LINE Bot API with Armeria

Armeria with Service Mesh

Soo Kim, LINE Bot Backend tema

2019.11.18

Contents

01 Armeria with LINE Bot Backend

02 Coexistence with Sidecar Proxy Pattern

Today’s Speaker

Soo Kim

Software Engineer, LINE Bot Backend

LINE Corporation

LINE Corp. Bot Backend team

Mainly Messaging API for Official Accounts

Speaker

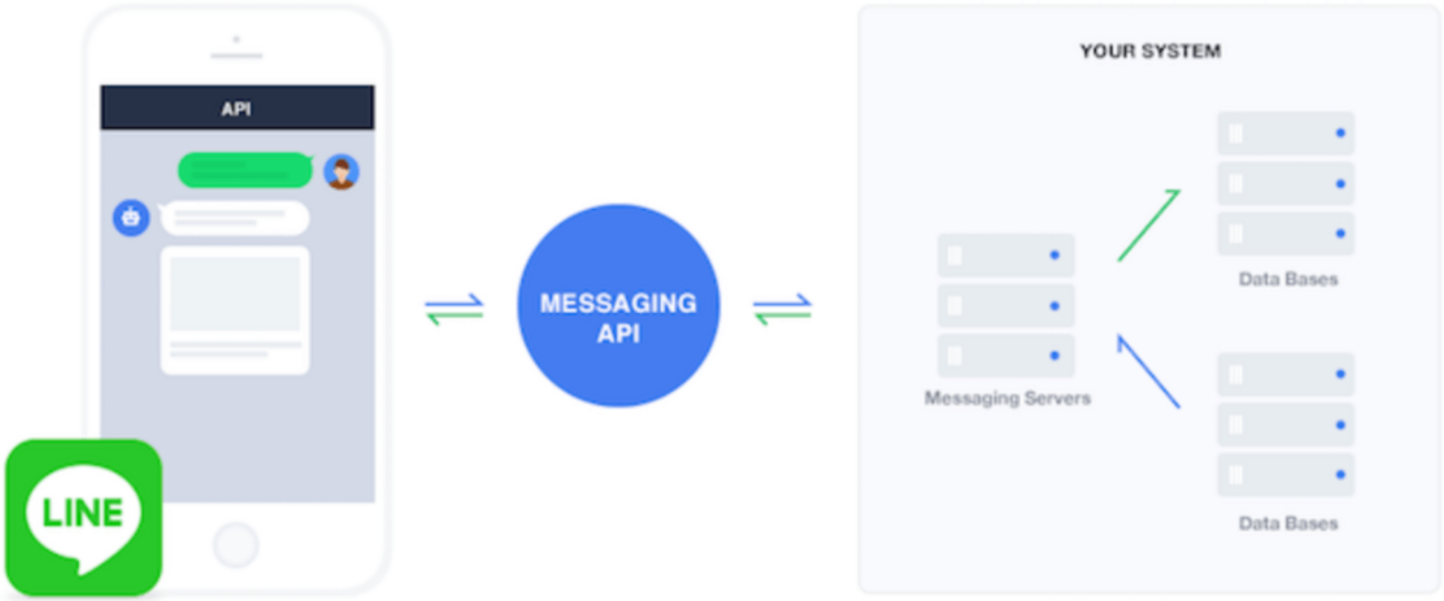
2018.10 ~ (New Grad.) LINE Corp.

Bot Backend Team

Bot API

API for official accounts open to public

https://developers.line.biz/en/docs/messaging-api/



From https://blog.feedmatic.net/about-messaging-api/

🤖 ❤️

Adoption in progress

- The first option with new project
- 3 services fully powered by Armeria (among 10~20 services)
- Many others with partial adoptions (Client for RPC was the easiest first-step)

Combined with diverse stacks

- Java + Armeria + Cassandra
  - Fully async
- Scala + gRPC server + HBase + HDFS
  - HBase only has sync driver, so we isolated thread pool powered by Monix
- Scala + Thrift server + HBase + Redis (in-house version)
- Scala + gRPC server + CentralDogma

- And Many other
  - Java/Kotlin + Spring Boot + HTTP/RPC client

Microservice problems that Armeria solves

Server

- Out of box async server/client
- Mix of gRPC/thrift/http
- Easy RPC with DocService

Client

- Client side Load-balancing
- Service discovery
- Logging
- Metrics
- Distributed tracing
- Automatic retry

※Source from ****** Research Center

Microservice problems that Armeria solves

Overlaps with Service Mesh with sidecar proxy

Server

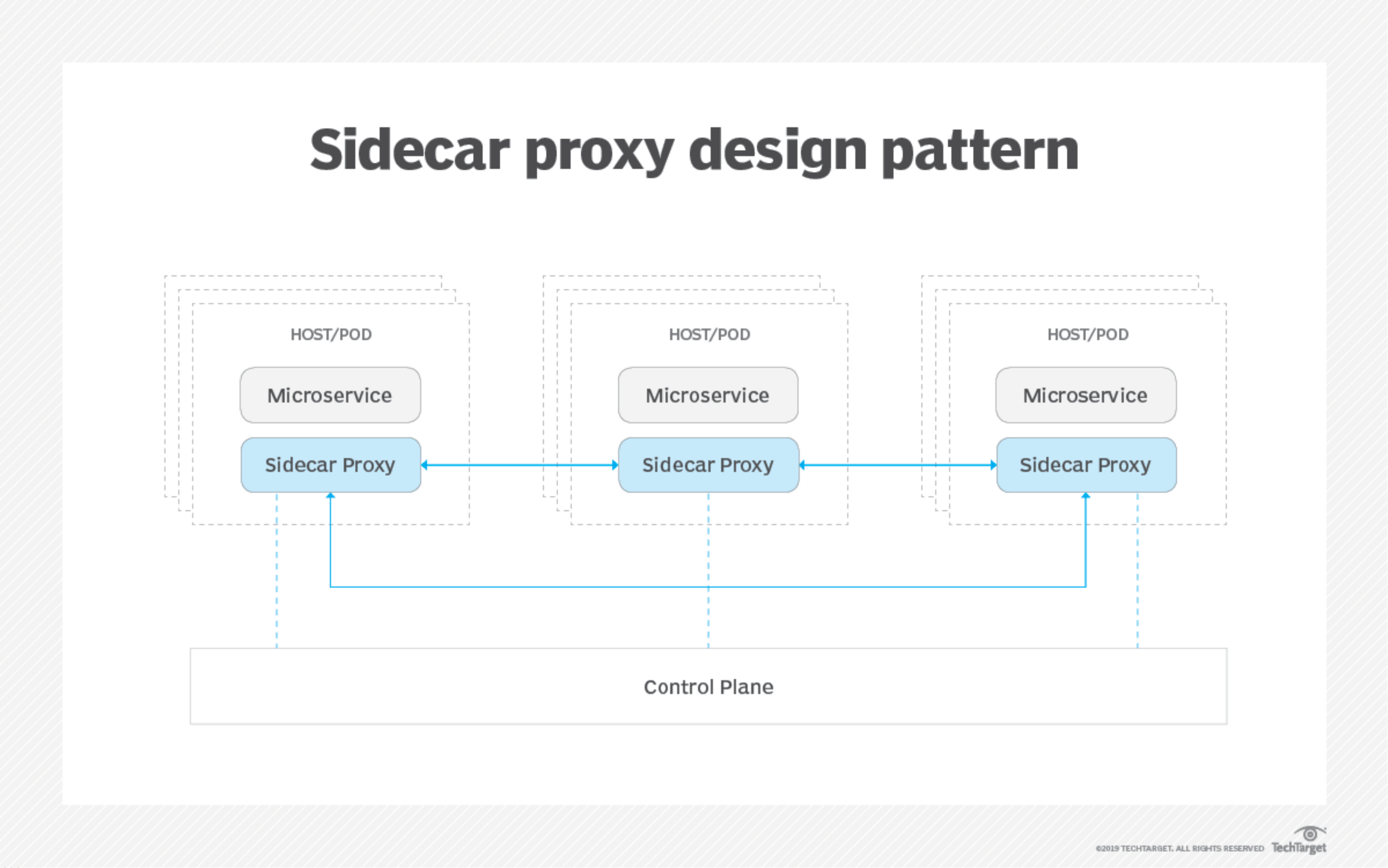
- Out of box async server/client
- Mix of gRPC/thrift/http
- Easy RPC with DocService

Client

- Client side Load-balancing
- Service discovery
- Logging
- Metrics
- Distributed tracing
- Automatic retry

※Source from ****** Research Center

What is Service Mesh



From https://searchitoperations.techtarget.com/definition/sidecar-proxy

Shared Library vs Sidecar Proxy

Pros and cons

Shared Library

- Sacrifice polyglot development
  - e.g. Only JVM for Armeria
- Harder to deploy and manage library version

Sidecar Proxy

- Application code ignorant about service mesh
- No dependency with application, so easy to maintain versions
  - No Netty dependency collision

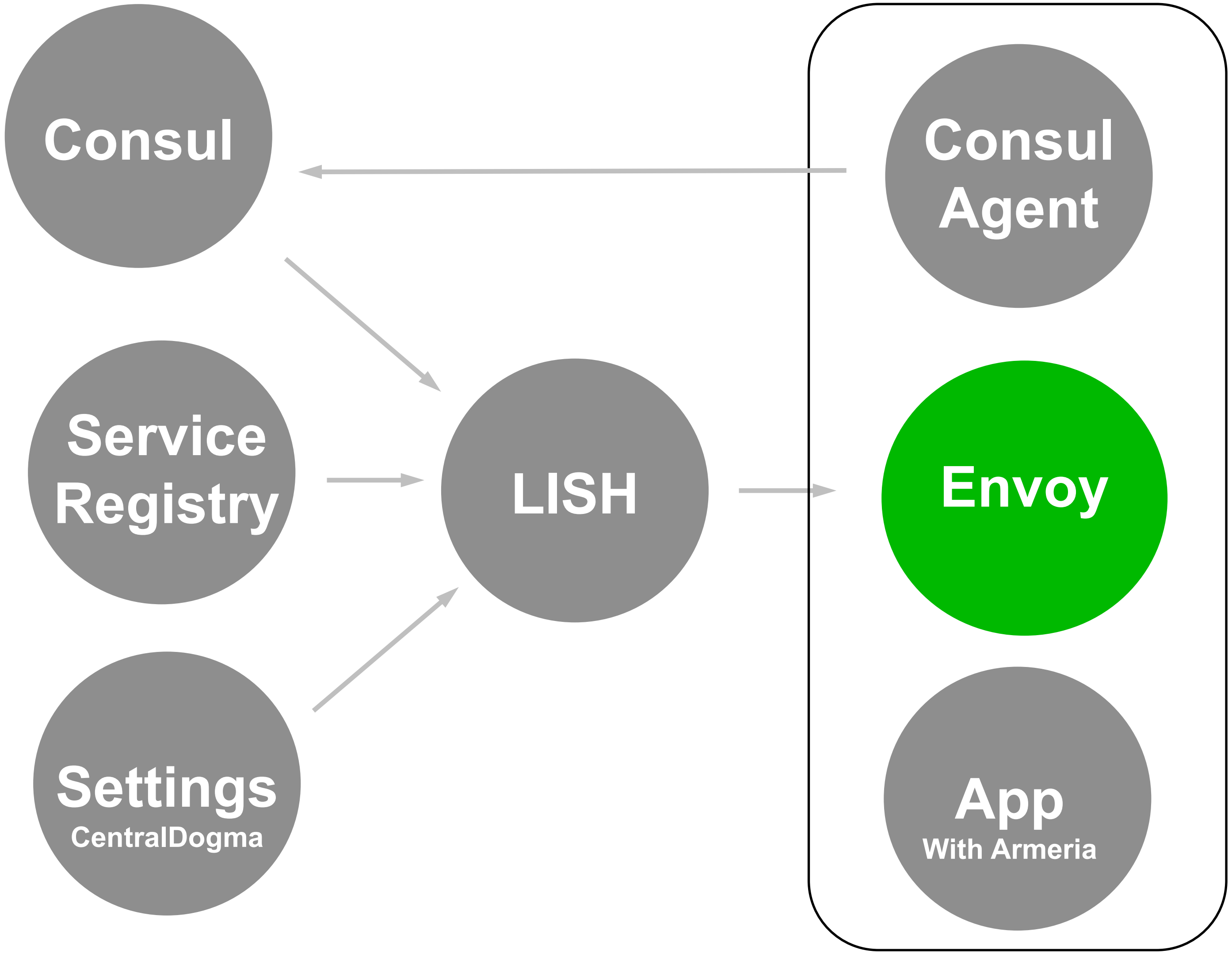
Service mesh of our team

LISH: Home-made Control Plane

2019 DevDay Lightening talk by Fukumuro-san

- https://linedevday.linecorp.com/jp/2019/sessions/S1-01

Service mesh of our team



FAQ

Questions and answers

THANK YOU