Increasing Delivery Velocity with a Service Mesh at Indeed

Joshua Shanks

Senior Software Engineer, Indeed

Indeed is the #1 job site worldwide

We help people get jobs.

what

Software Engineer

where

Seattle, WA

Find Jobs

60 countries

30 languages

200M unique visitors

20M jobs

About me

Agenda

- What is delivery velocity
- Our motivations
- Service mesh features
- How it helps
- Your options

Delivery Velocity

Delivery Velocity

- Automated Service Creation

Delivery Velocity

- Automated Service Creation
- Self Service VM provisioning

Delivery Velocity

- Automated Service Creation
- Self Service VM provisioning
- Self Service DB provisioning and migration

Delivery Velocity

- Automated Service Creation
- Self Service VM provisioning
- Self Service DB provisioning and migration
- Continuous Deployment & Integration

Where We Were

Where We Were

- Proprietary

Where We Were

- Proprietary
- Java

Where We Were

- Proprietary
- Java
- Data center Local

Where We Were

- Proprietary
- Java
- Data center Local
- Low Latency

Where We Were

- Proprietary
- Java
- Data center Local
- Low Latency
- 1 request = 1 connection

Where We Wanted To Be

- Open source

Where We Wanted To Be

- Open source
- Language agnostic

Where We Wanted To Be

- Open source
- Language agnostic
- gRPC, HTTP2, REST

Where We Wanted To Be

- Open source
- Language agnostic
- gRPC, HTTP2, REST
- Consul integration

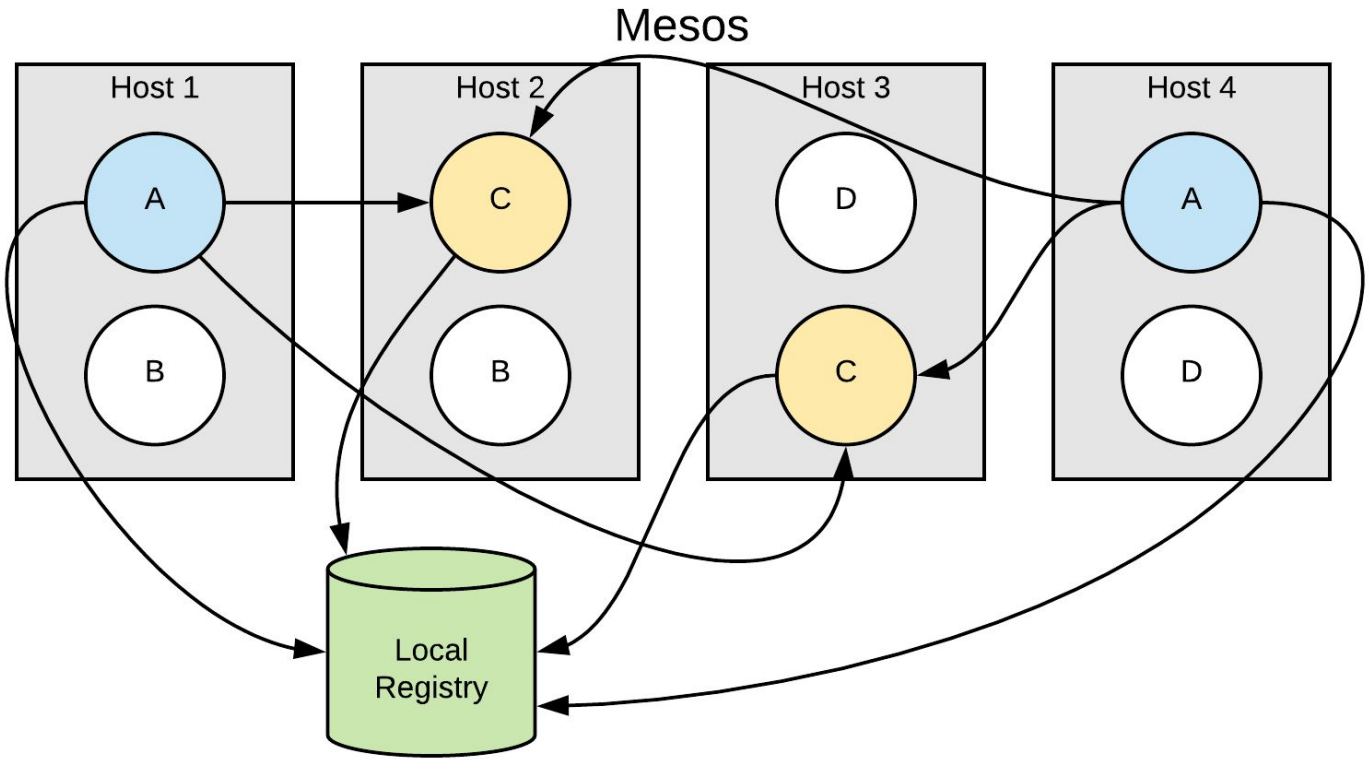
Where We Wanted To Be

- Open source
- Language agnostic
- gRPC, HTTP2, REST
- Consul integration
- Cross data center

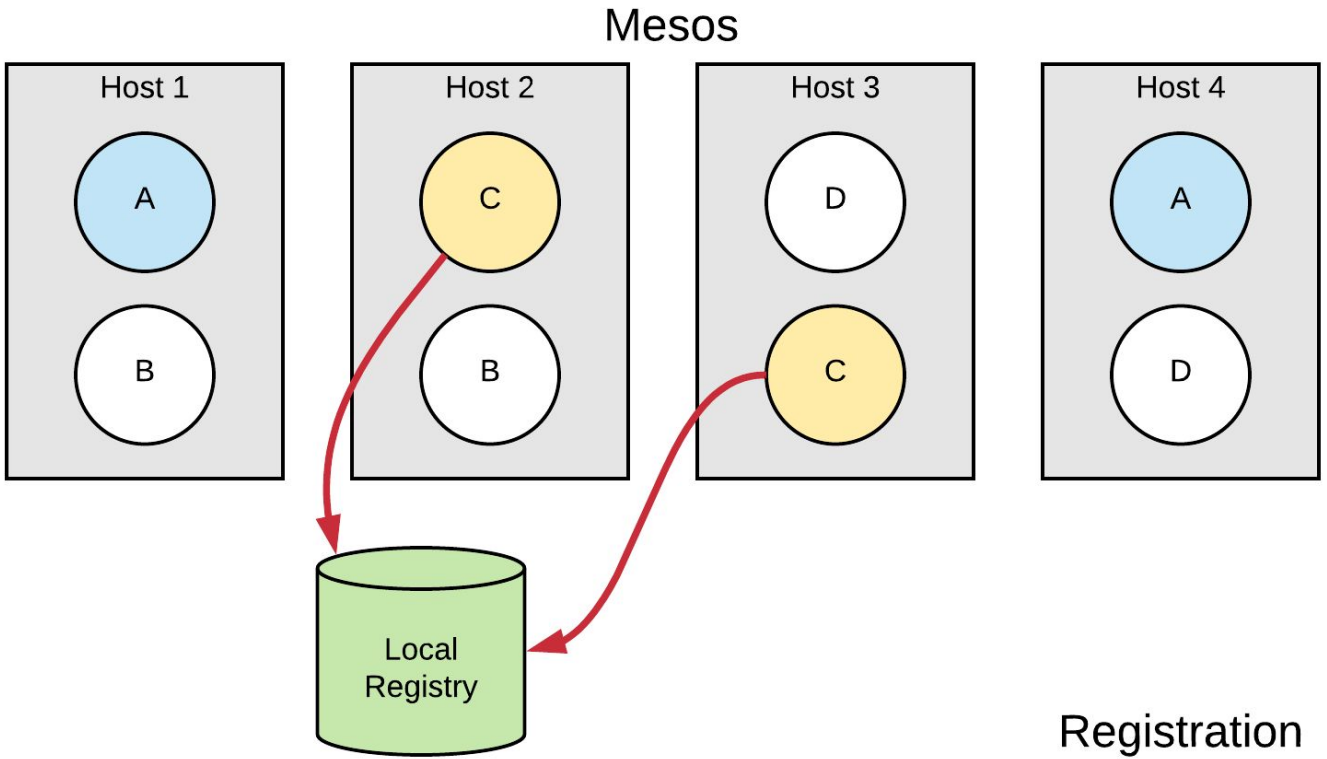
Where We Wanted To Be

- Open source
- Language agnostic
- gRPC, HTTP2, REST
- Consul integration
- Cross data center

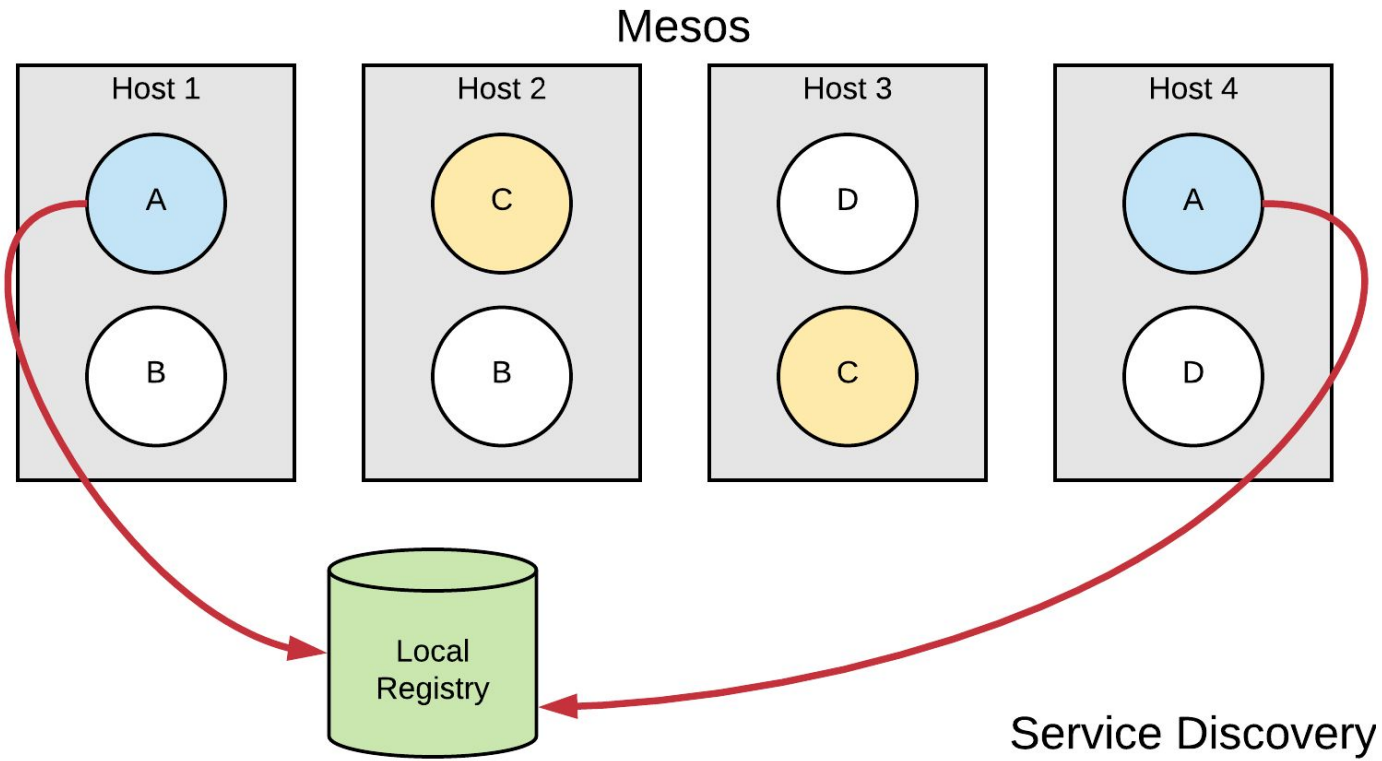
Where We Are



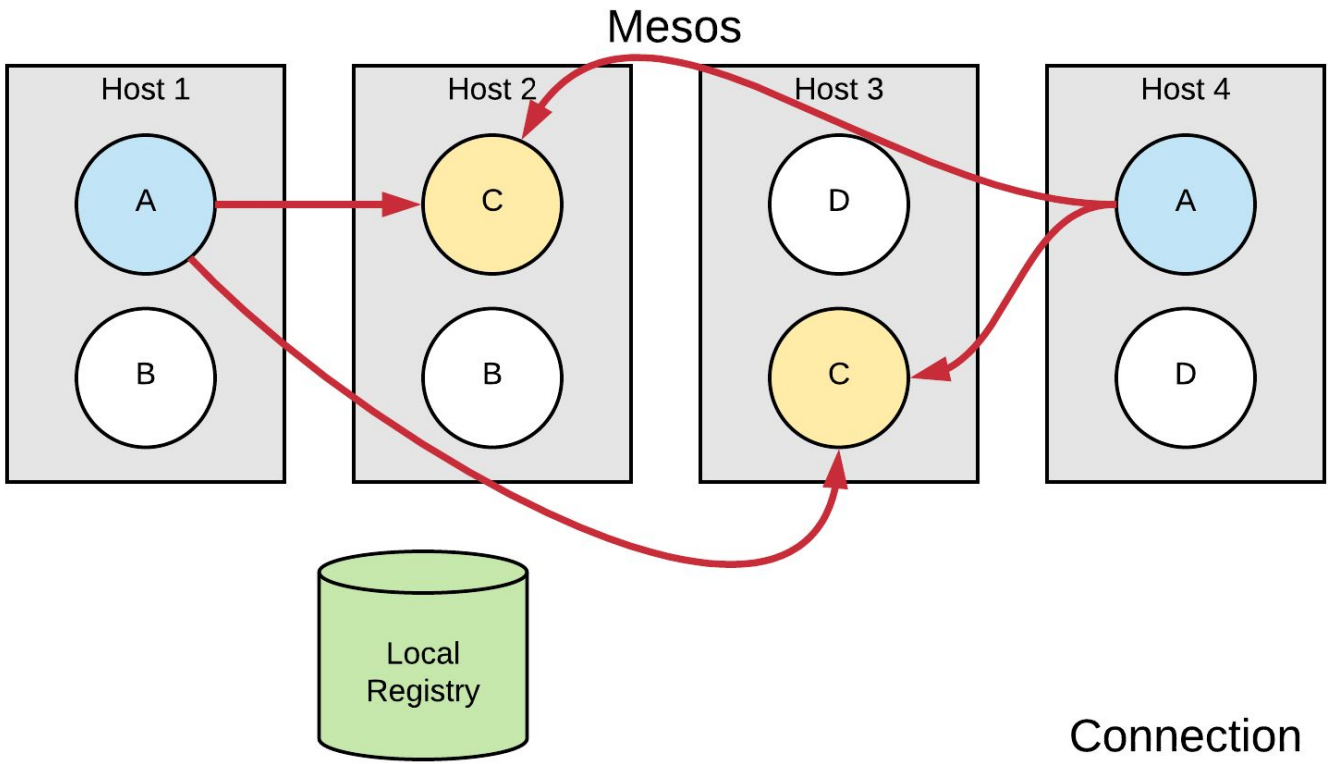
Where We Are


Mesos
Host 1
A
B
Host 2
C
B
Host 3
D
C
Host 4
A
D
Local Registry
Registration

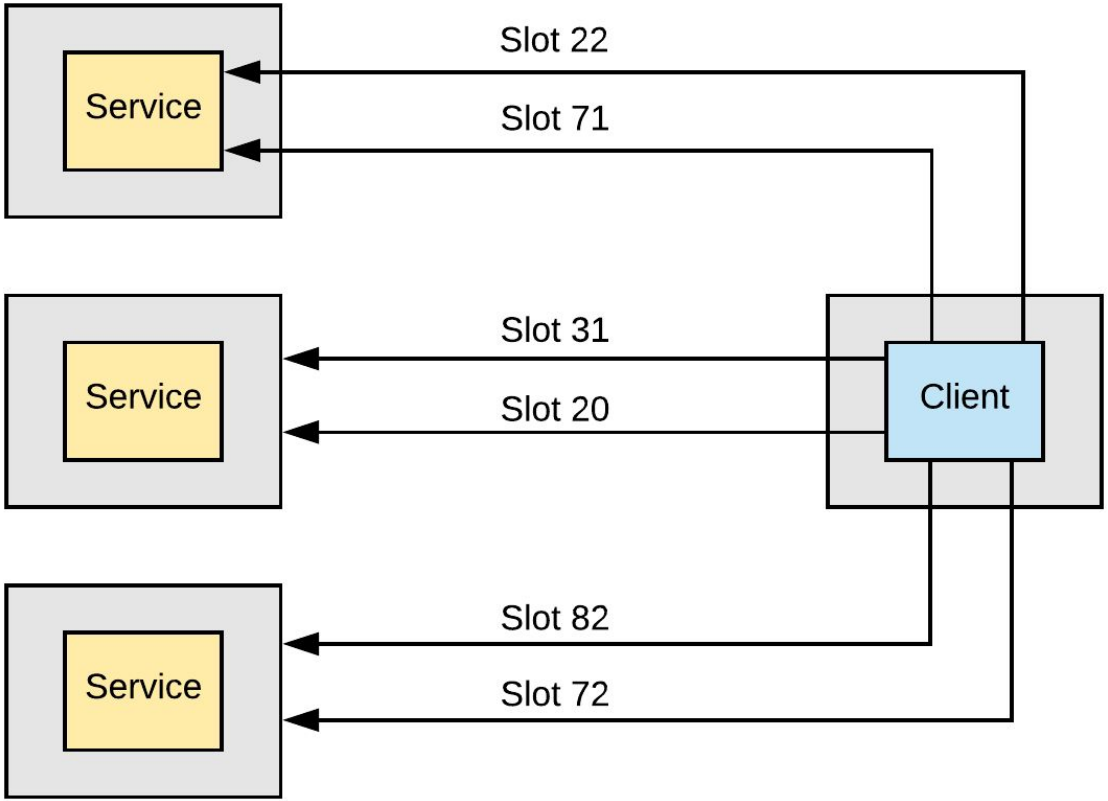
Where We Are


Mesos
Host 1
A
B
Host 2
C
B
Host 3
D
C
Host 4
A
D
Local
Registry
Service Discovery

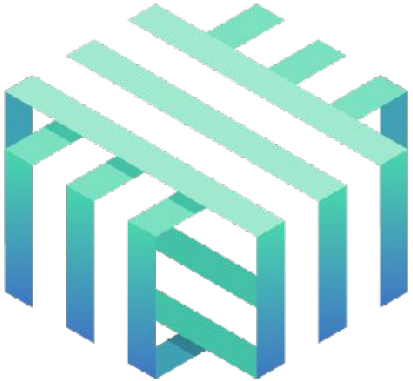
Where We Are


Mesos
Host 1
Host 2
Host 3
Host 4
A
B
C
B
D
C
A
D
Local
Registry
Connection

Where We Were

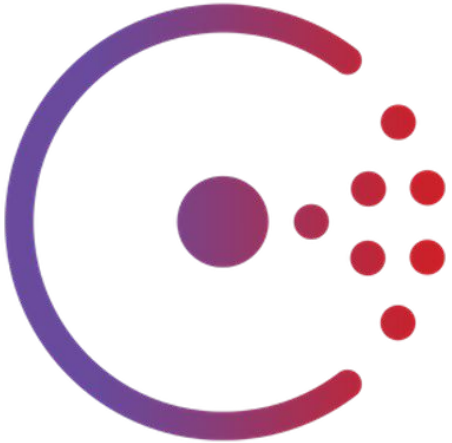


Service Mesh



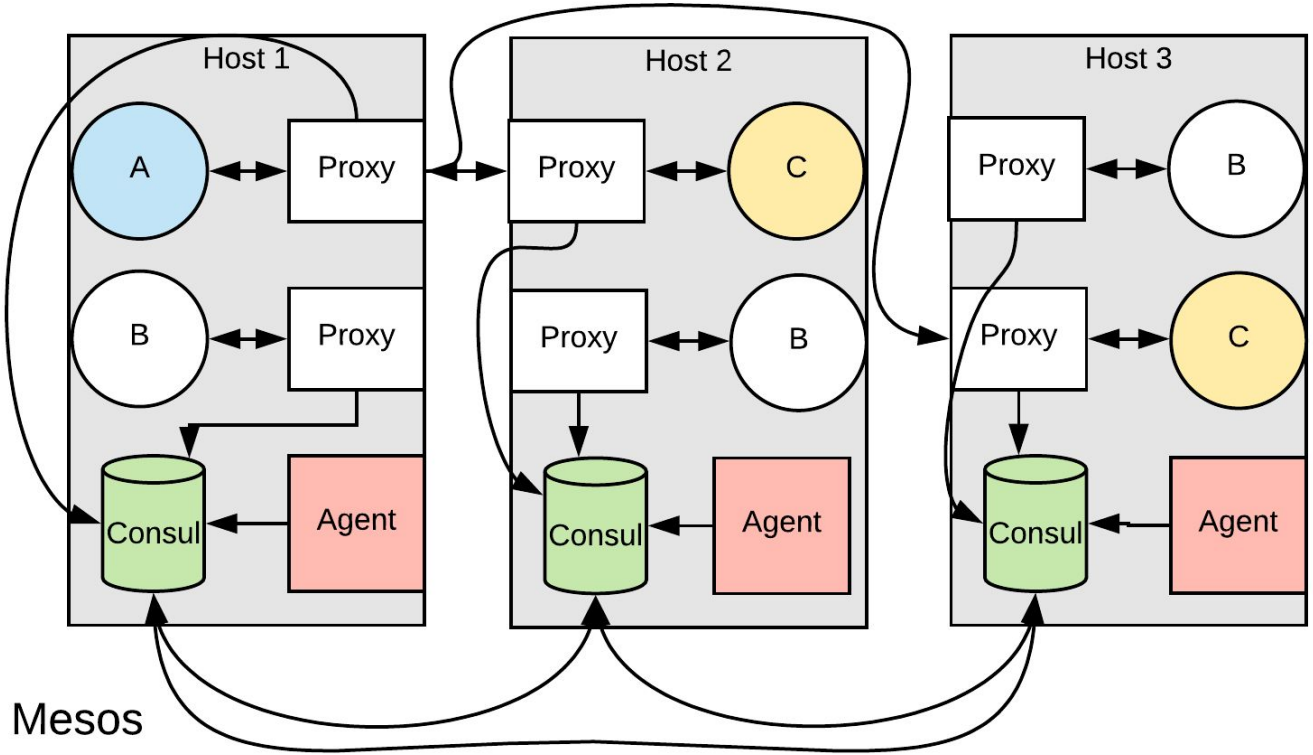
Linkerd

+

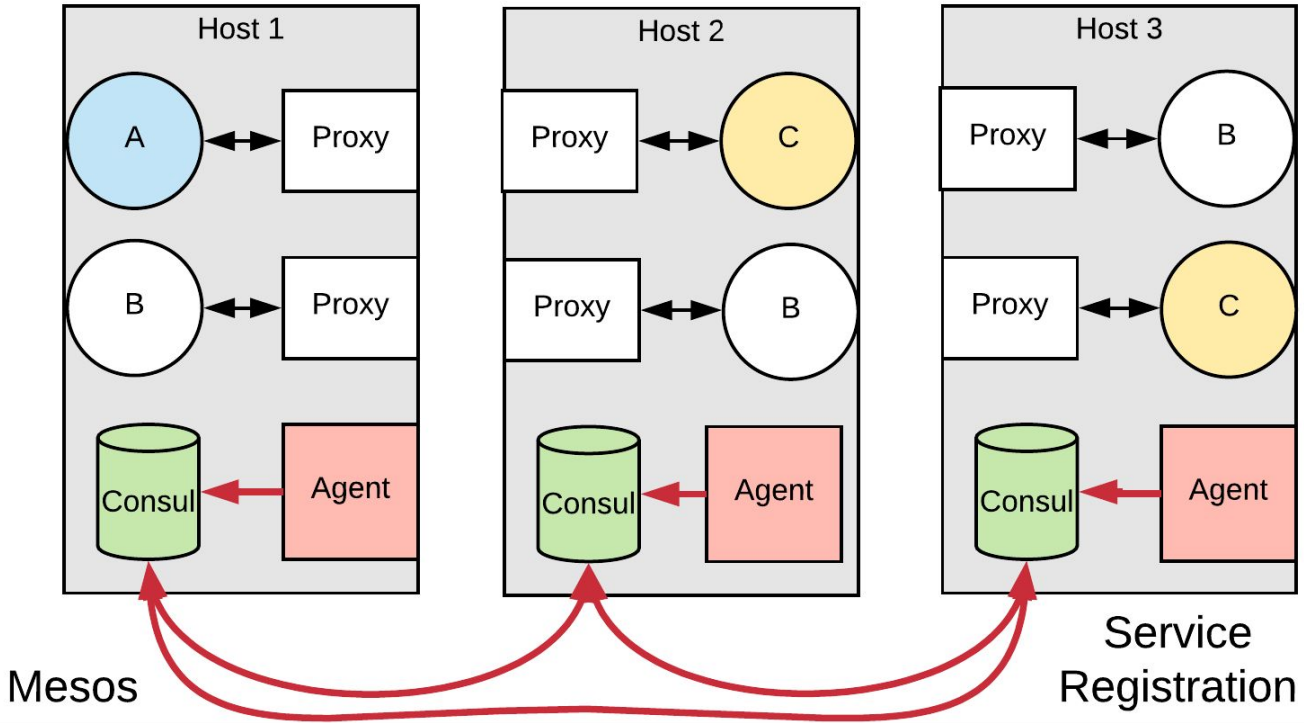


Consul

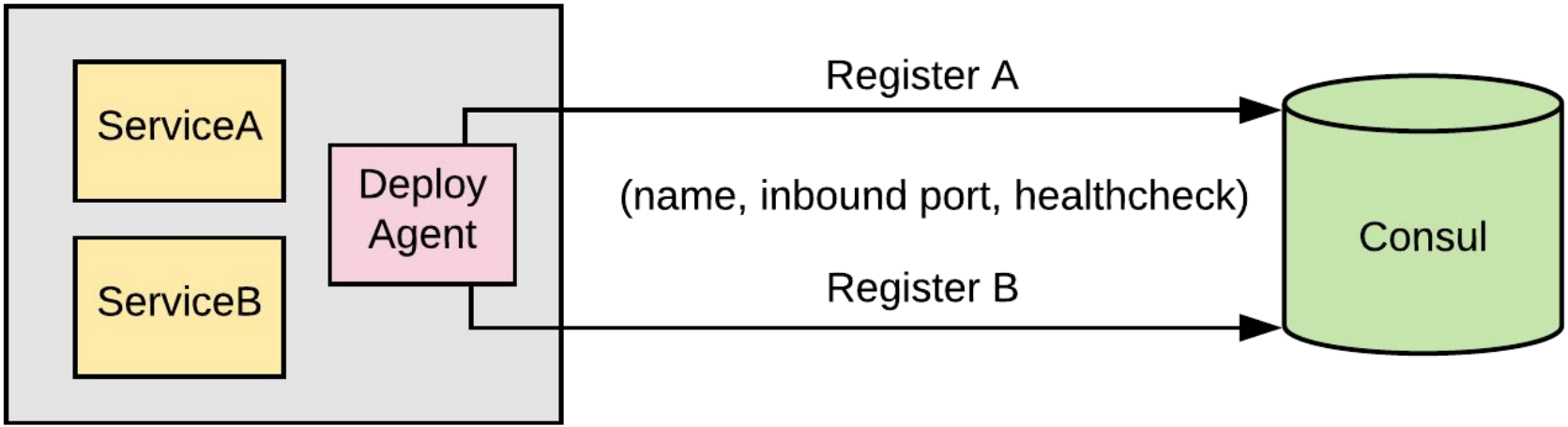
Big Picture


Host 1
Host 2
Host 3
A
B
C
Proxy
Proxy
Proxy
Proxy
Proxy
Proxy
B
C
B
Consul
Consul
Consul
Agent
Agent
Agent
Mesos

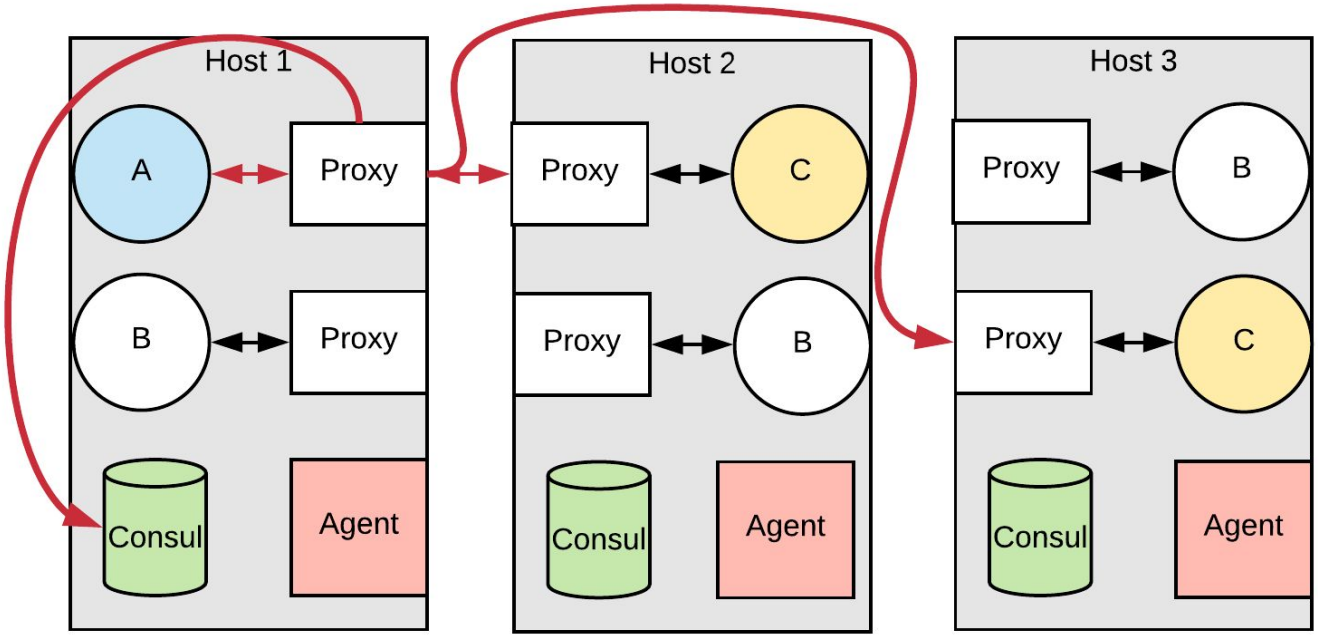
Big Picture


Host 1
A
Proxy
B
Proxy
Consul
Agent
Host 2
Proxy
C
Proxy
B
Consul
Agent
Host 3
Proxy
B
Proxy
C
Consul
Agent
Mesos
Service
Registration

Service Registration



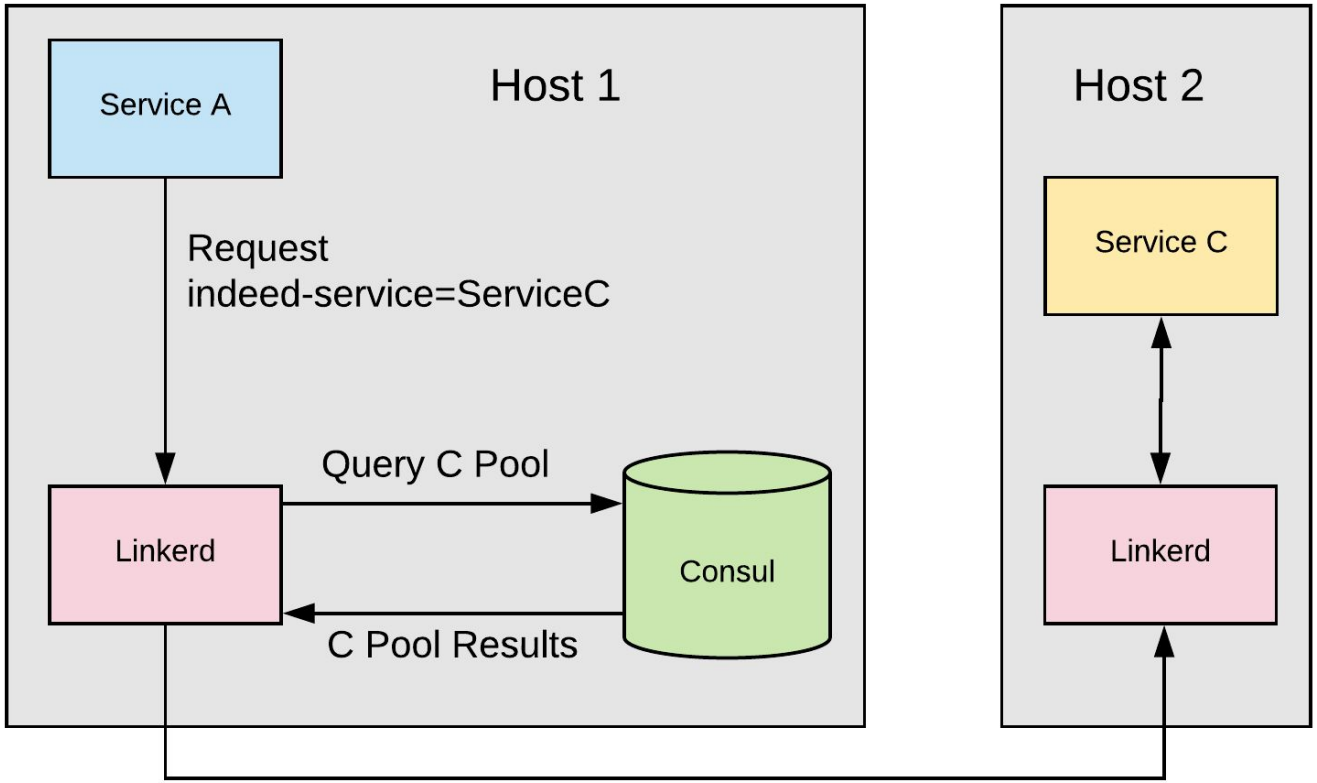
Big Picture



Mesos

Service Discovery

Service Discovery

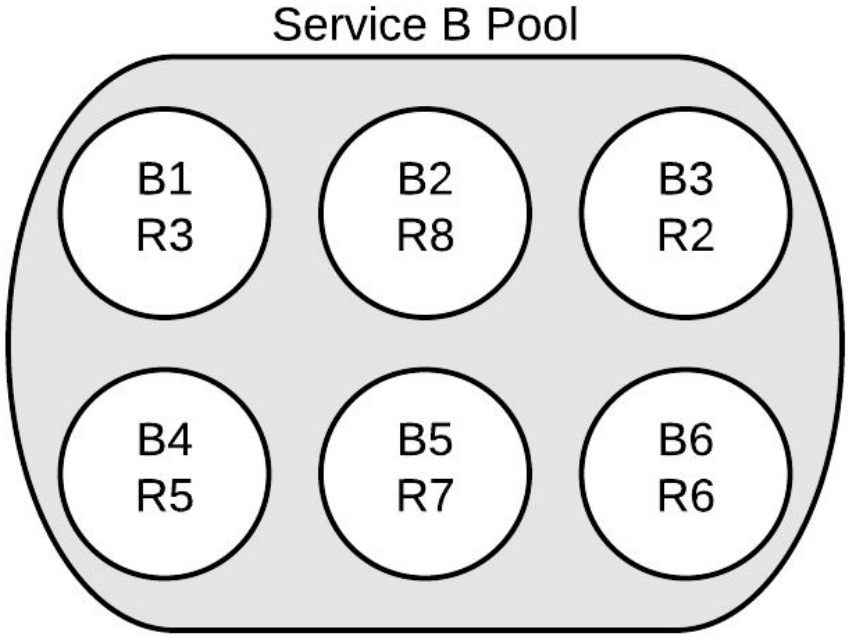


Classifiers

- Error
  - HTTP 5XX
  - gRPC non-zero
- Retryable
  - HTTP GET, HEAD, OPTIONS, TRACE
  - gRPC UNAVAILABLE (14)

Load Balancing

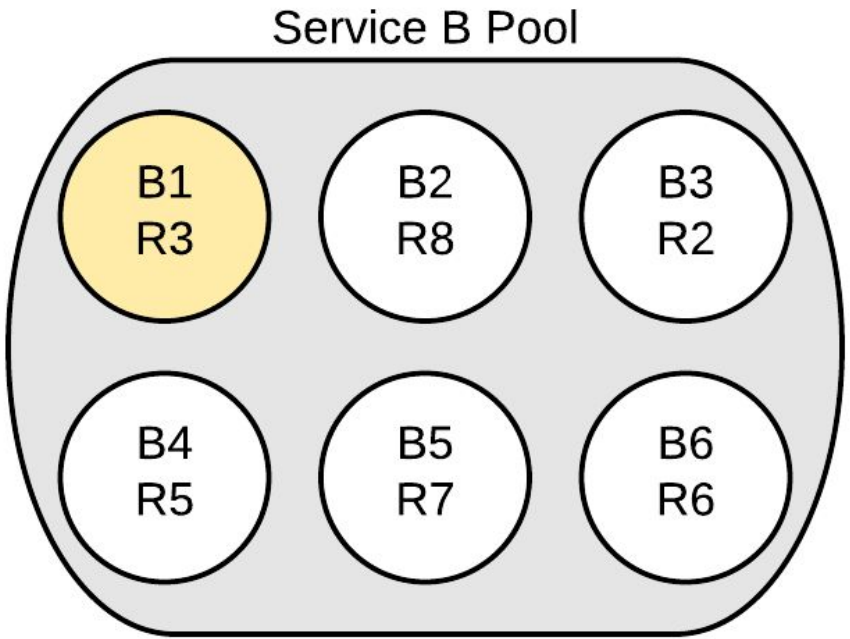
Power of Two Choices: Least Loaded



Pool Size == 6

Load Balancing

Power of Two Choices: Least Loaded

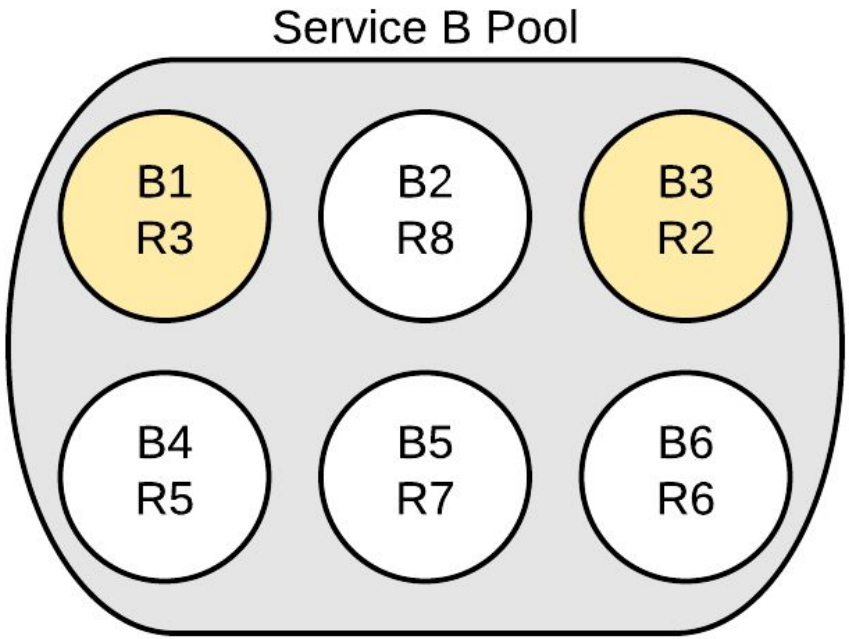


Pool Size == 6

RNG(6) => 1

Load Balancing

Power of Two Choices: Least Loaded

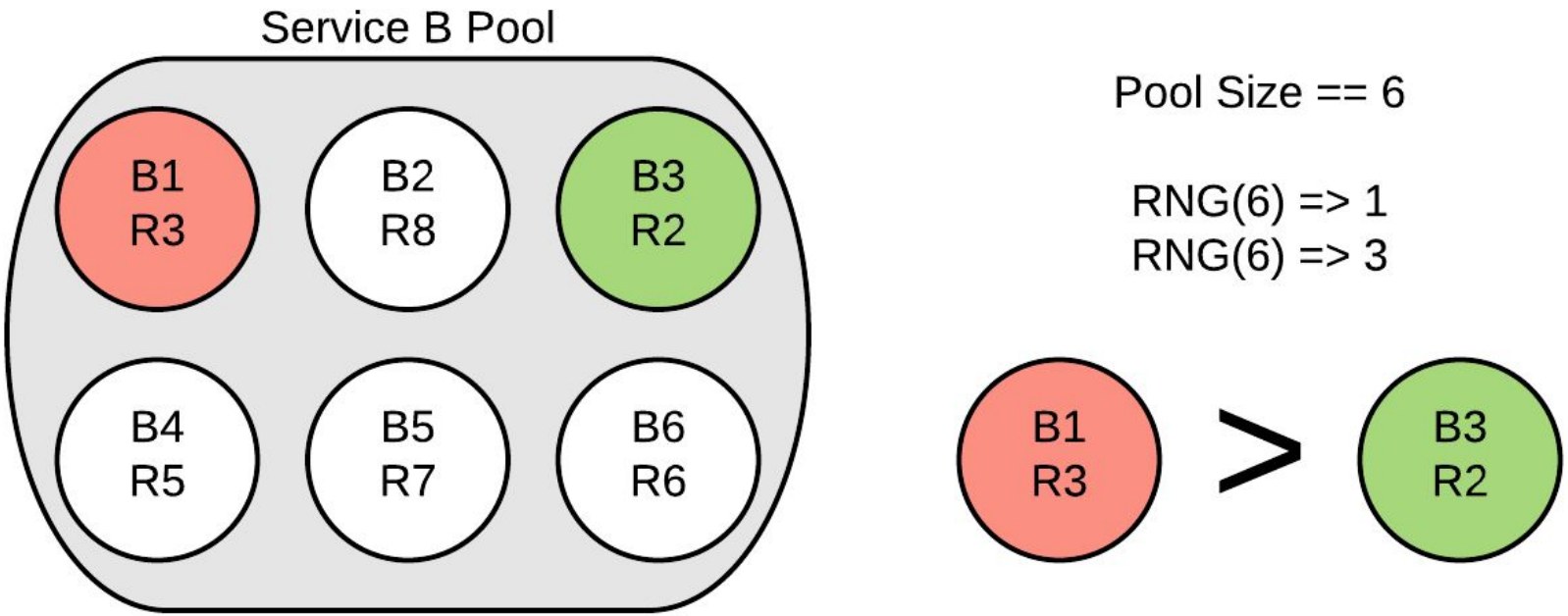


Pool Size == 6

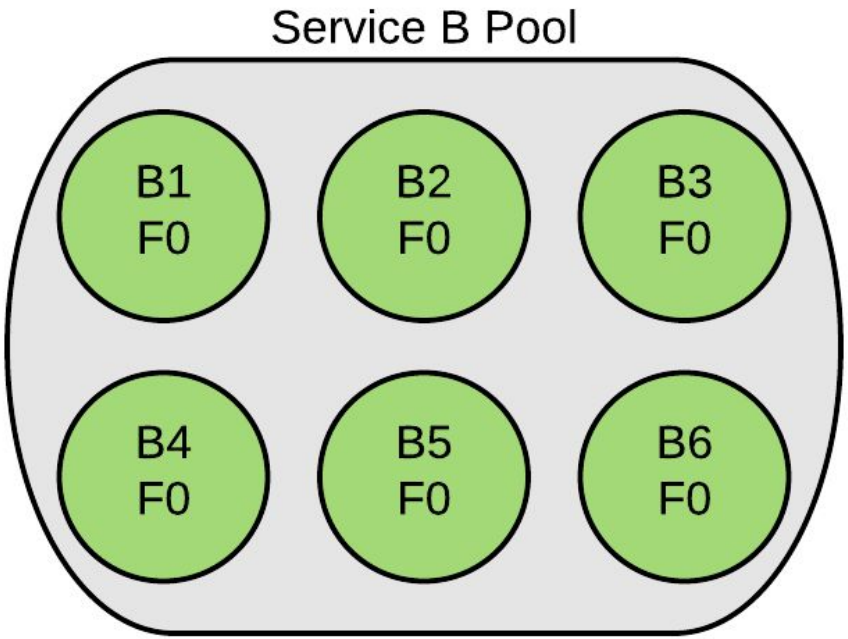
RNG(6) => 1
RNG(6) => 3

Load Balancing

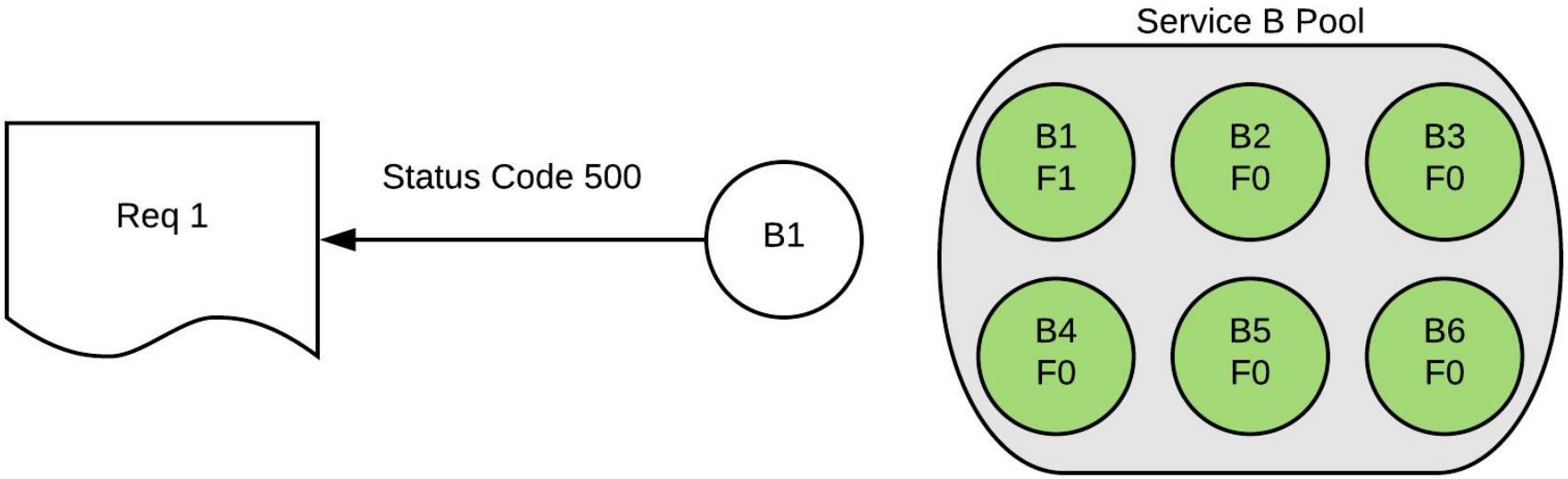
Power of Two Choices: Least Loaded



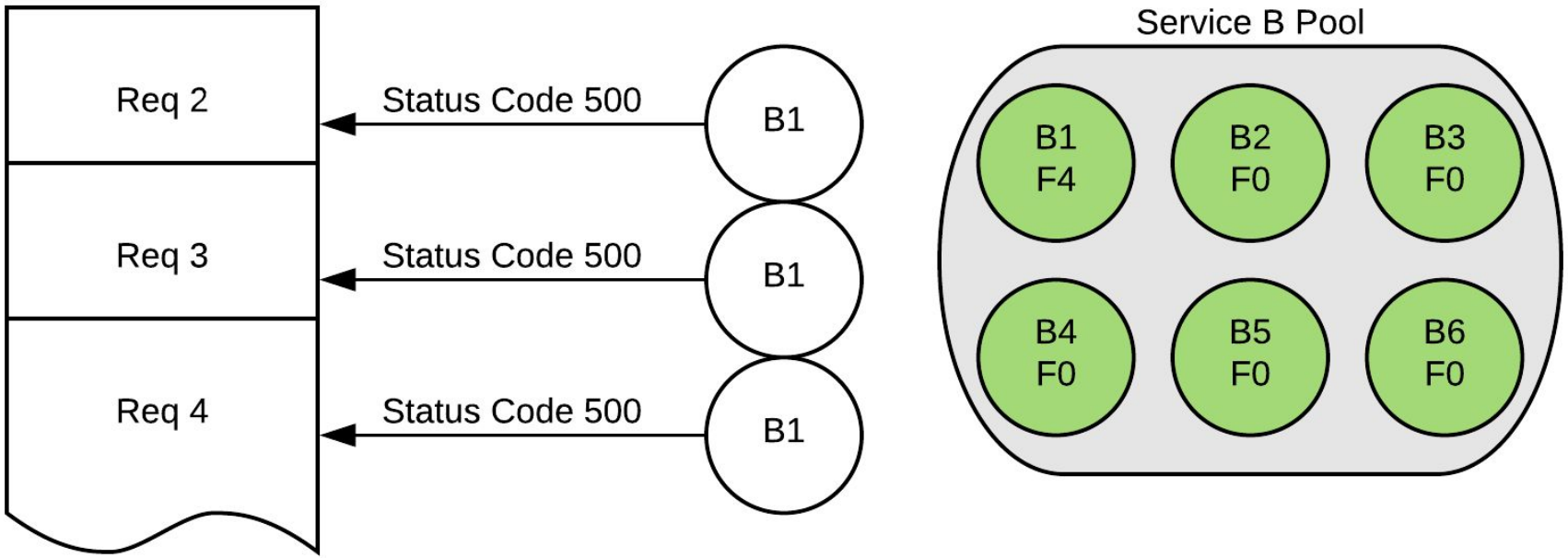
Circuit Breaking



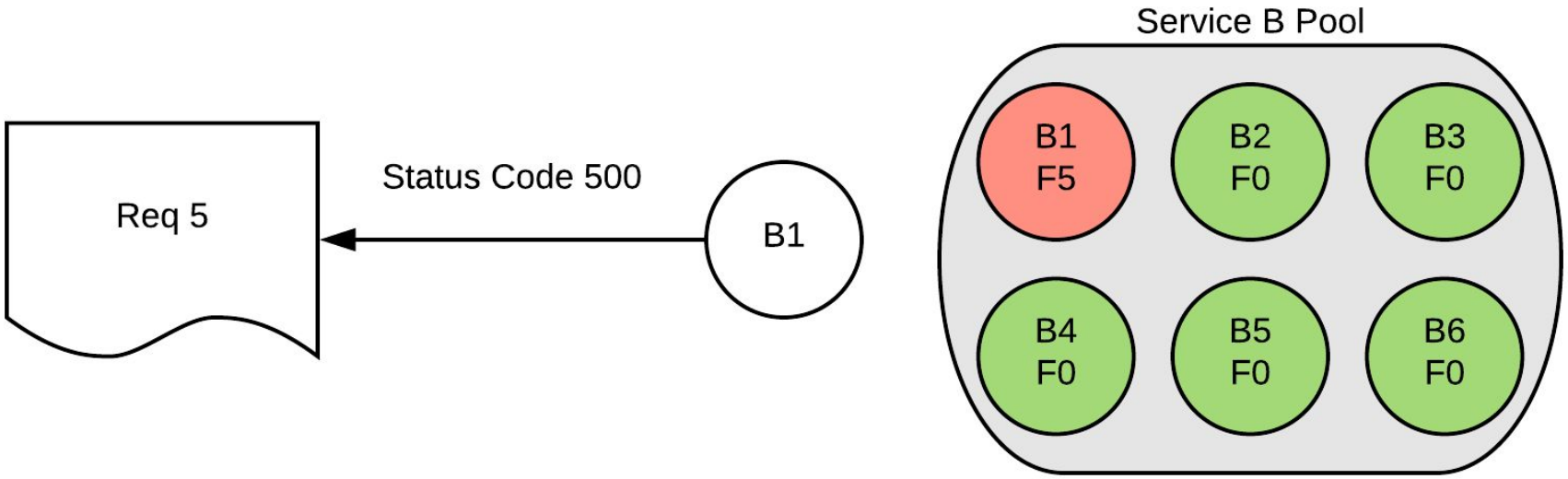
Circuit Breaking



Circuit Breaking

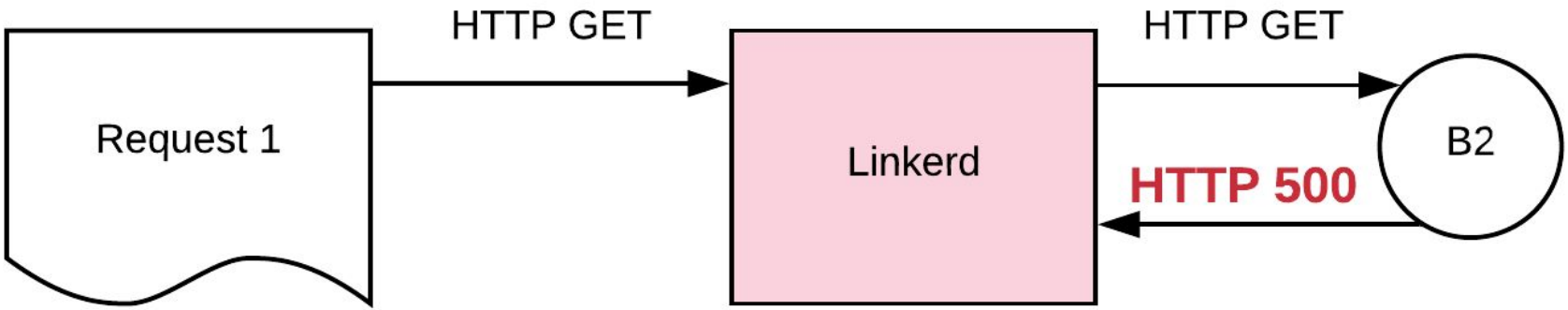


Circuit Breaking



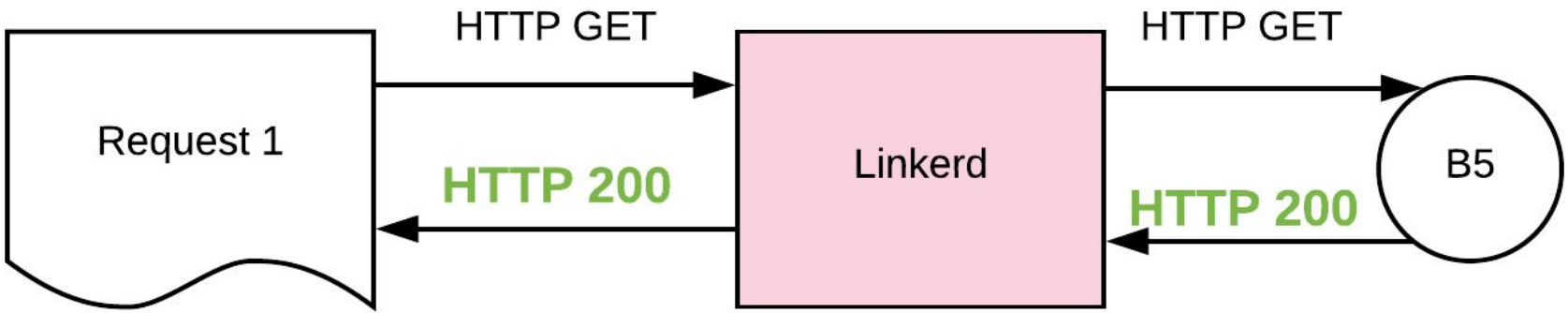
Retries

1st Try



Retries

2nd Try



Delivery Velocity

Where we are now

- Open Source
  - okhttp h2c patch from Jaye Pitzeruse
- Language agnostic
  - Java & Python
- HTTP2
  - 95% 2ms added latency
- Consul Integration
- Cross Data center

Client Implementation

Client Implementation

1. Retrieve outbound port

Client Implementation

1. Retrieve outbound port

```
port = Env.get("egressPort");
```

Client Implementation

1. Retrieve outbound port

```
port = Env.get("egressPort");
cl = Client("http://localhost:" + port);
```

Client Implementation

1. Retrieve outbound port
2. Inject service header

```
port = Env.get("egressPort");
cl = Client("http://localhost:" + port);
```

Client Implementation

1. Retrieve outbound port
2. Inject service header

```
port = Env.get("egressPort");
cl = Client("http://localhost:" + port);
req = cl.makeFooRequest();
req.setHeader("indeed-service", "ServiceB");
```

Service Implementation

Service Implementation

Future Plans

Future Plans

- Transparent TLS

Future Plans

- Transparent TLS
- Authentication
- Authorization

Future Plans

- Transparent TLS
- Authentication
- Authorization
- Rate Limiting

Future Plans

- Transparent TLS
- Authentication
- Authorization
- Rate Limiting
- Tracing

Future Plans

- Transparent TLS
- Authentication
- Authorization
- Rate Limiting
- Tracing
- Metrics

Future Plans

- Transparent TLS
- Authentication
- Authorization
- Rate Limiting
- Tracing
- Metrics
- Chaosify

Linkerd

- Finagle
- Cloud Native Computing Foundation
- Scala
- HTTP
- Plugin Support

Conduit

- Kubernetes
- Alpha
- Rust & golang
- TCP
- Linkerd 2

Envoy

- Lyft
- CNCP
- C++
- TCP
- Extensions

Istio

- IBM & Google
- Envoy underneath
- golang
- TCP
- Security

Consul Connect

- HashiCorp
- Beta
- golang
- TCP
- ACLs

indeed

Thanks for coming

jshanks@indeed.com

jjshanks