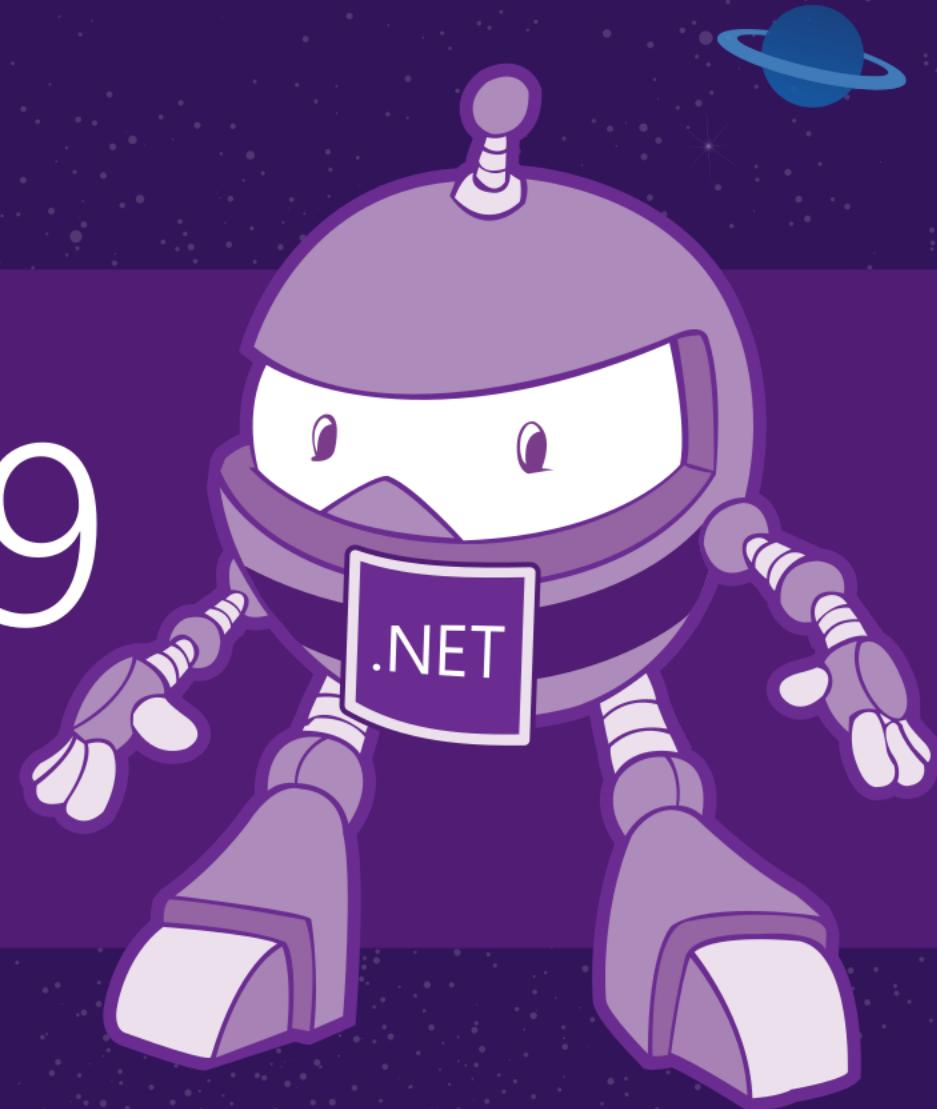


# .NET Conf 2019

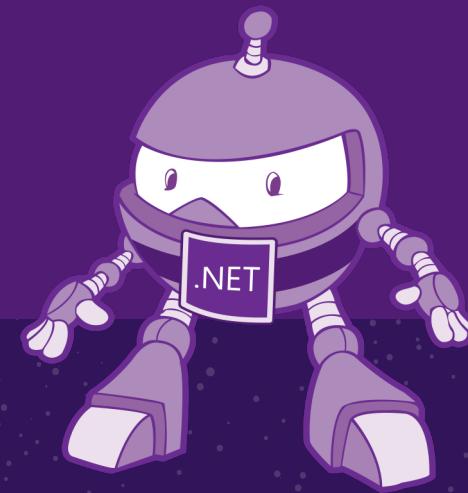
Discover the world of .NET



[www.dotnetconf.net](http://www.dotnetconf.net)

# Stretching the Service Mesh

Rosemary Wang



# Rosemary Wang

she/her

Developer Advocate

[rosemary@hashicorp.com](mailto:rosemary@hashicorp.com)



@joatmon08



joatmon08



**HashiCorp**



# Overview

- What's the buzz about service mesh?
- Examination: Tracing
- Examination: Traffic Management
- What can we conclude?

# What's the buzz about service mesh?

.NET



# We need a service mesh.

We can't  
trace  
transactions.

We don't  
have a way to  
control traffic.

We can't  
secure every  
application.

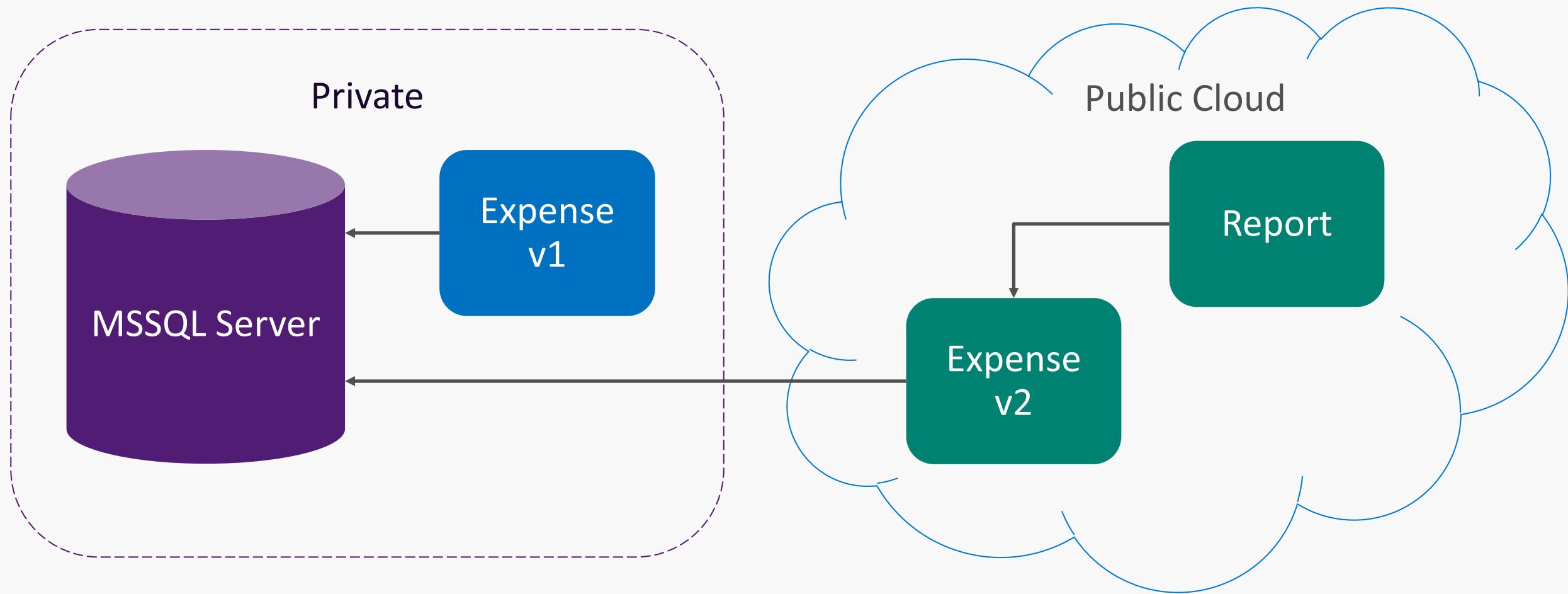
# We need a service mesh because Kubernetes has one.

We can't trace transactions.

We don't have a way to control traffic.

We can't secure every application.

# Monolith to Microservices, Private to Public Cloud...



# We need a library OR some proxy to control.

We can't  
trace  
transactions.

We don't  
have a way to  
control traffic.

We can't  
secure every  
application.

# Examination: Tracing\*

.NET



\* Tracing is non-trivial.

# Using a Library

```
public void ConfigureServices(IServiceCollection services)
{
    // sections omitted for clarity
    services.AddOpenTracing();
    services.AddSingleton<ITracer>(serviceProvider =>
    {
        // sections omitted for clarity
    });

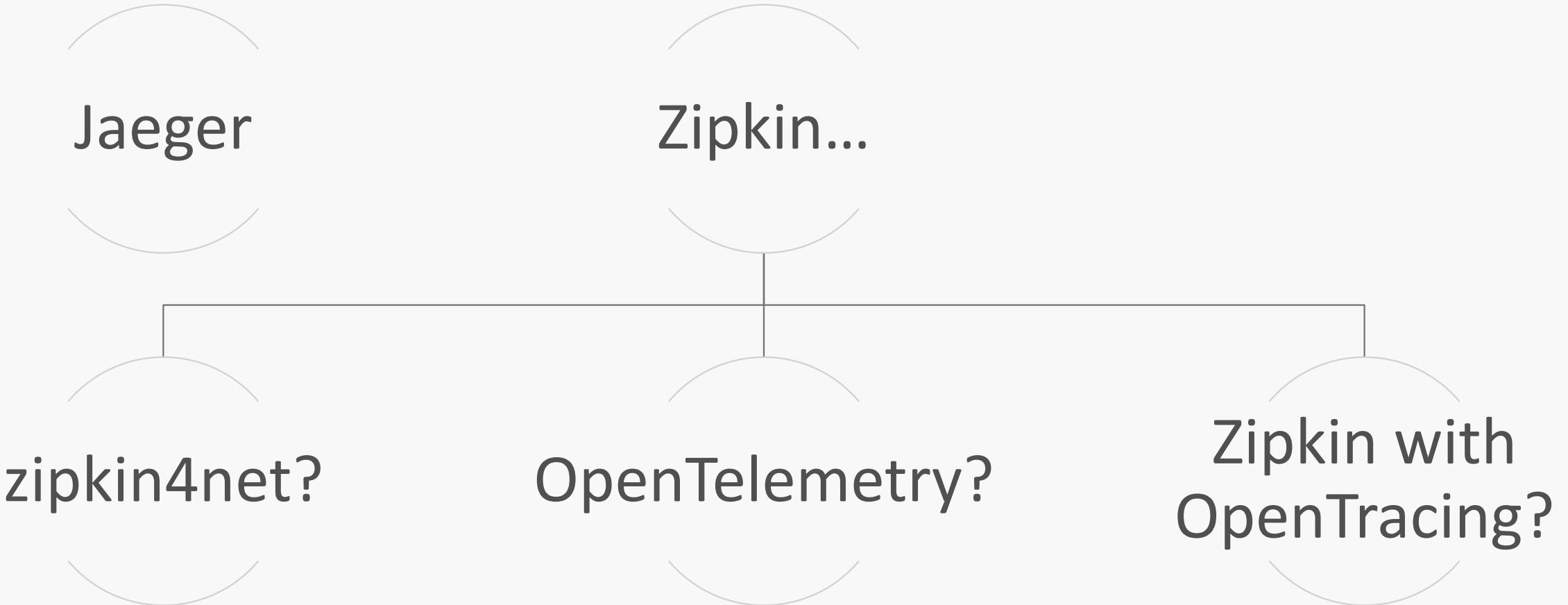
    services.Configure<HttpHandlerDiagnosticOptions>(options =>
    {
        options.IgnorePatterns.Add(request => request.RequestUri.ToString().Contains(Configuration.GetConnectionString("JaegerURL"))));
    });
}
```

# Using a Library (Jaeger)

```
services.AddSingleton<ITracer>(serviceProvider =>
{
    string serviceName = serviceProvider.GetRequiredService<IHostingEnvironment>().
        ApplicationName;
    var sampler = new ConstSampler(sample: true);
    var reporter = new RemoteReporter.Builder()
        .WithSender(new HttpSender(
            Configuration.GetConnectionString("Jaeger")))
        .Build();
    var tracer = new Tracer.Builder(serviceName)
        .WithSampler(sampler)
        .WithReporter(reporter)
        .Build();

    return tracer;
});
```

# Consul + Envoy



# Takeaways

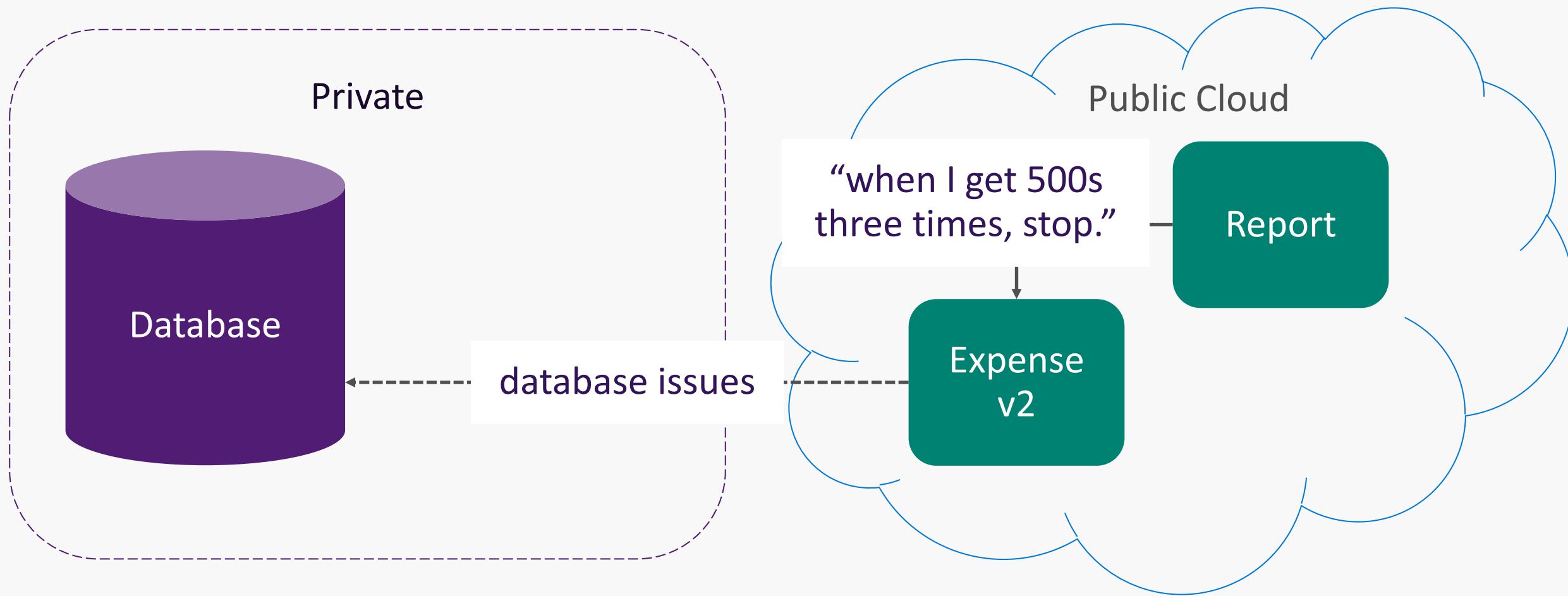
- Still need a library and code to mark spans
- Lots of movement in this space
  - OpenCensus + OpenTracing = OpenTelemetry
  - Currently, no stable library for .NET + Zipkin + OpenTracing
  - Envoy 1.12 (Jaeger)
- Keep it as library?

# Examination: Traffic Management

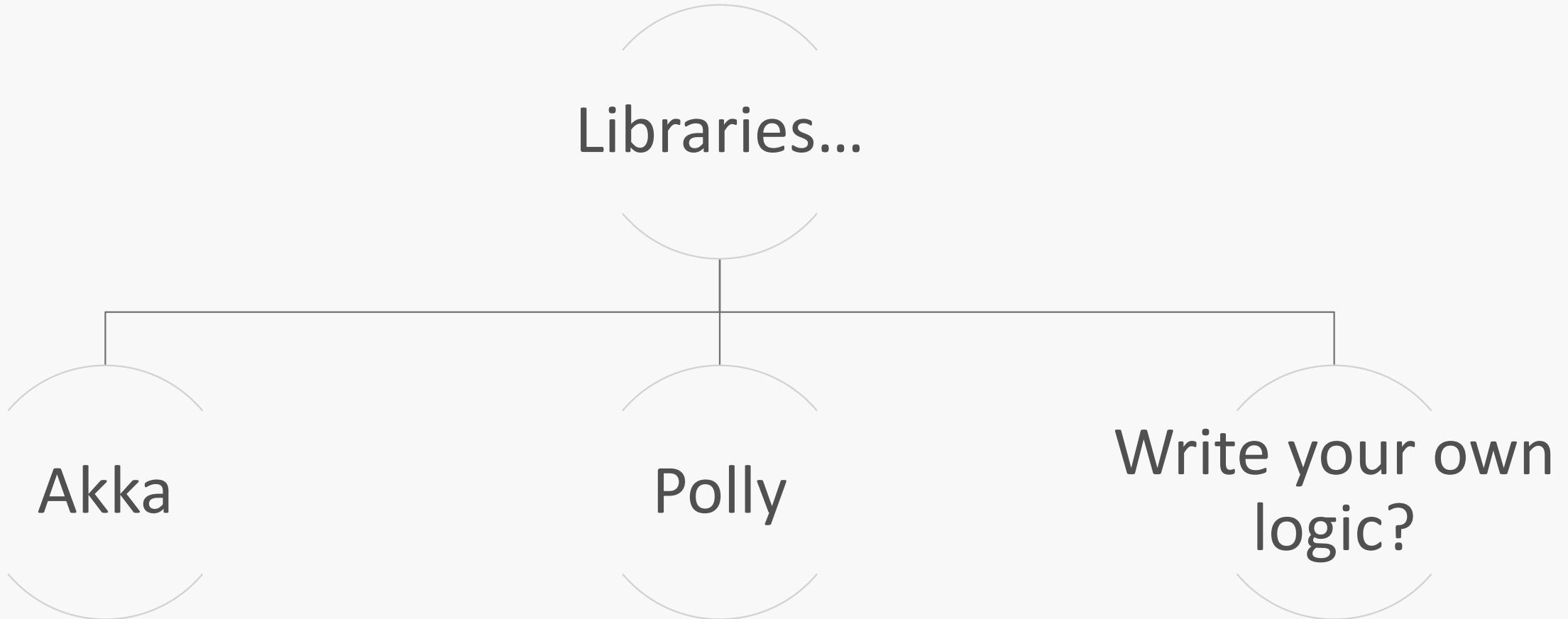
.NET



# Circuit Breaking



# Circuit Breaking Libraries

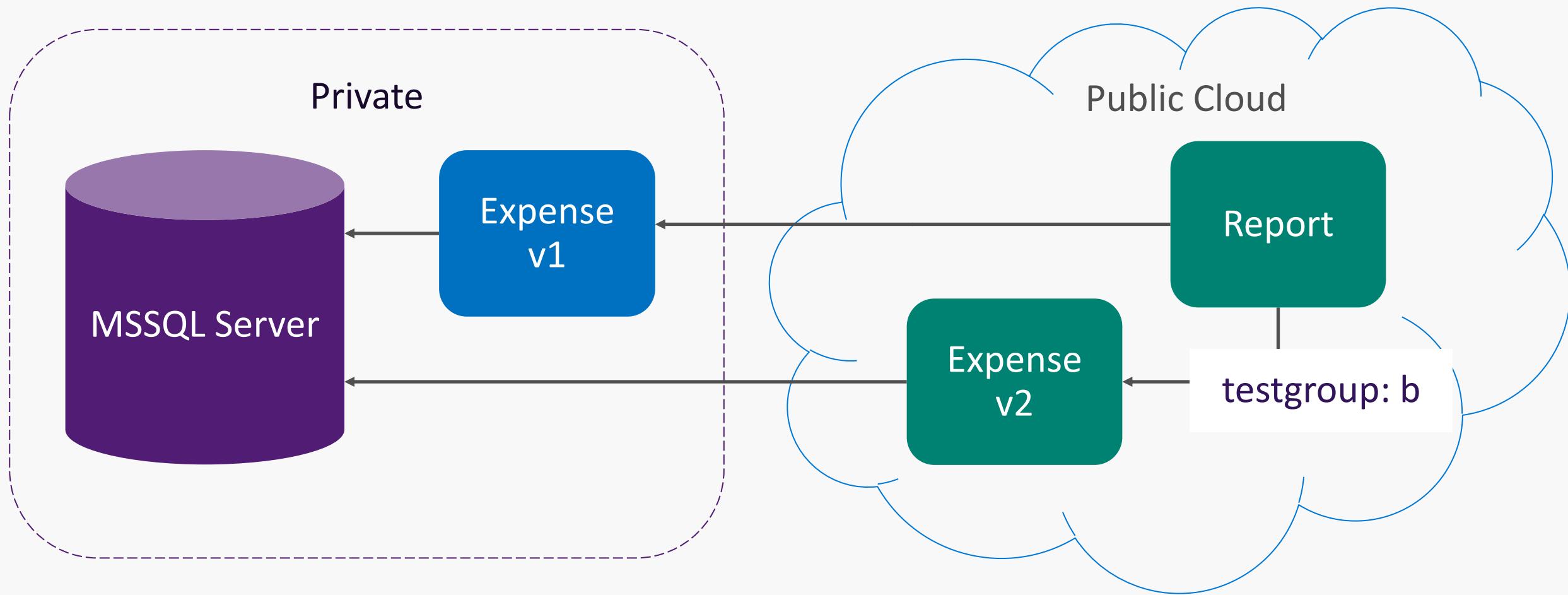


# Circuit Breaking (Envoy)

```
"circuit_breakers": {  
    "thresholds": [  
        {  
            "priority": "HIGH",  
            "max_requests": 10  
        }  
    ]  
}
```

```
"outlier_detection": {  
    "consecutive_5xx": 10,  
    "consecutive_gateway_failure": 10,  
    "base_ejection_time": "30s"  
}
```

# Canary or A/B Testing



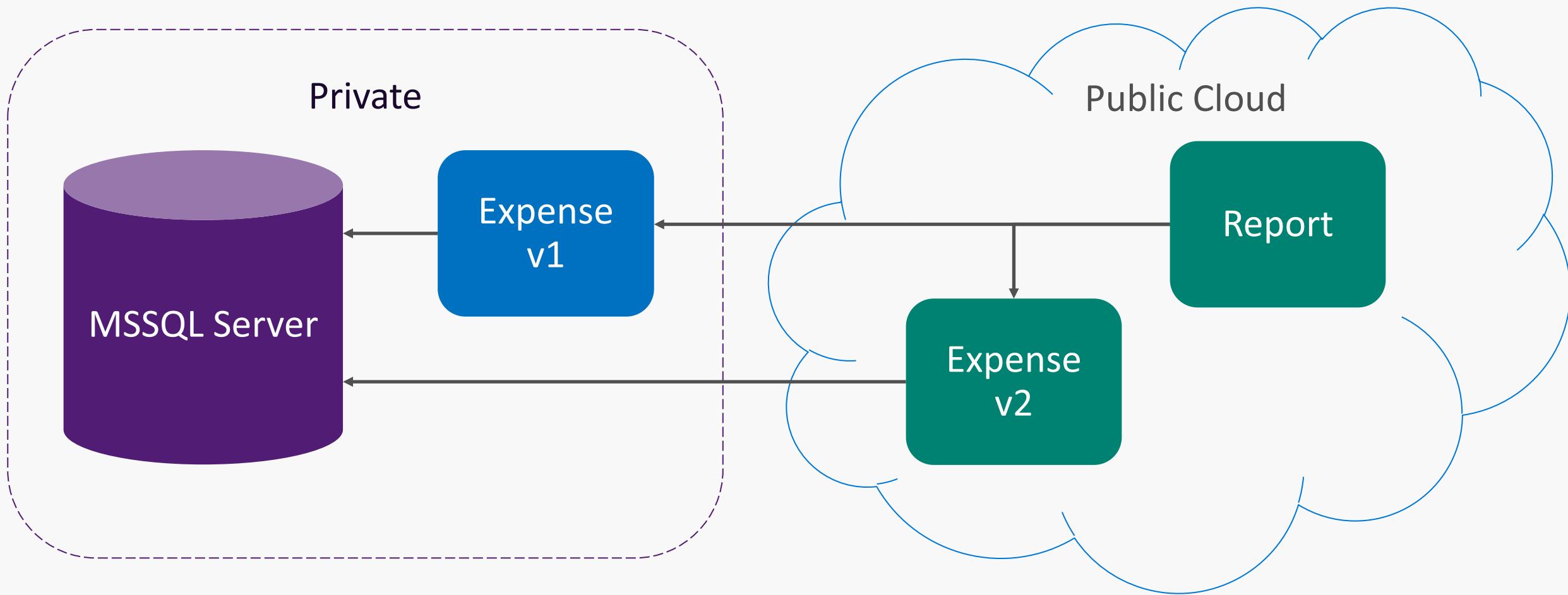
# Canary or A/B Testing

```
kind = "service-router"
name = "expense"
routes = [
    {
        match {
            http {
                header = [
                    {
                        name  = "testgroup"
                        exact = "b"
                    },
                    ]
                }
            }
        }

        destination {
            service = "expense"
            service_subset = "v2"
        }

    }
]
```

# Traffic Splitting



# Traffic Splitting

```
kind = "service-splitter",
name = "expense"

splits = [
    {
        weight = 50,
        service_subset = "v1"
    },
    {
        weight = 50,
        service_subset = "v2"
    }
]
```

# What can we conclude?

.NET



# Tracing

- Still non-trivial
- Libraries + mesh compatibility requires investigation
- *Maybe* greatest value in proxy in front of database?

# Traffic Shaping

- Worth attempting (fewer libraries)
- Less configuration within application code
- Control traffic based on intent

# References

- [envoyproxy.io/docs](https://envoyproxy.io/docs)
- Nic Jackson's Consul-Related Works
  - [github.com/nicholasjackson/demo-consul-service-mesh](https://github.com/nicholasjackson/demo-consul-service-mesh)
  - [hashicorp.com/resources/consul-service-mesh-deep-dive](https://hashicorp.com/resources/consul-service-mesh-deep-dive)
- [github.com/joatmon08/dotnet-service-mesh-example](https://github.com/joatmon08/dotnet-service-mesh-example)

# Thank you!

Rosemary Wang

[rosemary@hashicorp.com](mailto:rosemary@hashicorp.com)

@joatmon08