



SERVICE MESH FOR BETTER DEVOPS

IDIT LEVINE, SOLO.IO

TRANSFORMATION JOURNEY



MIGRATION



SERVICE MESH



& BEYOND

INNOVATION



MIGRATION

HUGE IMPROVEMENT ...



PRODUCTIVITY

improved productivity and speed.



DESIGN

a convention that forces good software design.



ENCAPSULATION

encapsulation of knowledge.



FLEXIBLE

flexible span of technologies and scalability

NEW PROBLEMS ...



COMPLEXITY

more dependencies and complexities.



PERFORMANCE

network latency and others performance issues.



COST

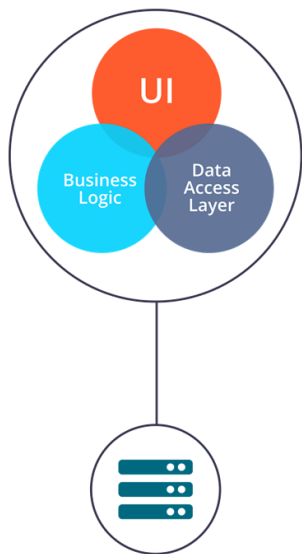
more complex and costly deployments.



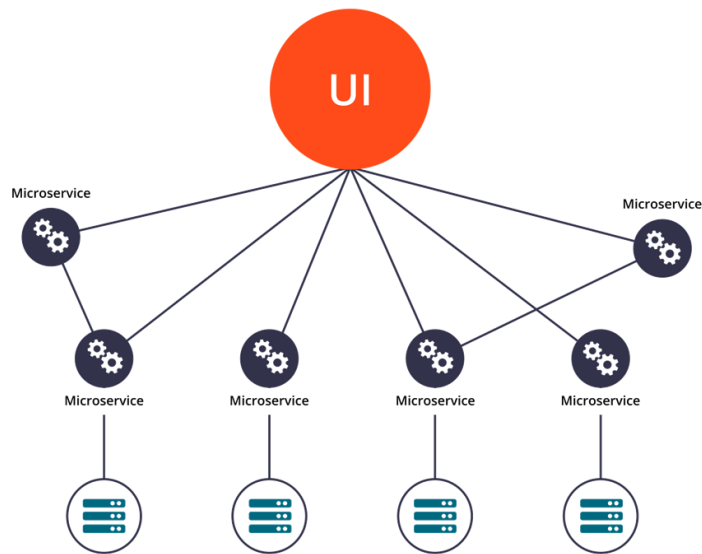
TESTING

end to end testing is more difficult.

MIGRATION



Monolithic Architecture



Microservice Architecture

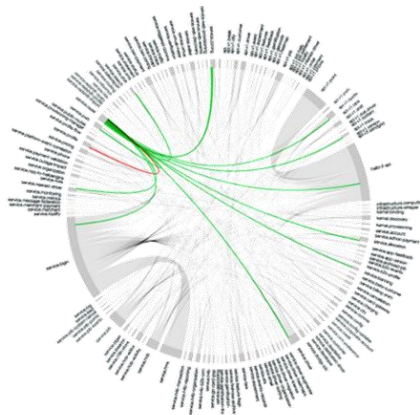
**A MONOLITHIC APPLICATION
CONSISTS OF A SINGLE
PROCESS**

**A MICROSERVICES APPLICATION
CONSISTS OF POTENTIALLY
HUNDREDS OF PROCESSES**

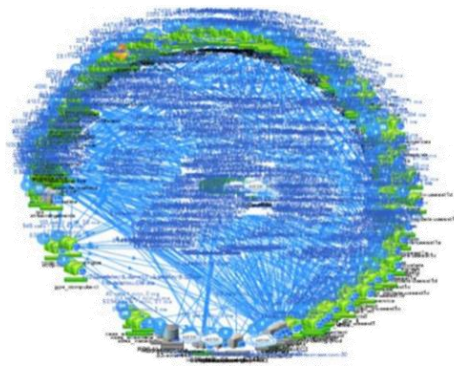
Is it possible to get a
complete view of the state of
a such application?!

MIGRATION

450+ microservices

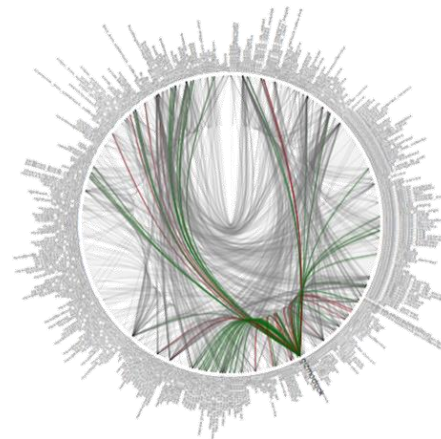


500+ microservices



NETFLIX

500+ microservices



THE PROBLEM



UNIFIED TOOLBOX

MONOLITHIC APPS

MICROSERVICES

METRICS

APM

PROMETHEUS

LOGGING

SPLUNK

OPEN TELEMETRY

DEBUG

NATIVE DEBUGGERS

ARCH

SOA

MICROSERVICES

DEPLOYMENT

PUPPET/CHEF/ANSIBLE

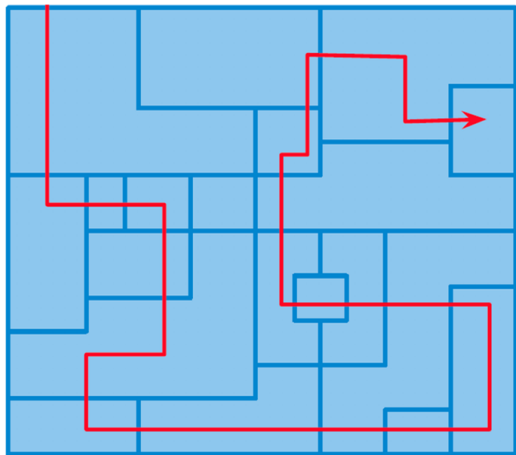
HELM

TESTING

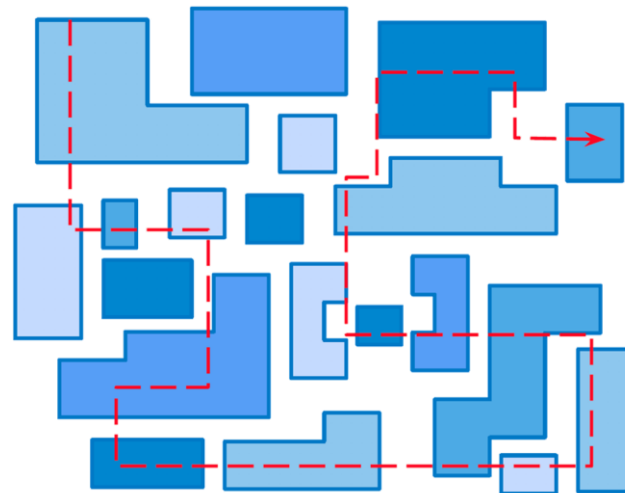
80% UNIT / 20% INTEGRATION

20% UNIT / 80% INTEGRATION

THE PROBLEM



MONOLITHIC



MICROSERVICES



HOW DO YOU MAKE SURE
MICROSERVICES CAN
COMMUNICATE?



HOW CAN YOU MAKE
SURE THEY CAN
COMMUNICATE SAFELY?



HOW CAN YOU SEE
WHAT IS HAPPENING
INSIDE THE CLUSTER?



SERVICE MESH

SERVICE MESH USE CASES

SECURITY

- ROOT CERT
- MTLS
- POLICY



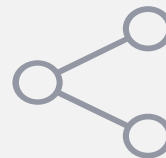
OBSERVABILITY

- METRICS
- LOGS

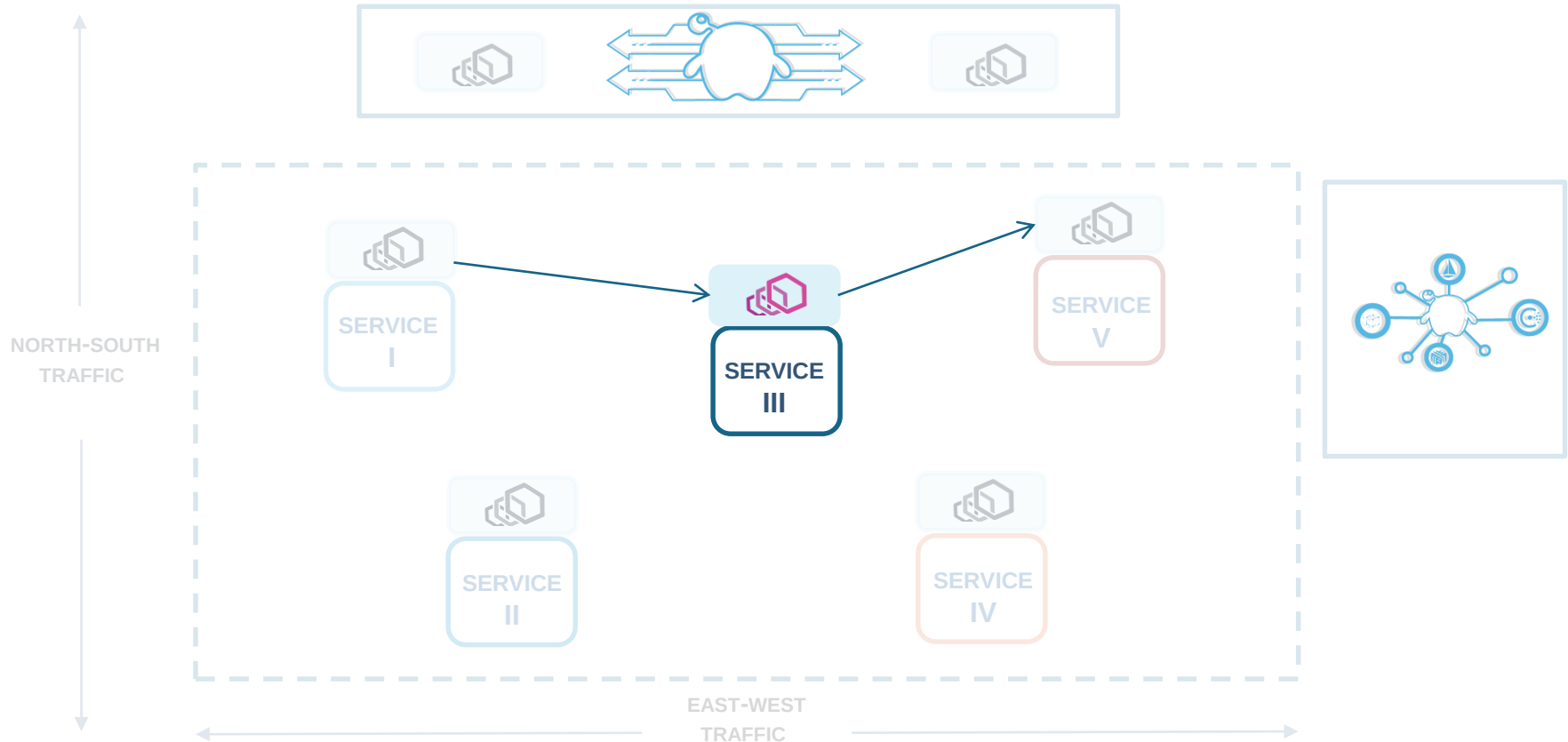


ROUTING

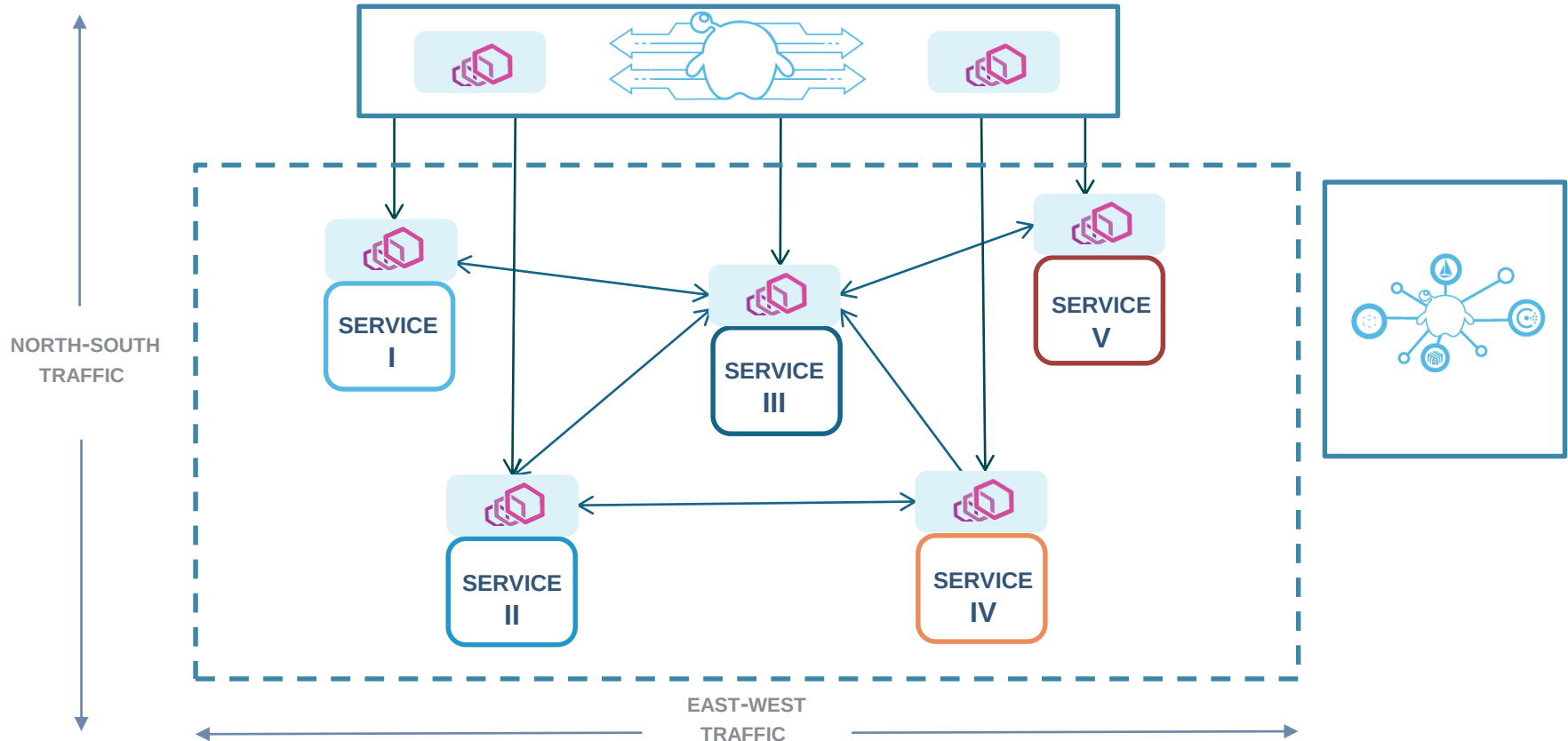
- TRAFFIC CONTROL
- RESILIENCE



NETWORK ABSTRACTION

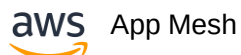


NETWORK ABSTRACTION



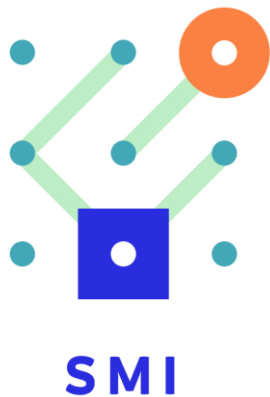
WHICH SERVICE MESH SHOULD I USED?

ABSTRACTION

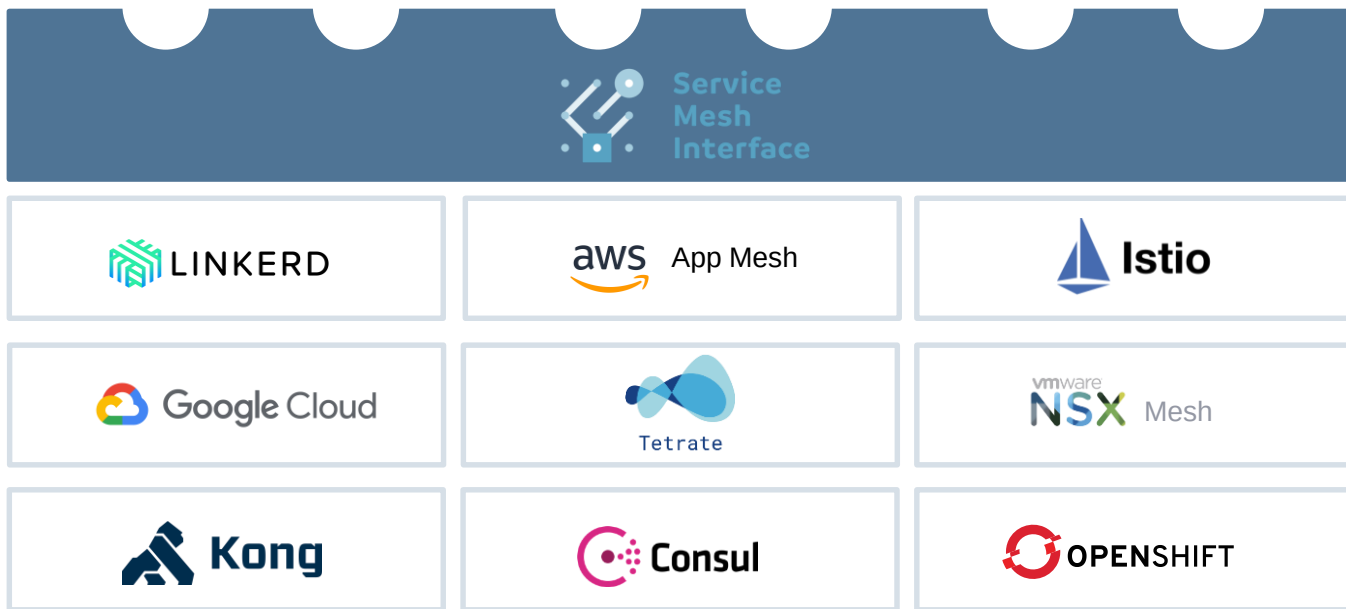


- DEAD SIMPLE API:
SIMPLER WORKFLOW,
MIGRATION,
PREVENTS VENDOR-
LOCKING
- FLAT NETWORK
ACROSS DIFFERENT
MESHERS
- SINGLE PANE FOR
ALL MESHERS

SERVICE MESH INTERFACE

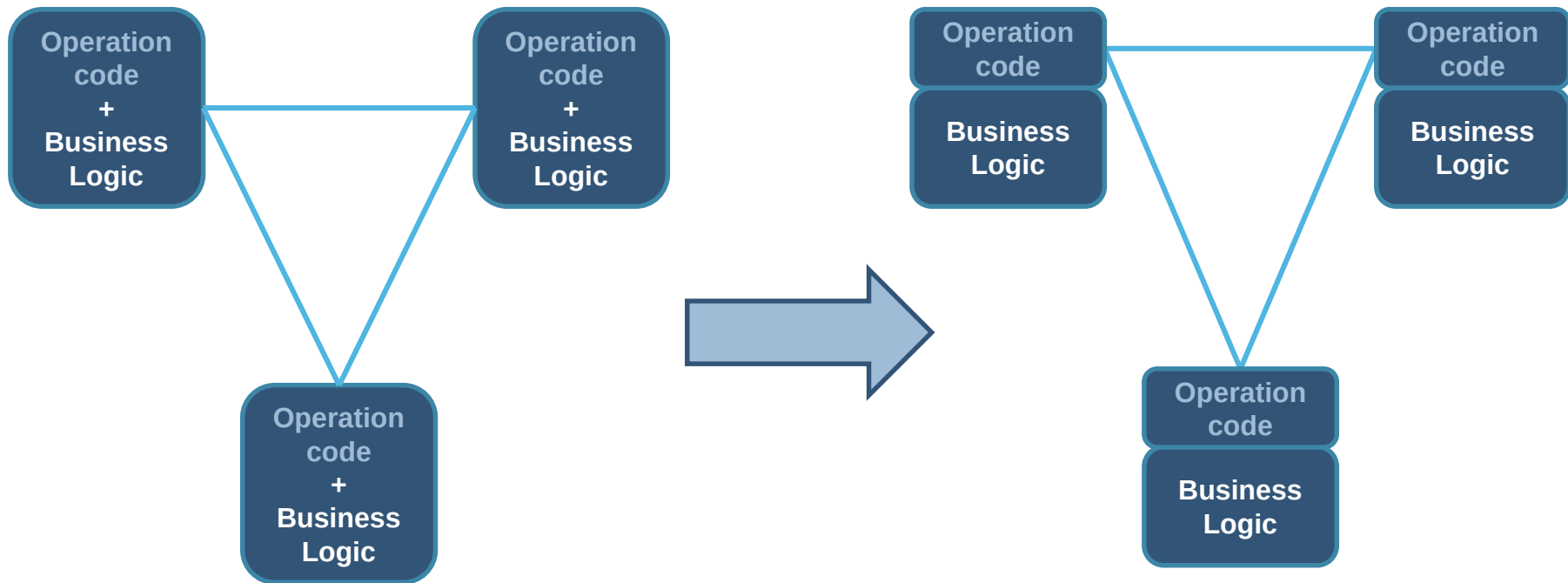


SERVICE MESH INTERFACE (SMI)

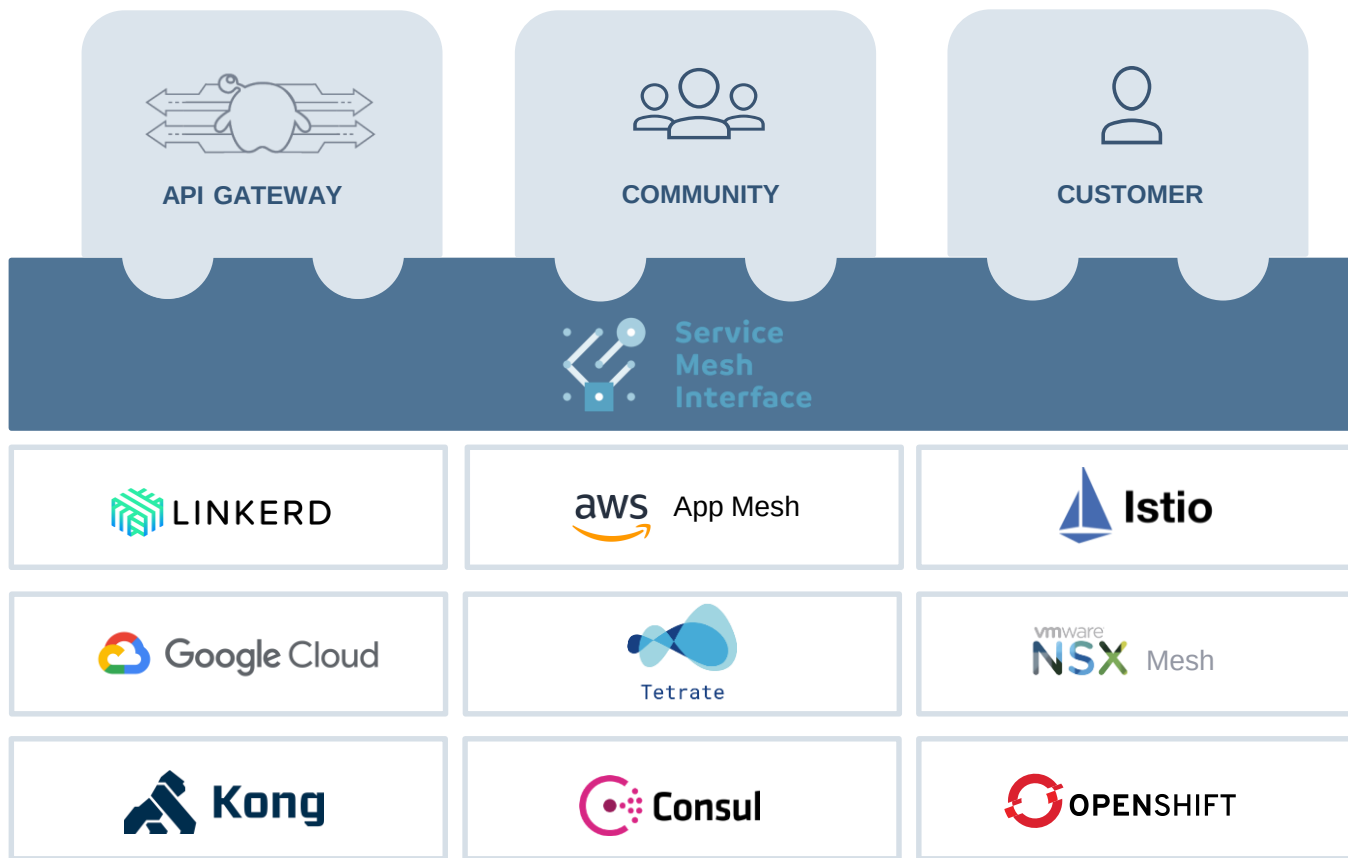


- DEAD SIMPLE API:
SIMPLER WORKFLOW,
MIGRATION,
PREVENTS VENDOR-
LOCKING
- FLAT NETWORK
ACROSS DIFFERENT
MESHERS
- SINGLE PANE FOR
ALL MESHERS

SERVICE MESH



SERVICE MESH & BEYOND



- DEAD SIMPLE API: SIMPLER WORKFLOW, MIGRATION, PREVENTS VENDOR-LOCKING
- FLAT NETWORK ACROSS DIFFERENT MESHES
- SINGLE PANE FOR ALL MESHES

Installed Meshes

+ Install a new Mesh



Linkerd

Namespace: linkerd

Version: 2.3.0

Mesh Health ●

Installed Extensions



gloo
Version:
0.13.26

Installed Services

linkerd-linkerd-
controller-api-8085
linkerd

linkerd-linkerd-
controller-api-
linkerd-linkerd-
controller-8085
linkerd

linkerd-linkerd-
destination-8086
linkerd

+ 29 more services

[View Mesh Details](#)



My-Aws-Mesh

Mesh Health ●

Installed Extensions



flagger
Version:
0.12.0

No Installed Services

[View Mesh Details](#)



LEARN MORE:

> SERVICE MESH INTERFACE (SMI)

[HTTPS://SMI-SPEC.IO](https://SMI-SPEC.IO)

> SERVICE MESH HUB

[HTTPS://SERVICEMESHHUB.IO](https://SERVICEMESHHUB.IO)

> CHAOS DEBUGGING

[HTTPS://WWW.YOUTUBE.COM/WATCH?V=JKCFFr8LLTA&t=3s](https://www.youtube.com/watch?v=JKCFFr8LLTA&t=3s)



@IDIT_LEVINE