



a simple, high performance open source messaging system for cloud native applications, IoT messaging, and microservices architectures

Benjamin Nørgaard

# Me



Benjamin Nørgaard

DevOps Engineer @  
Lix Technologies

<https://www.lix.com>



Enthusiastic about keyboards  
and death metal

<https://github.com/blacksails>

[https://twitter.com/b\\_noergaard](https://twitter.com/b_noergaard)

<https://www.linkedin.com/in/blacksails>



# What is NATS?



Copy-paste from website:

“a simple, high performance open source messaging system for cloud native applications, IoT messaging, and microservices architectures”

Translation with less buzzwords:

a message queue/pubsub comparable to RabbitMQ or Kafka.

# NATS vs NATS Streaming



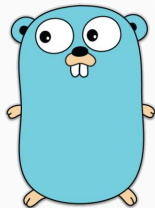
NATS comes in two flavors

- NATS
  - Simple
  - at-most-once delivery
  - PubSub, queuing and request/reply
- NATS Streaming
  - Persistence of messages and subject names (enables replay)
  - at-least-once delivery (acknowledgements)
  - Rate limiting
- NATS Streaming is built on top of NATS
- Both have HA capabilities and are very easy to operate on kubernetes
- I will do a demo of NATS (not streaming)

# Demo time



- Chat application
  - Vue frontend
  - Go backend
  - Websocket for sending chat messages back and forth between the backend and frontend clients



Problem:

We cannot scale the backend because clients connecting to different backends won't see each others' messages.

Solution:

Send messages through a NATS pubsub which will ensure that all backends see all messages.

Code: <https://github.com/blacksails/nchats>

# Questions?

<https://github.com/blacksails>

[https://twitter.com/b\\_noergaard](https://twitter.com/b_noergaard)

<https://www.linkedin.com/in/blacksails>