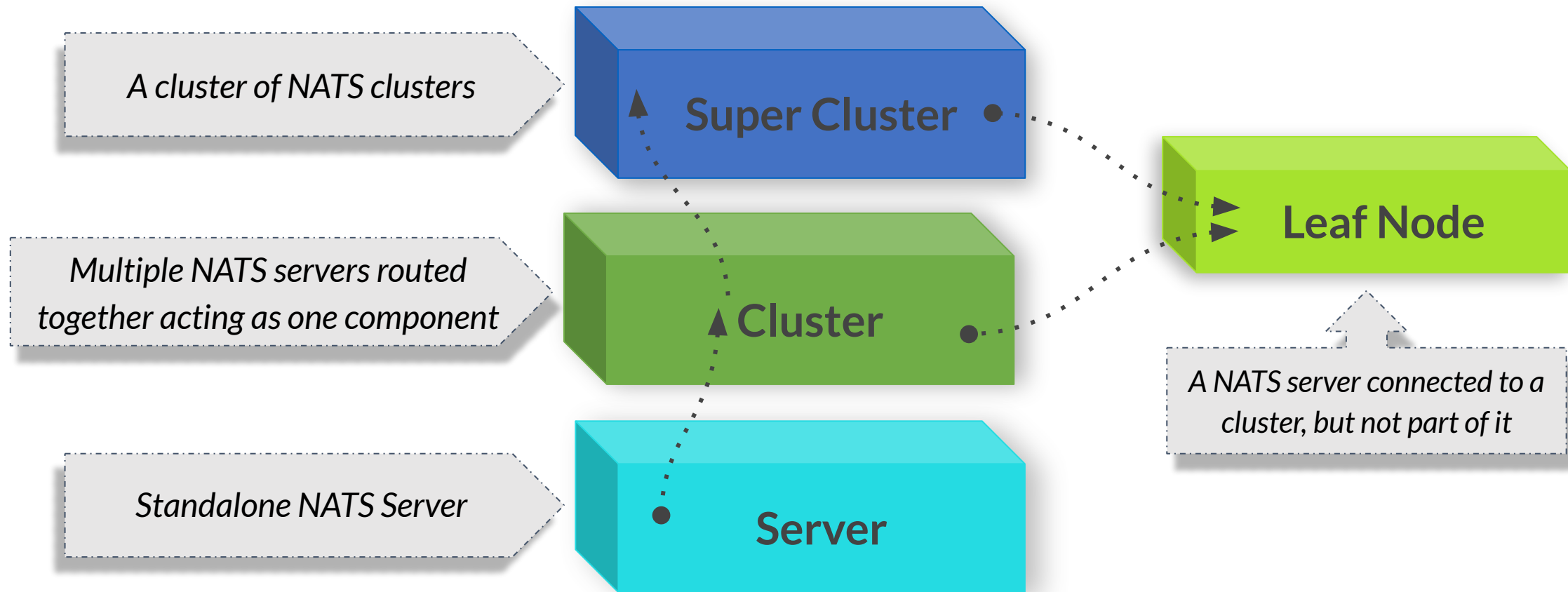


CloudNativeCon

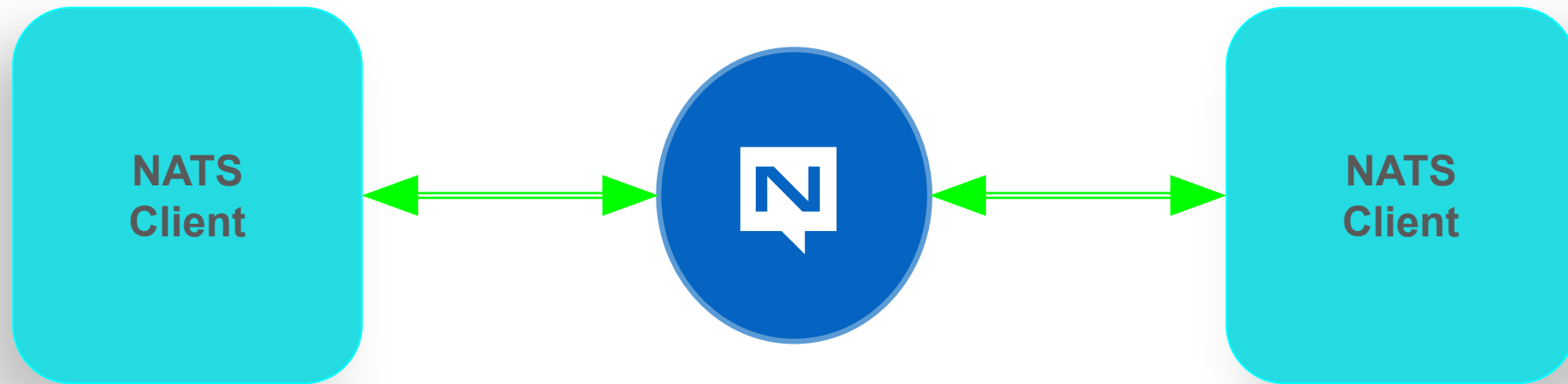
Europe 2020

Virtual

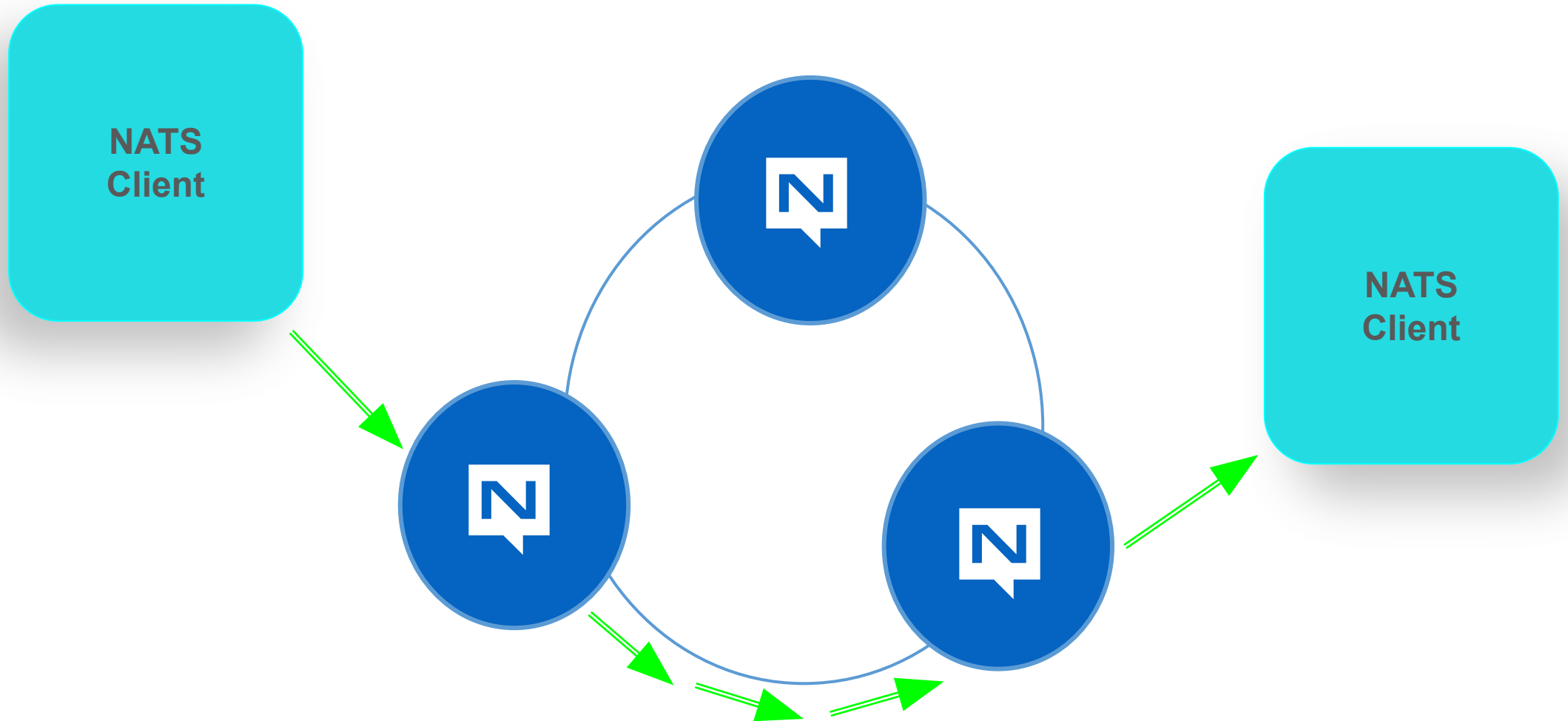
Topology Building Blocks



Clients require **no awareness** of server topology beyond a connection URL.

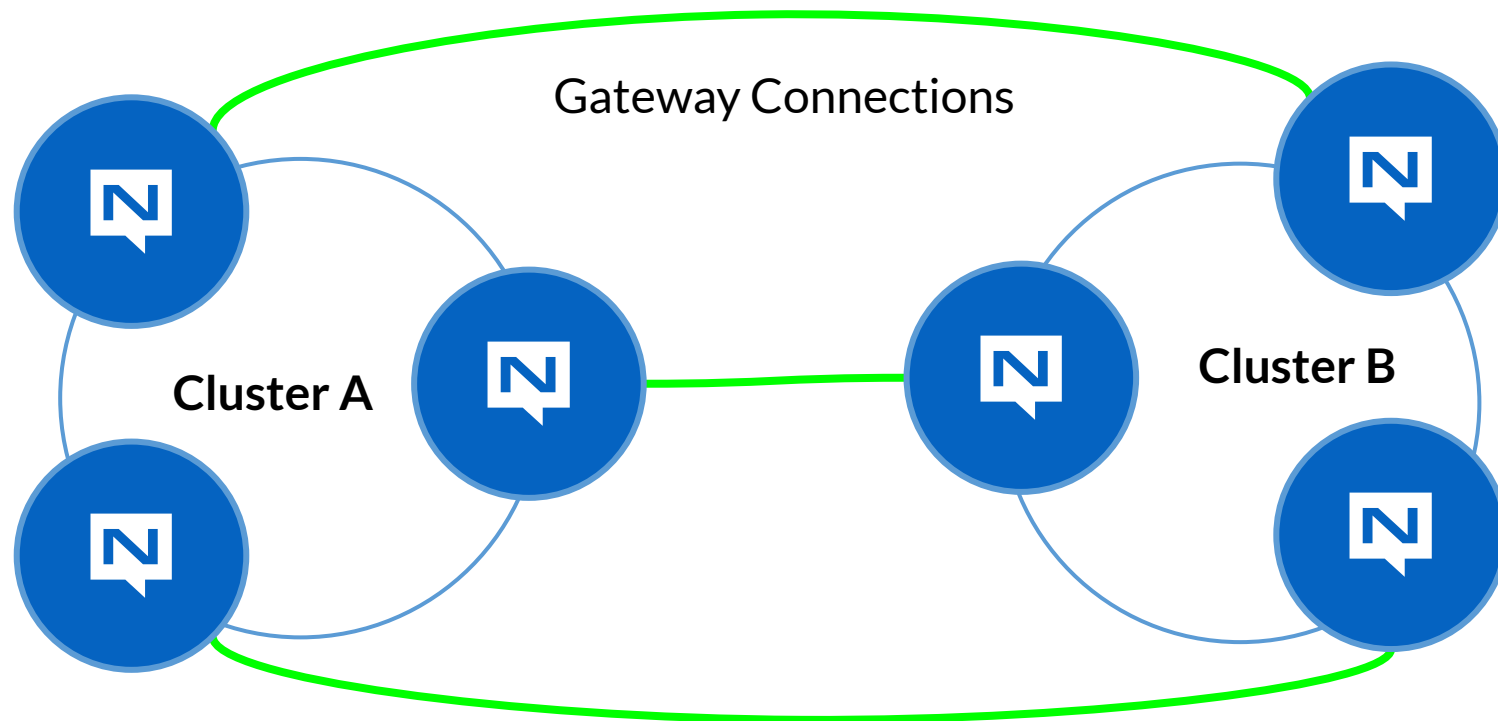


Clusters



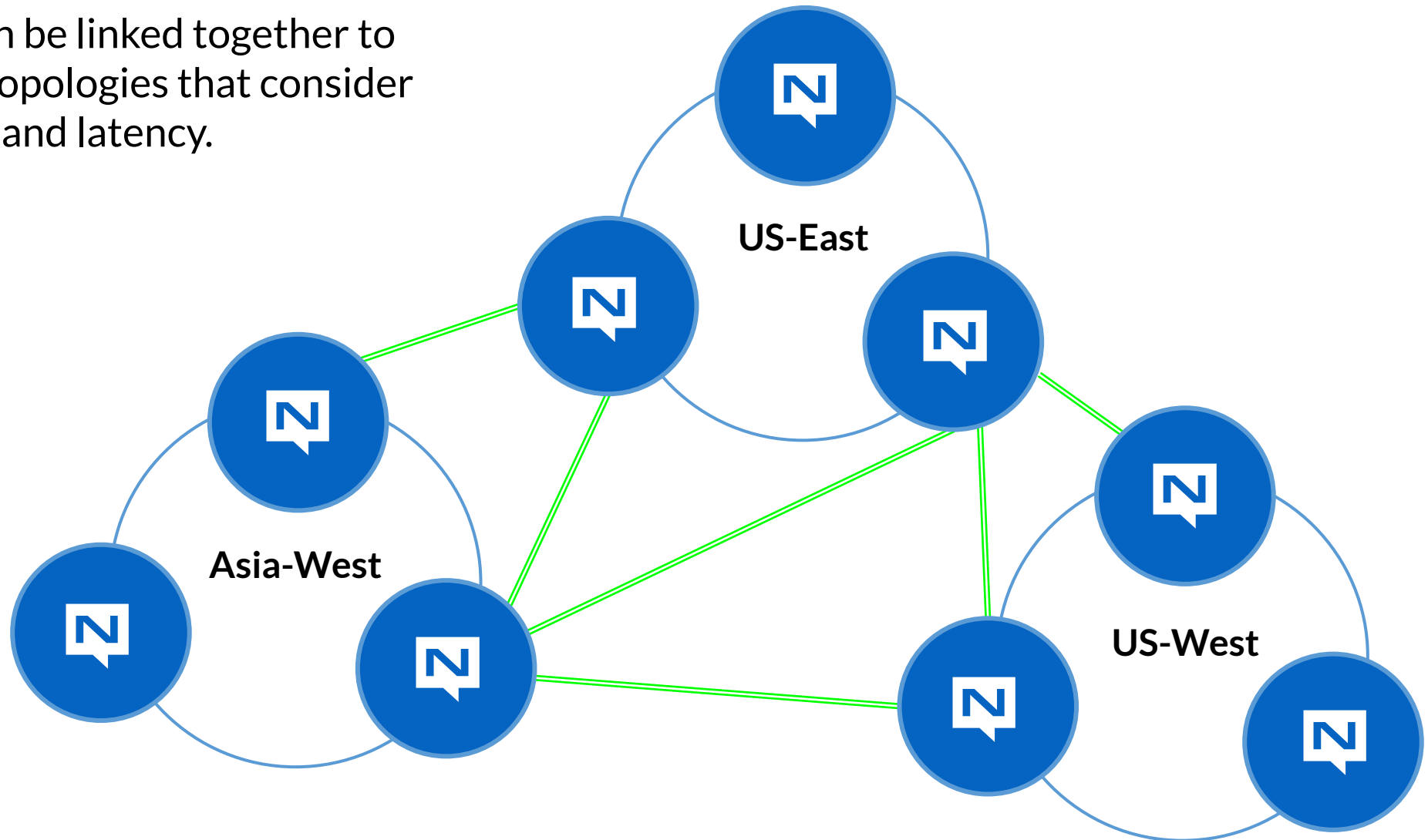
Superclusters

Superclusters are clusters of clusters connected together with gateway connections. They use a spline based technology to ensure resiliency and optimize traffic across clusters.



Superclusters



Multiple clusters can be linked together to form vast network topologies that consider WAN network links and latency.

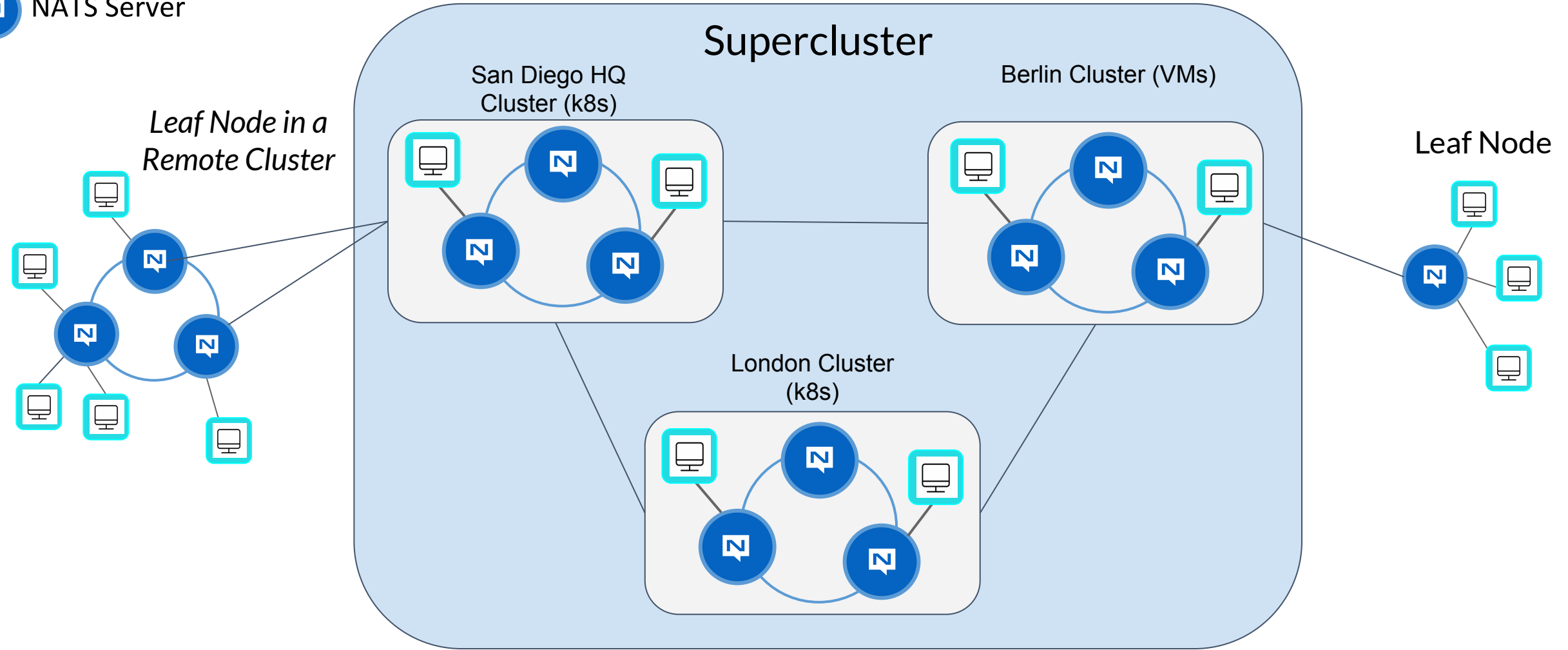


- ✓ A leaf node is a single NATS server extended out from a cluster or remote server.
- ✓ Leaf nodes extend clusters via a hub and spoke topology.
- ✓ Leaf nodes allow you to bridge separate security domains.
- ✓ Ideal for edge computing, IoT hubs, or data centers that need to be connected to a global, regional, or national NATS deployment.
- ✓ Transparently bridge on-premise and cloud deployments.

Global Deployment



-  Clients/Microservices/Devices
-  NATS Server



Security



KubeCon



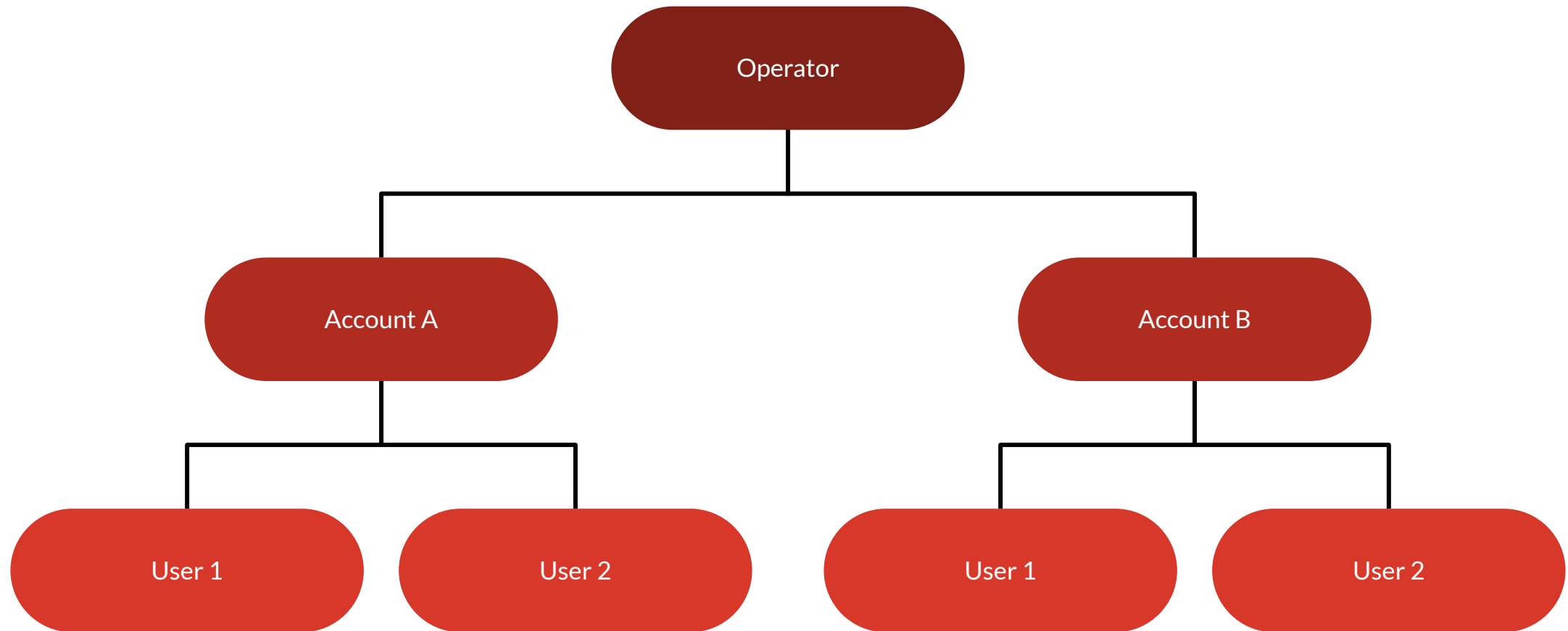
CloudNativeCon

Europe 2020

Virtual

- Full TLS Support: CA certificates, bidirectional support, default to most secure ciphers.
 - ✓ Support for DN or SAN in certificates for NATS user identity
- Support for standard user/password auth
- Permissions restrict who can send and receive on what subjects
- Change these through configuration reload at runtime with **zero downtime**.
- Operator Mode with NATS \geq 2.0

Multi-Tenancy



Distributed Security - Trust



NATS allows you to define **Operators**, **Accounts**, and **Users** within a NATS deployment.

- **Operator**: Root of trust for the system, e.g. An enterprise operator.
 - Create **Accounts** for account administrators. An account represents an organization with a secure context within the NATS deployment, for example a VAS system, an IT system monitoring group, a set of microservices, etc. Account creation would likely be managed by a central group.
- **Accounts** define limits and may securely expose services and streams
 - Account managers create **Users** with permissions
- **Users** have specific credentials and permissions.

Distribute Security - Accounts



- Accounts are isolated communication contexts allowing secure multi-tenancy
- Bifurcate technology from business driven use cases
 - ✓ Data silos are created by design, not software limitations
- Easy, Secure and Cost Effective
 - ✓ One NATS deployment for operators to manage
 - ✓ Decentralized - organizations can self-manage
- Share data between accounts
 - ✓ Secure Streams and Services
 - ✓ Only mutual agreement will permit data flow

JWTs are used to represent identities in NATS

- User, Account, Cluster, or Server

User JWTs Contain

- Account NKey (Issuer)
- Public NKey (Subject)
- Friendly Name
- Permissions, limits, not-before and expiration
- NKey is a NATS Key - ED25519 key made easy

Distributed Security - NKeys



Used by the NATS Identity authentication and authorization system.

- ED25519 based encoded keys made simple
 - ✓ Fast and resistant to side-channel attacks
 - ✓ Sign and Verify
- NATS servers **never see private keys**
 - ✓ Server sends nonce during connect then verifies the nonce signed by the user's private key, and user JWT signed by an account private key.
- JWT associate users with accounts and permission sets
- Managed with a the NATS `nsc` command line interface



KubeCon



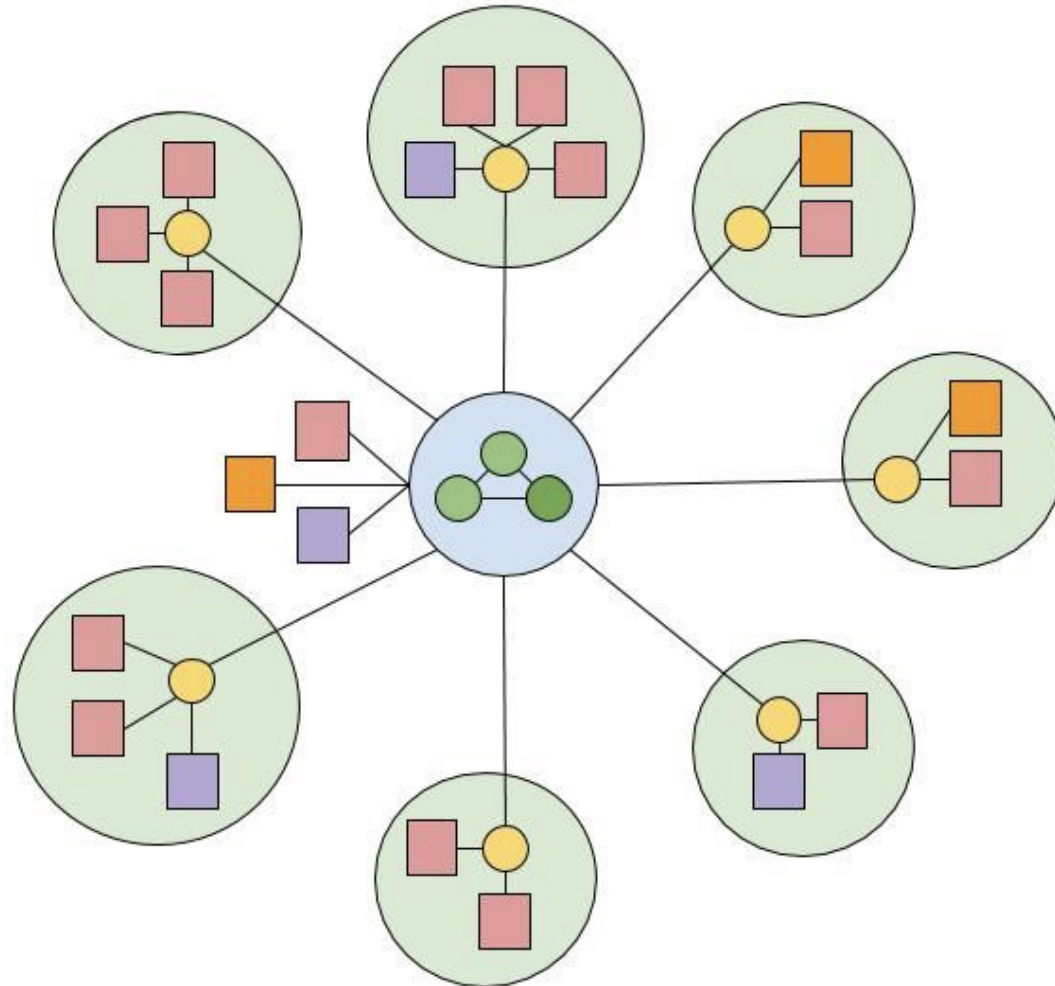
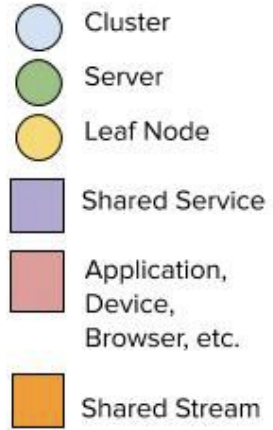
CloudNativeCon

Europe 2020

**Topology + Security =
Adaptive Edge Architecture**

Virtual

Adaptive Edge Architecture



Services and Streams can be located anywhere and have varying levels of accessibility based on security policy.

Adaptive Edge Architecture



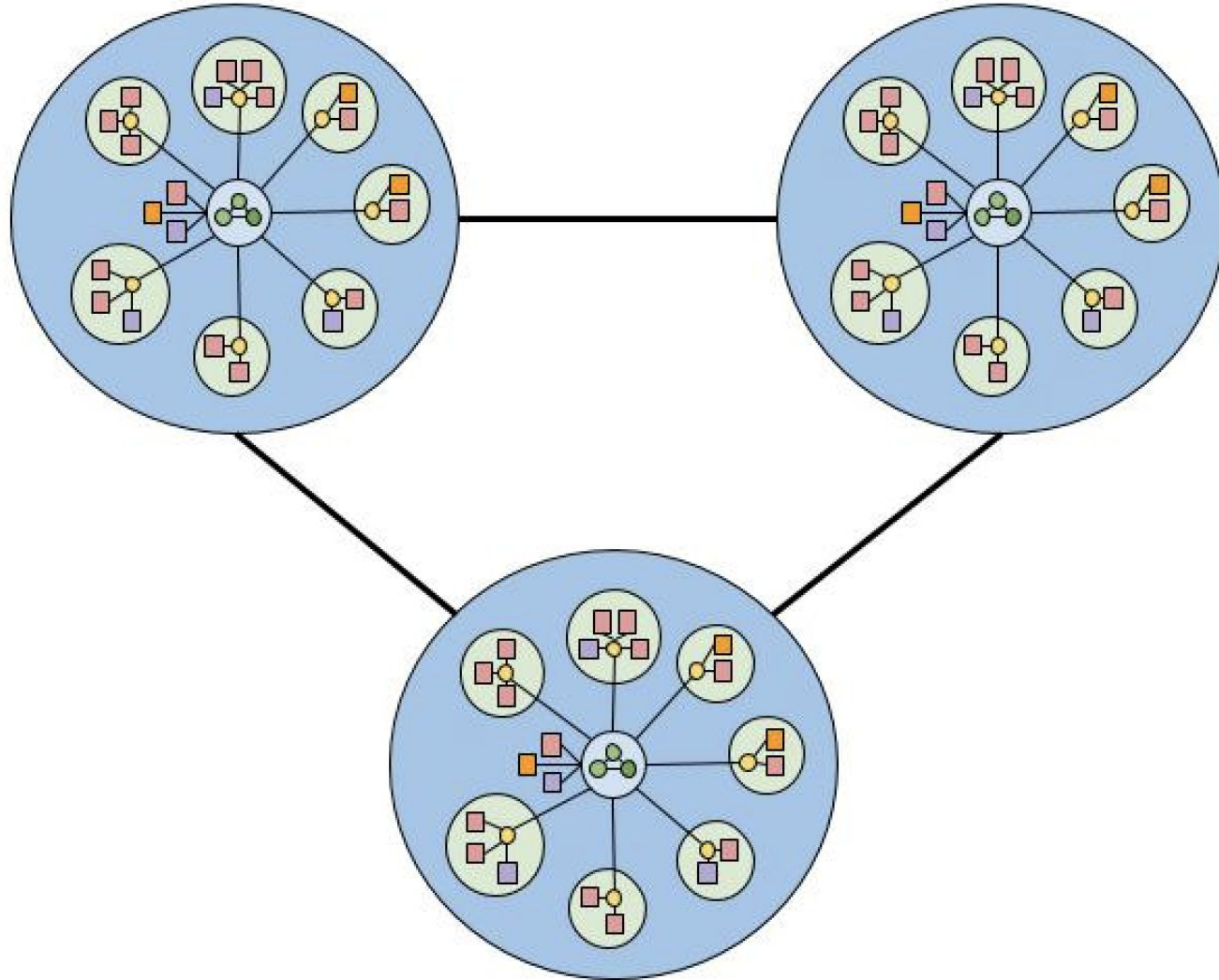
KubeCon



CloudNativeCon

Europe 2020

Virtual



Supercluster regional or remote deployments for large scale deployments.

Adaptive Edge Architecture



Use Case	Central	Shared Services or Streams	Remote Entity	Endpoints
Connected Car	Headquarters	Location Services, Weather, Metrics, Security	Vehicle	Various systems within the vehicle
Manufacturing	Regional, Divisional, or National Headquarters per cluster	Analytics, QA, Schematic updates, inventory	Factory	Line Equipment
Retail	Regional Headquarters per cluster	Points programs, Ad rewards, coupons, logistics	Stores, Distribution Centers	Scanners, POS devices, inventory
Energy	Headquarters and DR sites	Power source scheduling, outage recovery coordination, metrics	Microgrids, Wind Turbine sites, Feeder Lines, Mobile Substations	Photovoltaic, Turbines, power boxes, field diagnostics, smart meters.
Aviation	Each Airport hosts a cluster in the supercluster	Weather, Socials, Air Traffic	Gates, Airplanes, Luggage systems	Airline systems, Gate software, Airport Applications
Cellular/Mobile	Headquarters, with many regional clusters	Thousands of services, from call blocking, forwarding to IoT specific services.	Cell Towers (5g), Macrocell, Small Cell Sites	Web and phone applications, websites
Credit Card Services	Each cluster in a regional headquarters and DR sites	Points programs, fraud detection, country specific value-added services	Regional or by location (brick and mortar)	Websites, Applications, POS devices.
Cruise Lines	Global Supercluster	Logistics, manifest management, Passenger loyalty,	Cruise Ship	Various Systems from Engine to Inventory
Shipping Container Ships	Global Supercluster	Logistics, Traffic, Manifest management, planned maintenance,	Cargo Ship	Inventory management equipment, Location telemetry
Trucking	Regional / International Supercluster	Dispatch Services, Maintenance, Fleet management services, Traffic Services	Vehicle	Location telemetry, component health (engine/tire management).



KubeCon



CloudNativeCon

Europe 2020

Virtual

Adaptive Edge Example

Airline Topology Example

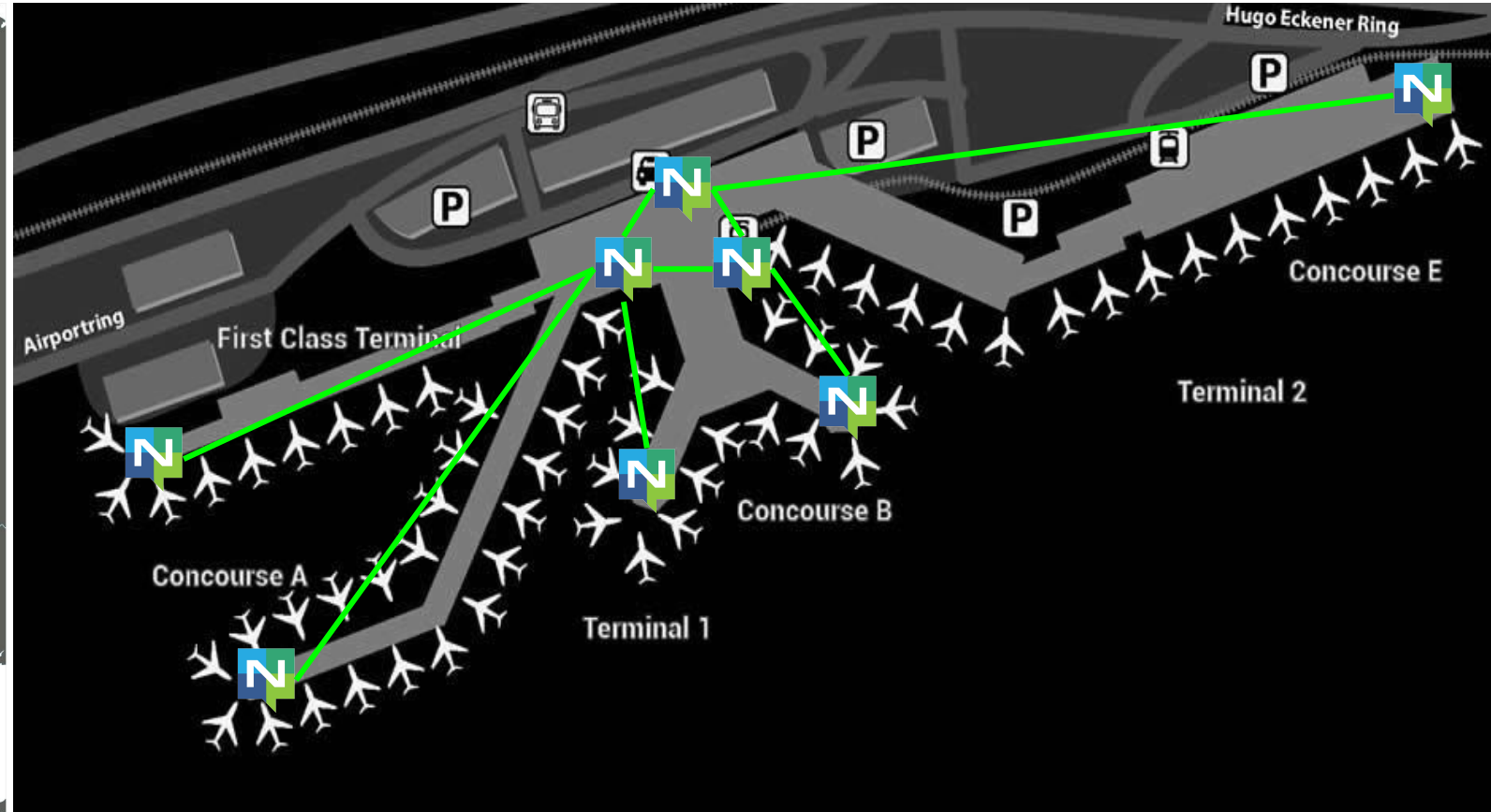


KubeCon



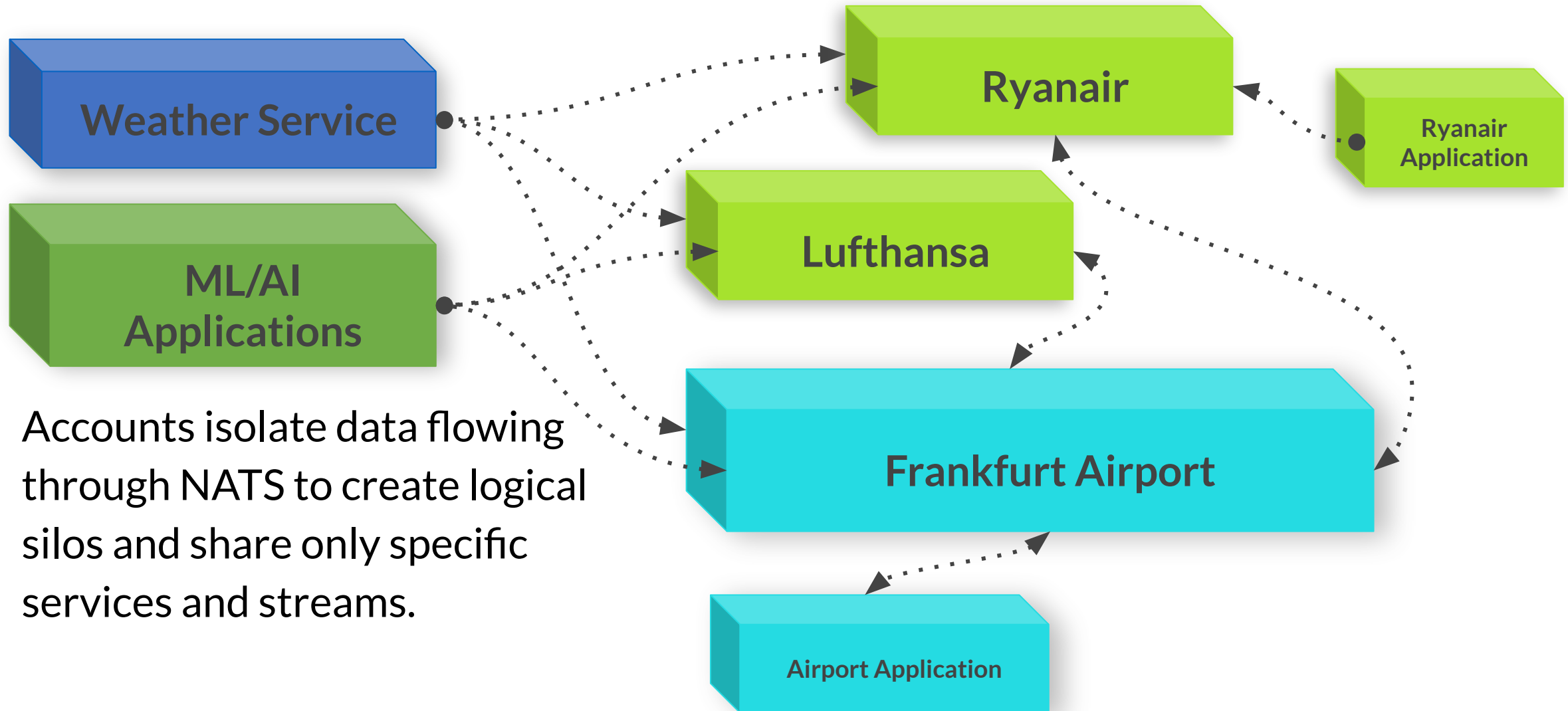
CloudNativeCon

Europe 2020



Airports connected to create a supercluster with leaf nodes extending to gates and even planes.

Airline Account Example





KubeCon



CloudNativeCon

Europe 2020

Virtual

Performance and Scalability

Performance



18 million messages per second with one server, one data stream.
Up to **80 million** messages per second per server with multiple data streams.

Benchmark_____Pub0b_Payload-20	30000000	55.1 ns/op	199.78 MB/s
Benchmark_____Pub8b_Payload-20	30000000	55.8 ns/op	340.21 MB/s
Benchmark_____Pub32b_Payload-20	20000000	63.4 ns/op	694.34 MB/s
Benchmark___Pub128B_Payload-20	20000000	79.8 ns/op	1766.47 MB/s
Benchmark___Pub256B_Payload-20	20000000	98.1 ns/op	2741.51 MB/s
Benchmark_____Pub1K_Payload-20	5000000	283 ns/op	3660.72 MB/s
Benchmark_____Pub4K_Payload-20	1000000	1395 ns/op	2945.30 MB/s
Benchmark_____Pub8K_Payload-20	500000	2846 ns/op	2882.35 MB/s
Benchmark_AuthPub0b_Payload-20	10000000	126 ns/op	86.82 MB/s
Benchmark_____PubSub-20	10000000	135 ns/op	
Benchmark_____PubSubTwoConns-20	10000000	136 ns/op	
Benchmark_____PubTwoQueueSub-20	10000000	152 ns/op	
Benchmark___PubFourQueueSub-20	10000000	152 ns/op	
Benchmark___PubEightQueueSub-20	10000000	152 ns/op	

The health and availability of the system as a whole is prioritized over servicing any individual client or server...

- ✓ NATS server “selfish optimization”
 - Protects against *Slow Consumers*
- ✓ Full Mesh clustering of NATS servers
- ✓ Server and client connections self heal

... this creates a NATS dial-tone, always on, always available.

- Auto-Discovery
 - ✓ Automatically Exchange Server Topology
 - ✓ Server ↔ Server
 - ✓ Server → Client
- No configuration updates
 - ✓ Failover to auto-discovered servers
- Great for rolling upgrades



KubeCon



CloudNativeCon

Europe 2020

Virtual

Message Delivery Guarantees

Message Guarantees



KubeCon



CloudNativeCon

Europe 2020

Virtual

NATS supports two delivery modes providing the following guarantees:

- At most once (*Core*)
 - ✓ No guarantee of delivery - messages can be lost - applications must detect and handle lost messages
- At least once (*NATS Streaming or JetStream enabled core servers*)
 - ✓ A message will always be delivered, but in certain cases may be delivered more than once

Exactly once is arguably unnecessary, always complex, and inevitably slow. But due to popular demand **we've decided to support it in JetStream.**

JetStream supports:

- ✓ **At-least-once** delivery
- ✓ Store messages and replay by time or sequence
- ✓ Embedded NATS server subsystem with an option to enable
- ✓ Wildcard Support
- ✓ NATS 2.0 Security
- ✓ Data at rest encryption
- ✓ Cleanse specific messages (GDPR)
- ✓ Horizontal scalability
- ✓ Persist **Streams** and replay via **Consumers**

JetStream and NATS Streaming



NATS Streaming will continue to be supported.

- ✓ 50 million docker downloads
- ✓ Deployed in production globally
- ✓ Bug fixes and Security fixes until June of 2022

Moving forward...

- ✓ New NATS enabled applications should prefer Jetstream
- ✓ We will provide a migration path to use JetStream
- ✓ New NATS streaming development will occur in JetStream



KubeCon



CloudNativeCon

Europe 2020

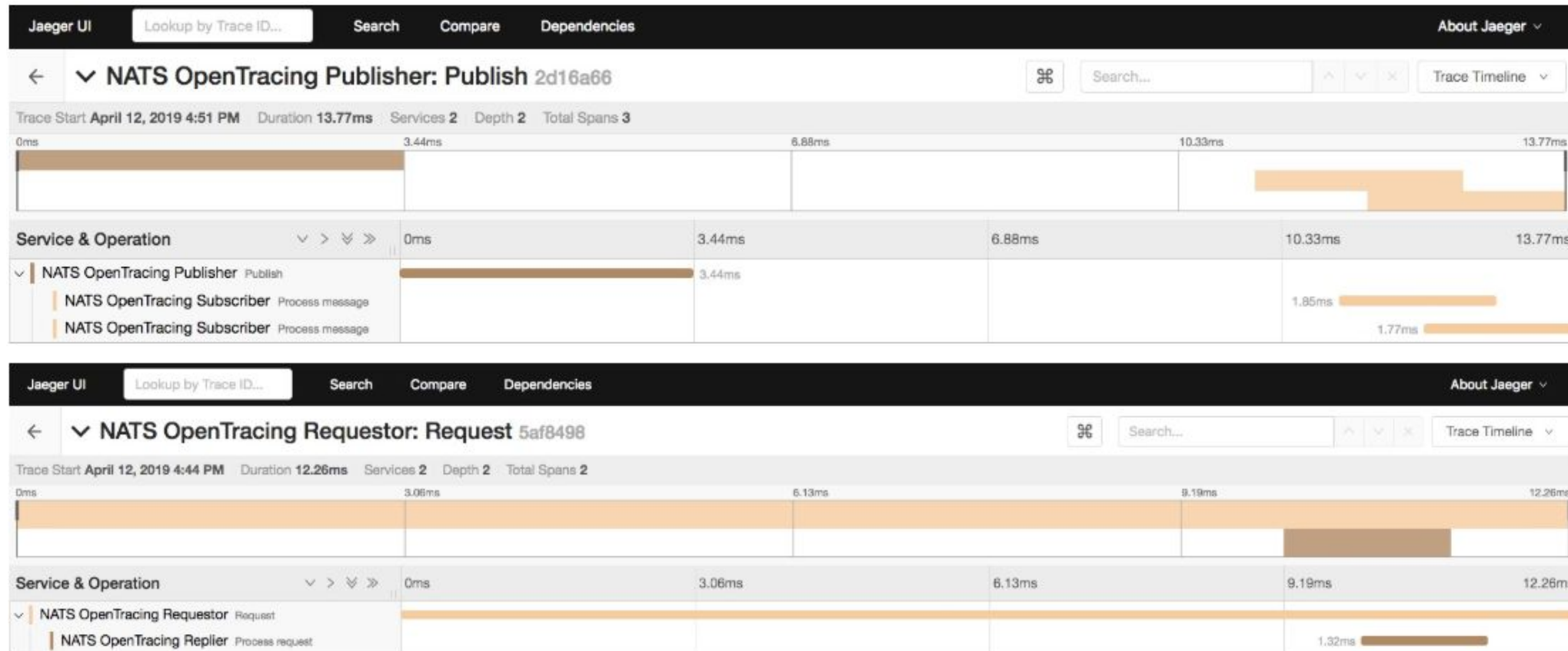
Virtual

Tracing, Monitoring, and Kubernetes Deployments

Distributed Tracing



OpenTracing reference implementations are provided for the **java** (not.java repo) and **go** (not.go repo). Using a simple API, encode and decode NATS messages to be traced with **Jaeger**.



Integrations



We're continuing to integrate NATS with other technologies.

- Spring.io
 - ✓ NATS Spring Boot Starter
 - ✓ NATS Cloud Stream Binder
- NATS Kafka Bridge
 - ✓ Support for bridging to and from Kafka topics
- NATS JMS Bridge
 - ✓ Support for bridging to and from JMS vendors, first with IBM MQ series

Surveyor can monitor your entire deployment from a single container or process paired with Prometheus and Grafana.

- ✓ Provides a comprehensive view of entire NATS deployment
- ✓ No sidecars to deploy
- ✓ K8s, docker compose, or bare metal deployments
- ✓ Run using Docker Compose
- ✓ Requires NATS 2.0 Security and System Credentials

NATS Surveyor

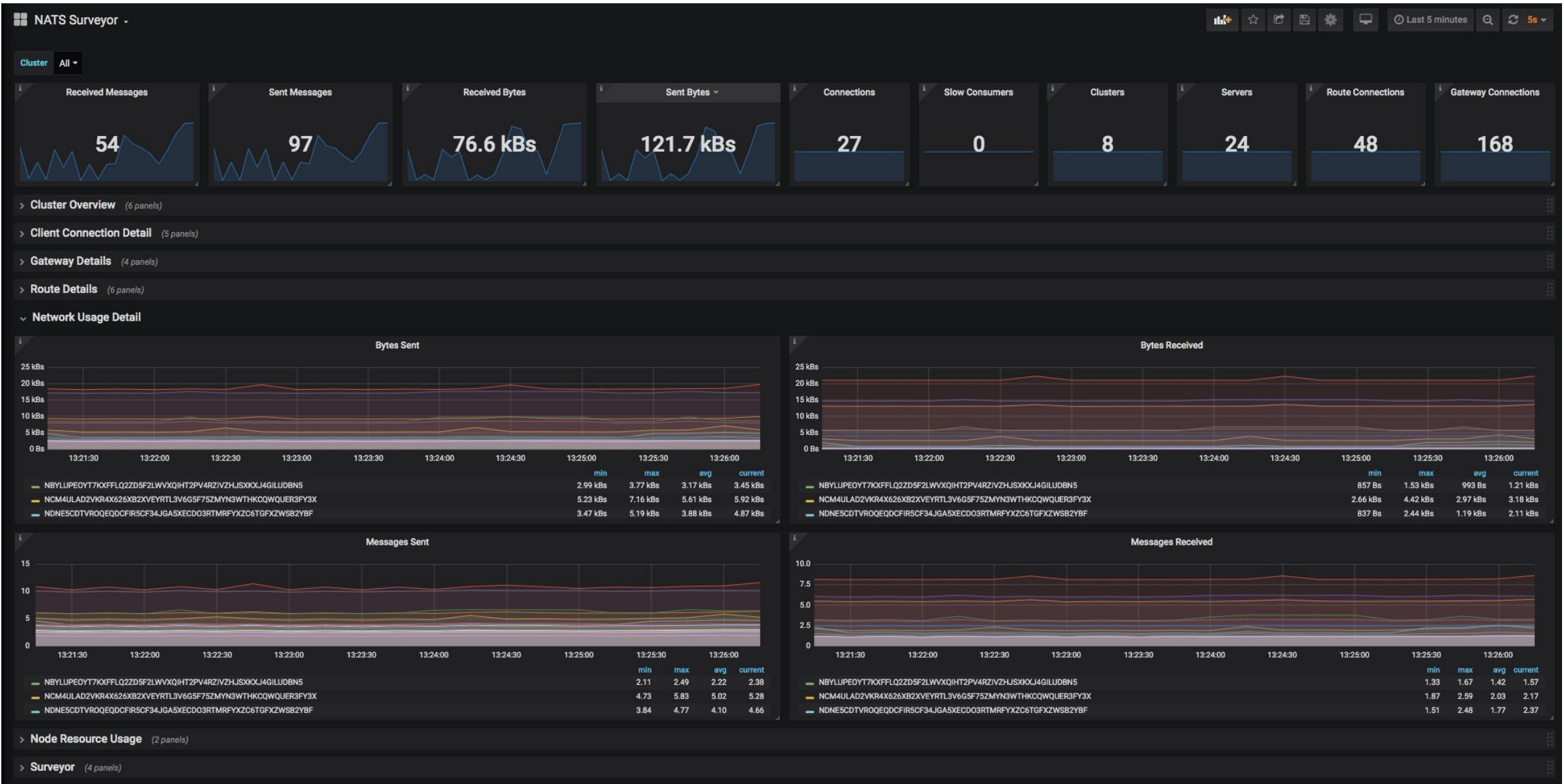


KubeCon



CloudNativeCon

Europe 2020



Kubernetes Deployments



- A single command line to install (NATS v2 auth included)
 - ✓ `curl -sSL https://nats-io.github.io/k8s/setup.sh | sh`
- Stateful Sets (used via installer)
 - ✓ NATS Server / NATS Streaming Server official examples
 - ✓ NATS Operator is also changing to use StatefulSets internally
- Monitoring
 - ✓ Surveyor Installation

Roadmap



Latest	2020-Q3	2020-Q4	2021-Q1 - Q2
<p>Websocket Support</p> <ul style="list-style-type: none">• NATS Server• NATS.ws Client <p>Leafnode Improvements</p> <p>Default User Assignment</p> <p>Monitoring via Services</p> <p>Client Releases and Fixes</p>	<p>JetStream in Core NATS</p> <ul style="list-style-type: none">• Persistent Streaming• Streams and Consumers• Clustering (HA/FT) <p>Message Headers</p> <p>NATS-JMS Bridge</p> <p>Client Service/Stream APIs</p>	<p>Native MQTT Support</p> <ul style="list-style-type: none">• 3.1 <p>Monitoring Enhancements</p> <p>NATS-Kafka Bridge Enhancements</p> <p>Edge to Edge Zero-Trust Security</p>	<p>Native MQTT Support</p> <ul style="list-style-type: none">• 5.0/SN <p>WASM Support in the NATS Ecosystem</p> <p>Additional Ops/Dev Tooling</p> <ul style="list-style-type: none">• No Touch Distributed Tracing• System-wide Debug Tooling



KubeCon



CloudNativeCon

Europe 2020

Virtual

Questions?

More info:

slack.nats.io

nats.io/community

github.com/nats-io

[@nats io](https://twitter.com/nats_io)

<https://nats.io>

info@nats.io



KubeCon



CloudNativeCon

Europe 2020



Virtual



KEEP CLOUD NATIVE

CONNECTED

