

API Simulation Basics

Agenda

SpectoLabs

- Why we need API simulation

Agenda

- Why we need API simulation
- Introduction to API simulation

Agenda

- Why we need API simulation
- Introduction to API simulation
- Hoverfly Demonstration

Agenda

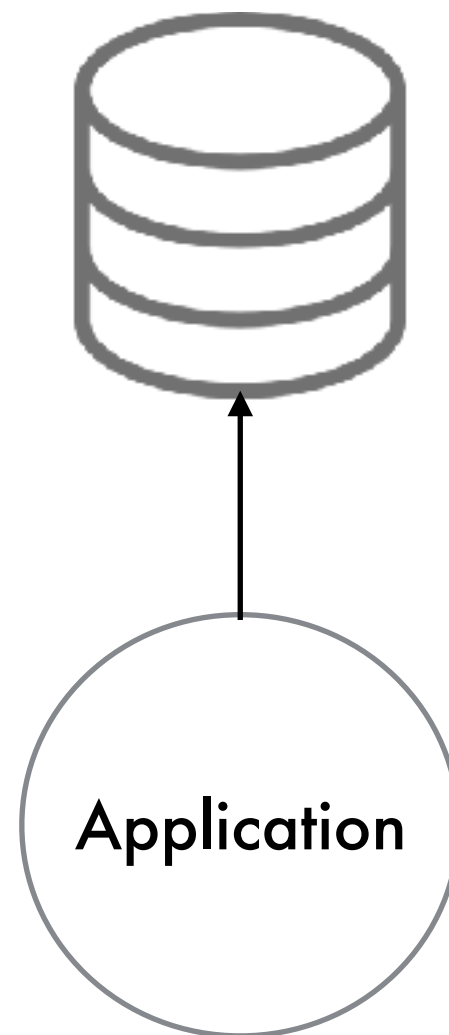
- Why we need API simulation
- Introduction to API simulation
- Hoverfly Demonstration
- Exercise One: Simulating Your First API

Agenda

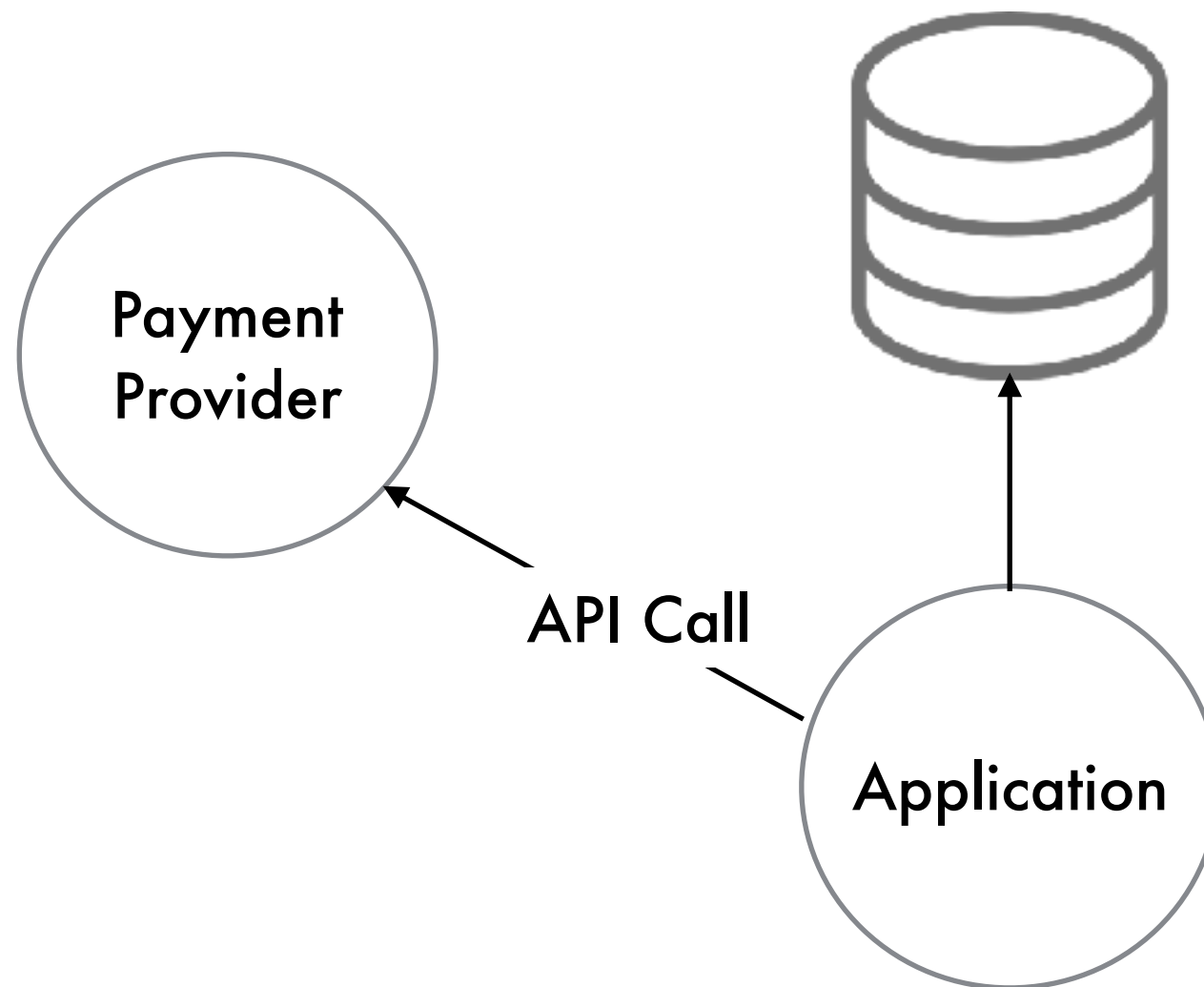
- Why we need API simulation
- Introduction to API simulation
- Hoverfly Demonstration
- Exercise One: Simulating Your First API

API-Driven Architecture

SpectoLabs

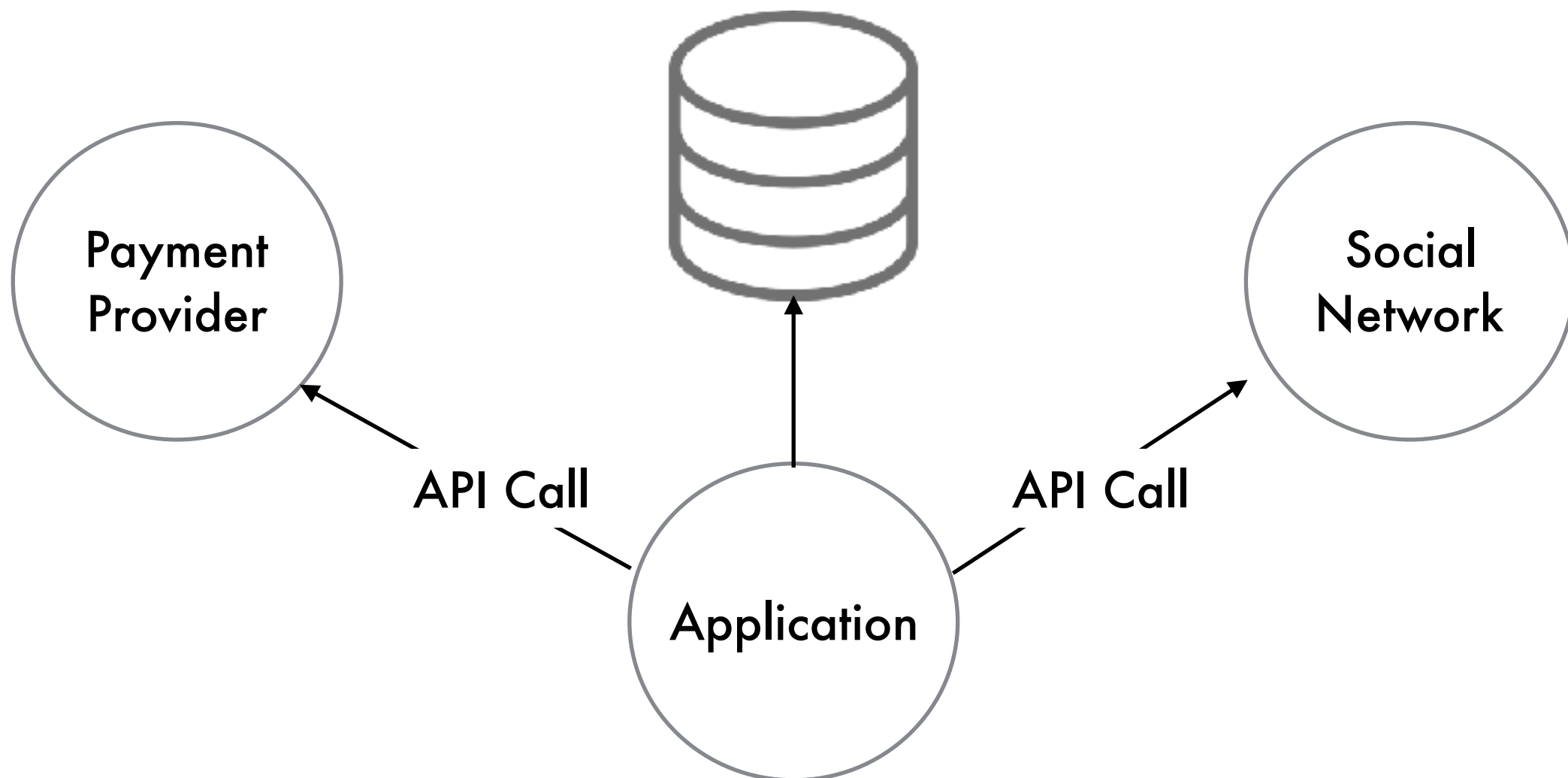


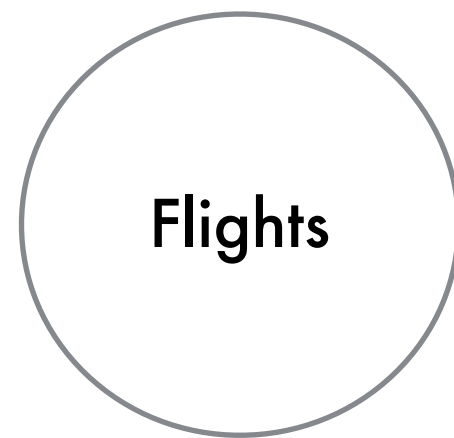
API-Driven Architecture

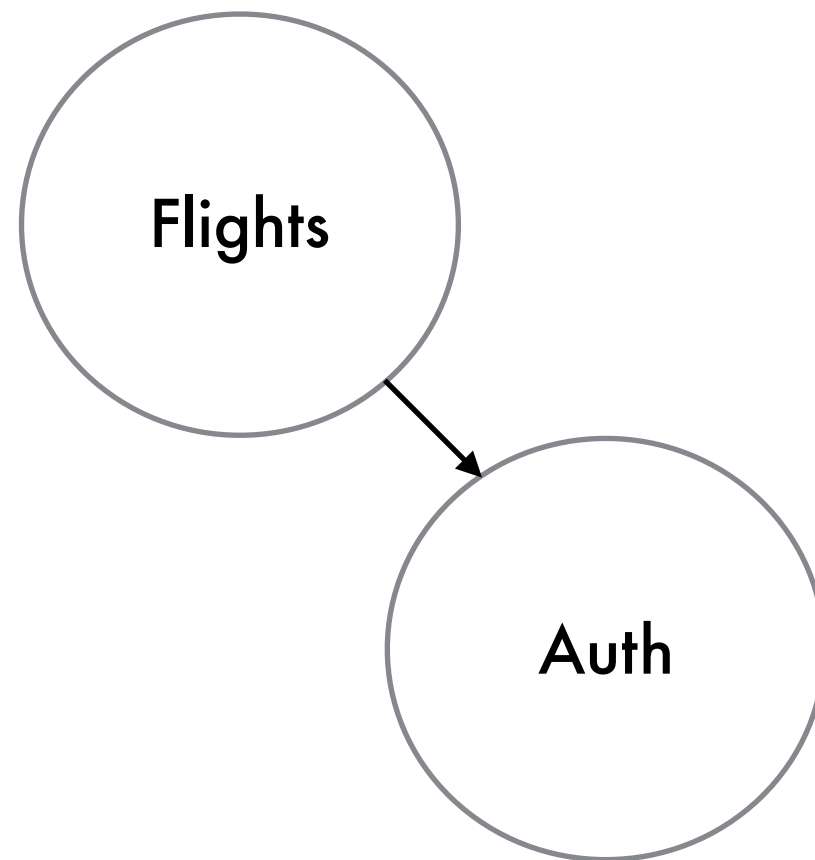


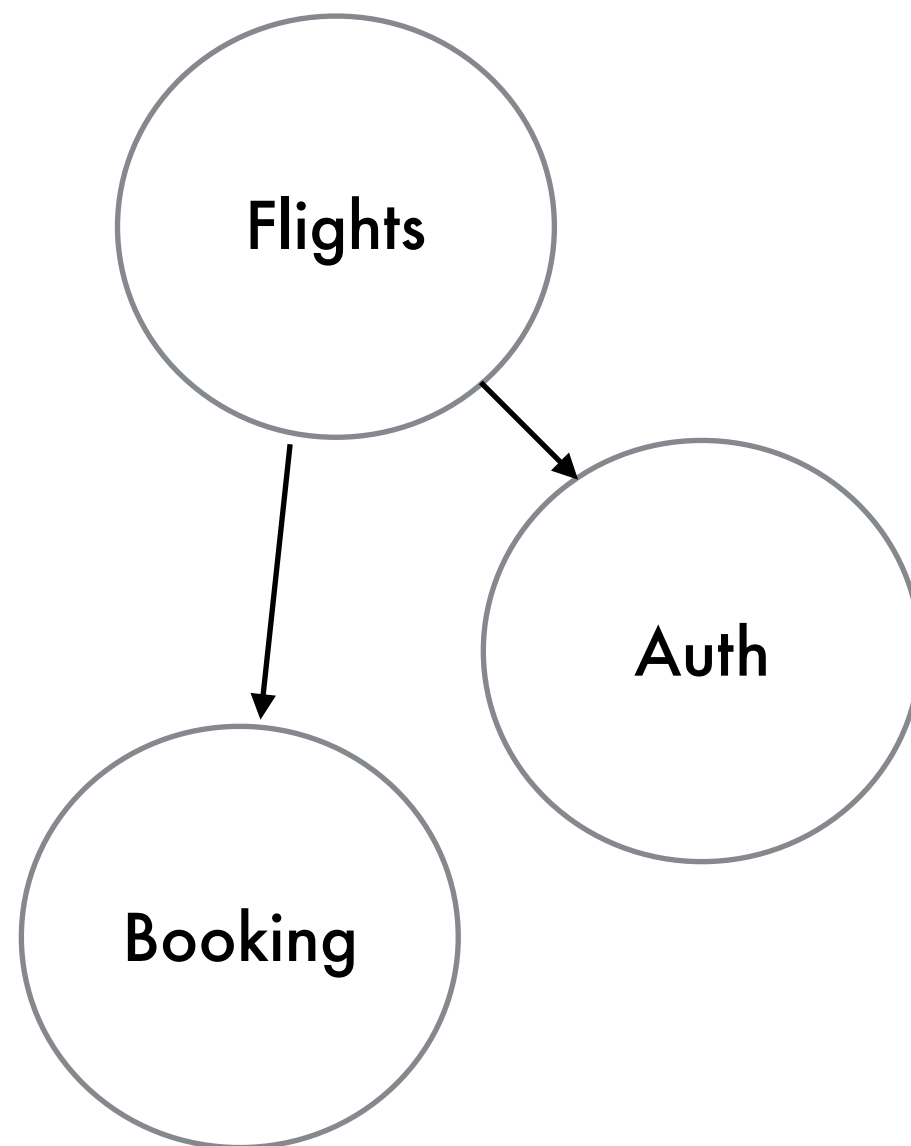
API-Driven Architecture

SpectoLabs

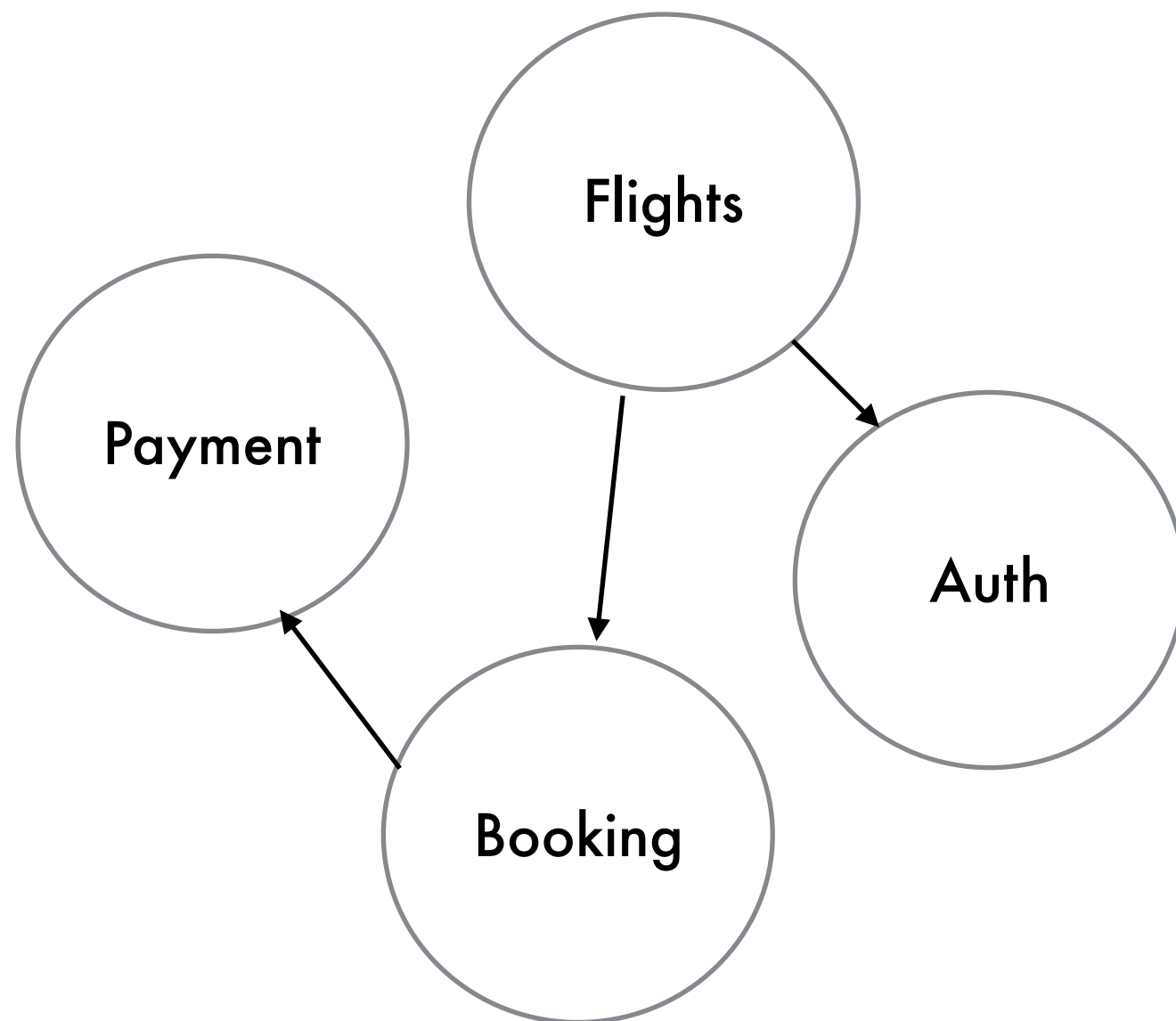


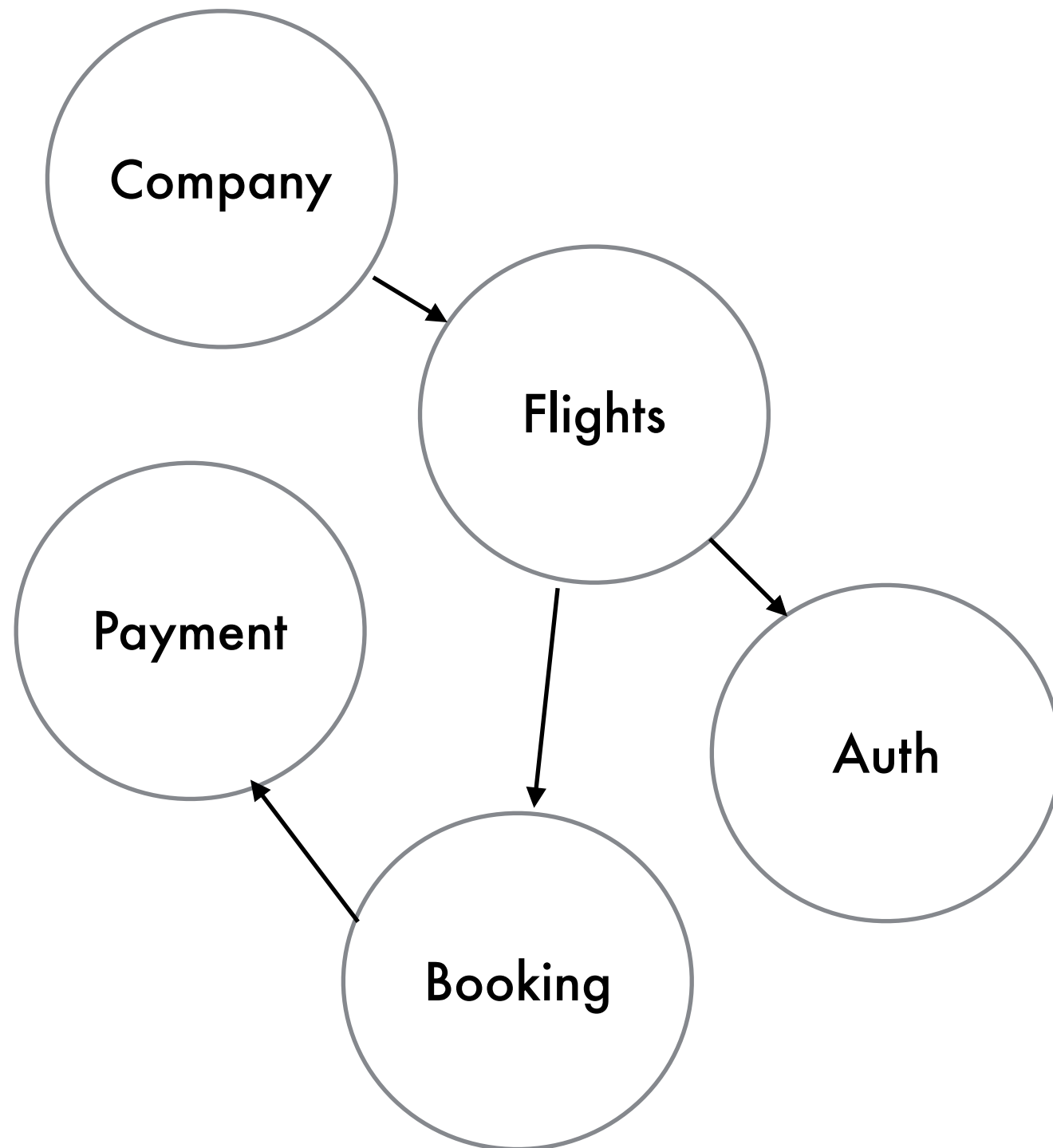




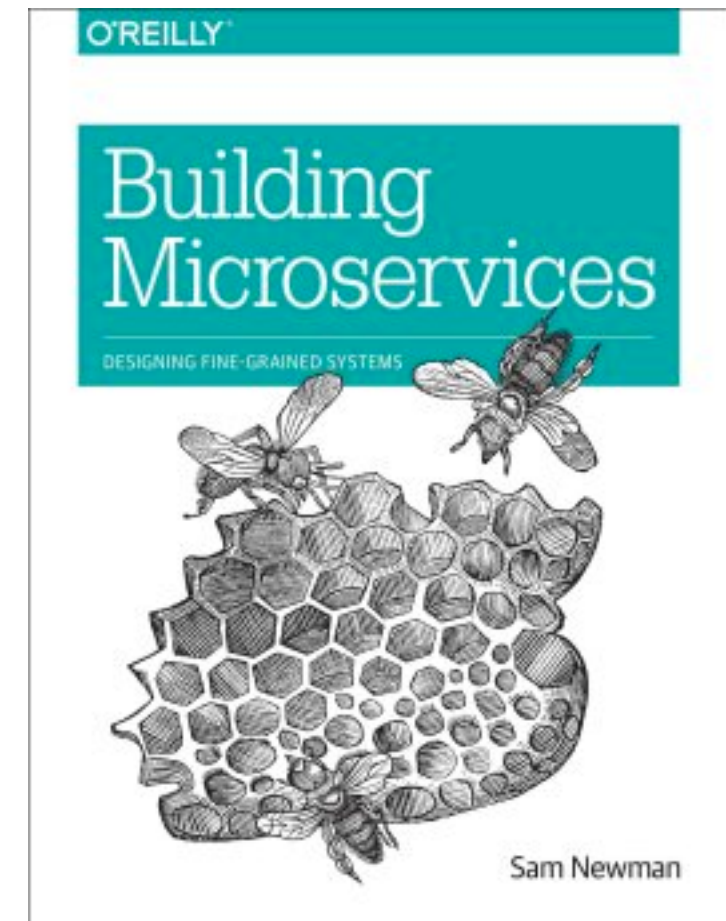
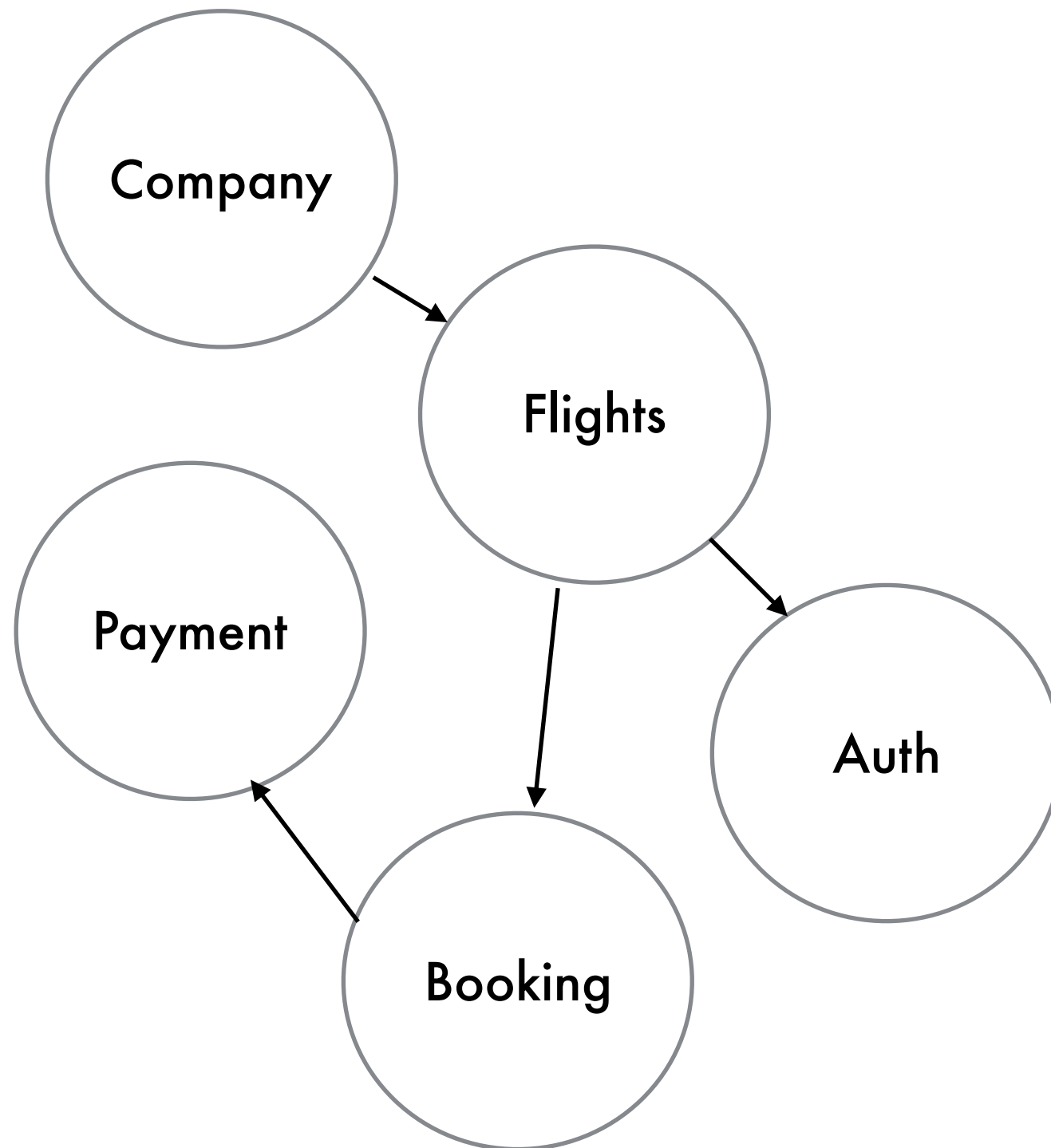


Microservices








Microservices

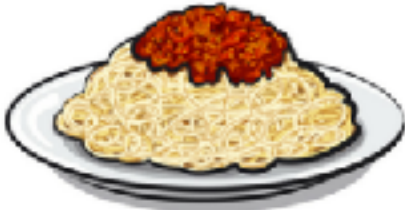


A natural evolution...


 **Benoit Hediard**
@benorama  

My take on the evolution of software architecture (& italian food) [medium.com/@benorama/the- ...](https://medium.com/@benorama/the-...)
[#microservices](#) [#ravioli](#)


1990's
SPAGHETTI-ORIENTED
ARCHITECTURE
(aka Copy & Paste)




2000's
LASAGNA-ORIENTED
ARCHITECTURE
(aka Layered Monolith)







2010's
RAVIOLI-ORIENTED
ARCHITECTURE
(aka Microservices)



RETWEETS **95** LIKES **63** 

1:51 AM - 28 May 2015

Paris, France

  **95**  **63** 

Challenges...

Airline Booking Application **SpectoLabs**

Airline Booking Application **SpectoLabs**

- Microservice architecture

Airline Booking Application **SpectoLabs**

- Microservice architecture
- Integration with third party services

Airline Booking Application **SpectoLabs**

- Microservice architecture
- Integration with third party services
- Integration with legacy internal services

Airline Booking Application **SpectoLabs**

- Microservice architecture
- Integration with third party services
- Integration with legacy internal services
- Heavy resource consumption

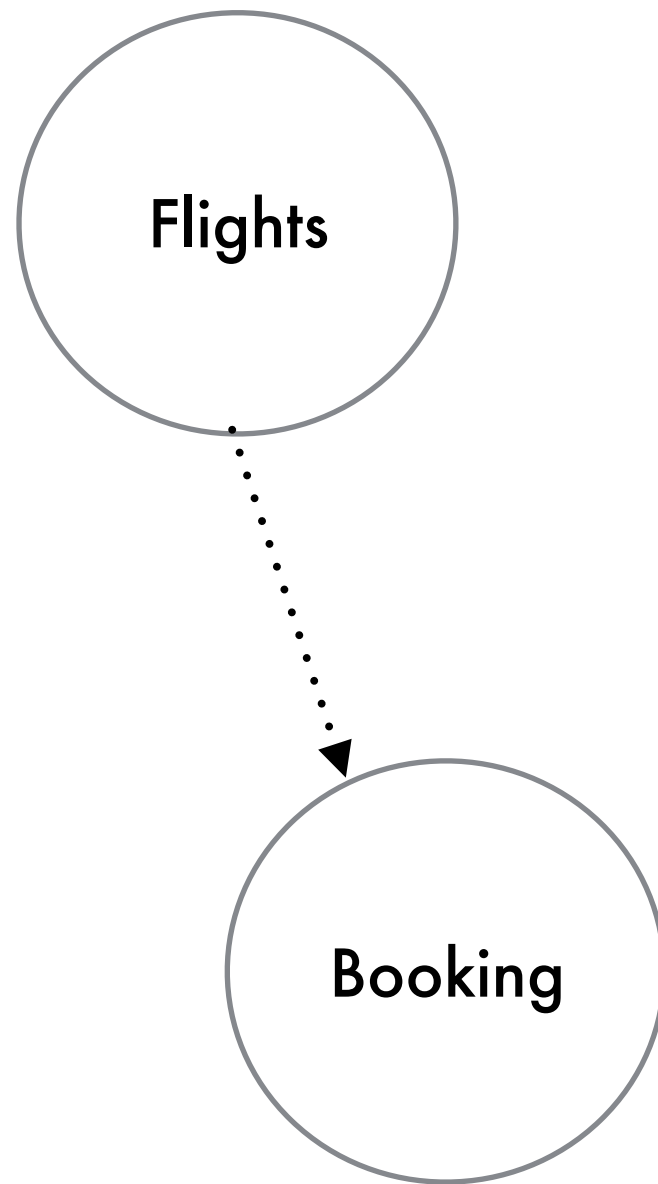
Airline Booking Application **SpectoLabs**

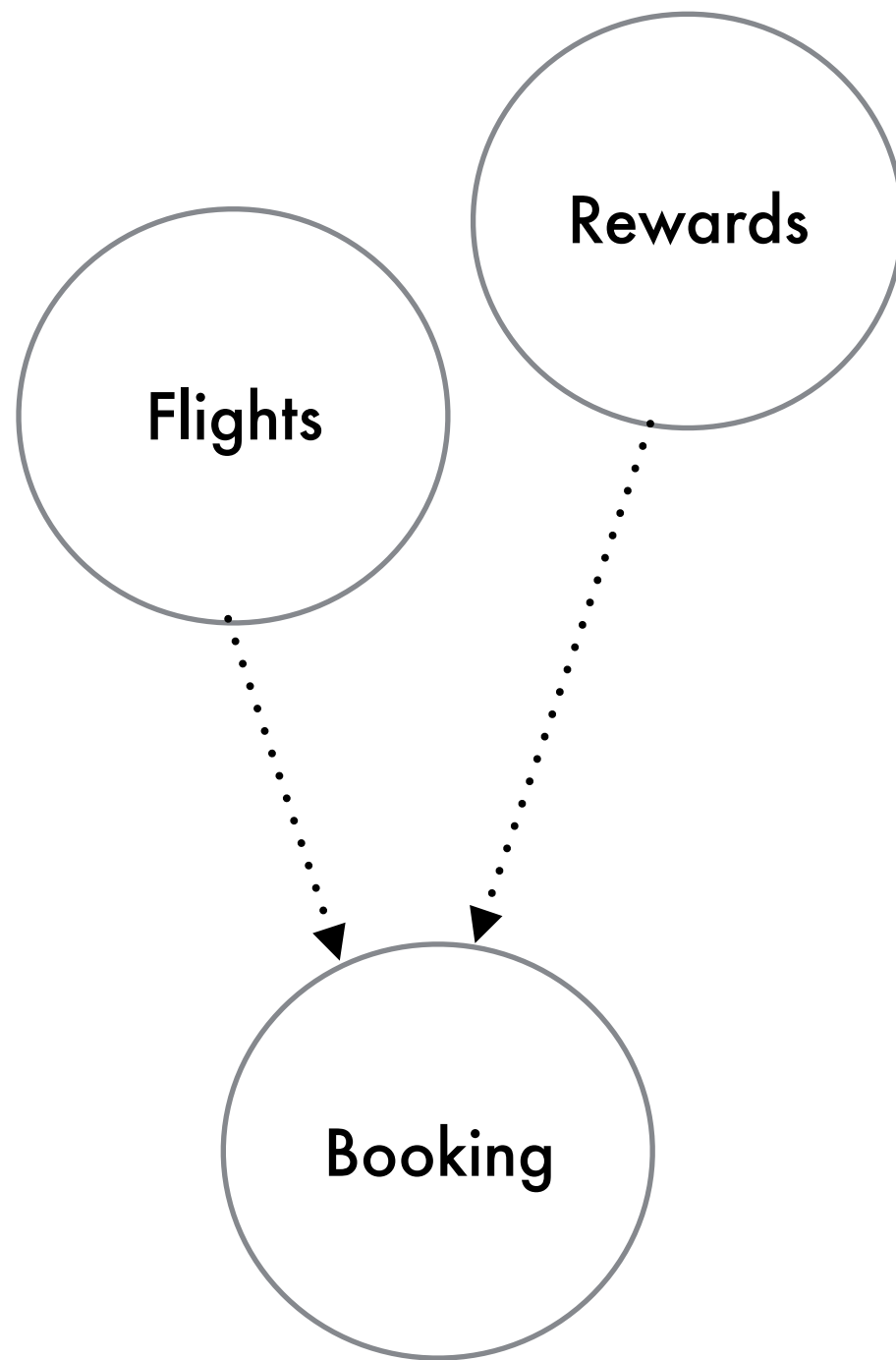
- Microservice architecture
- Integration with third party services
- Integration with legacy internal services
- Heavy resource consumption
- A standard CI/CD pipeline

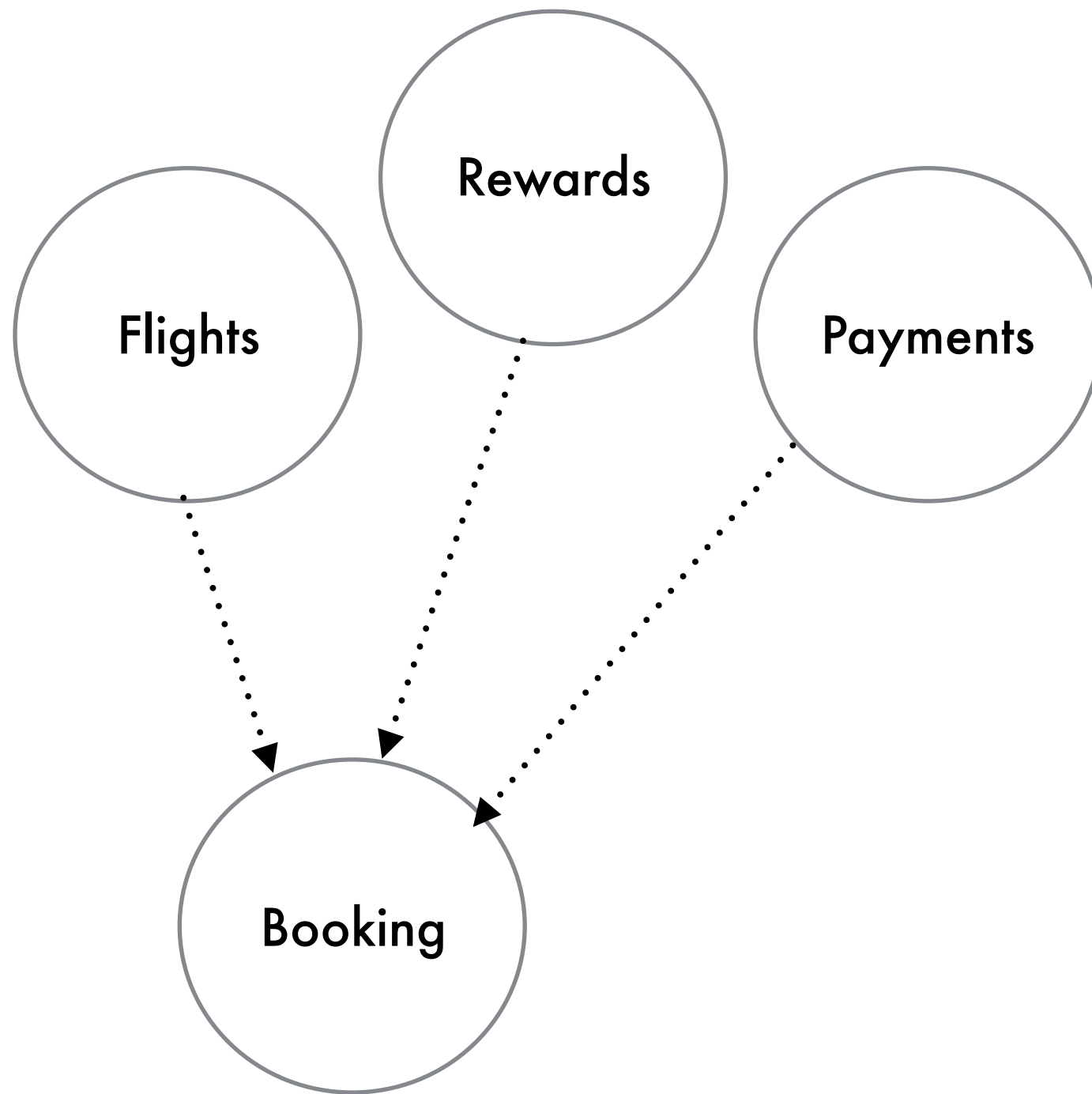
Resources

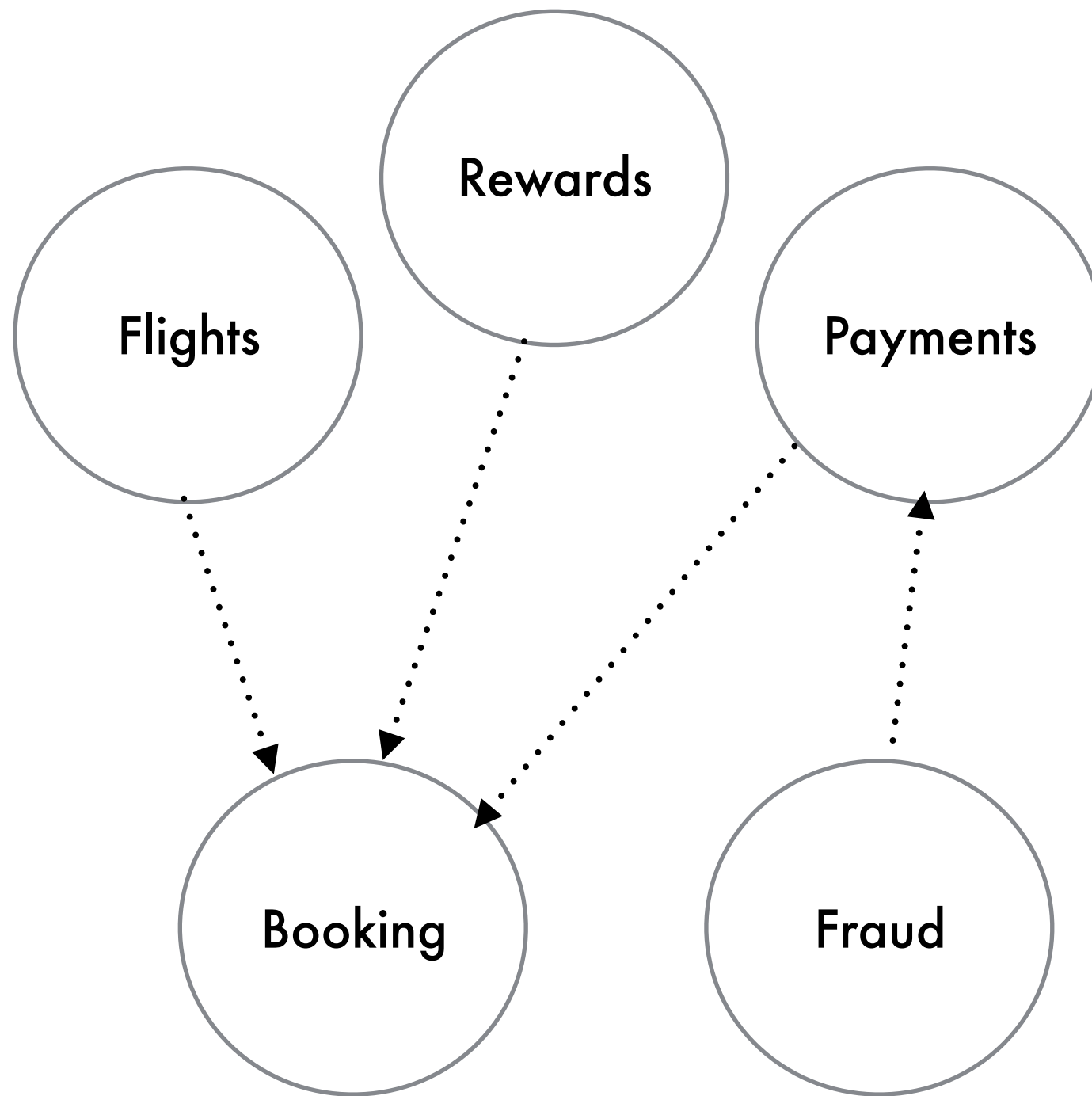


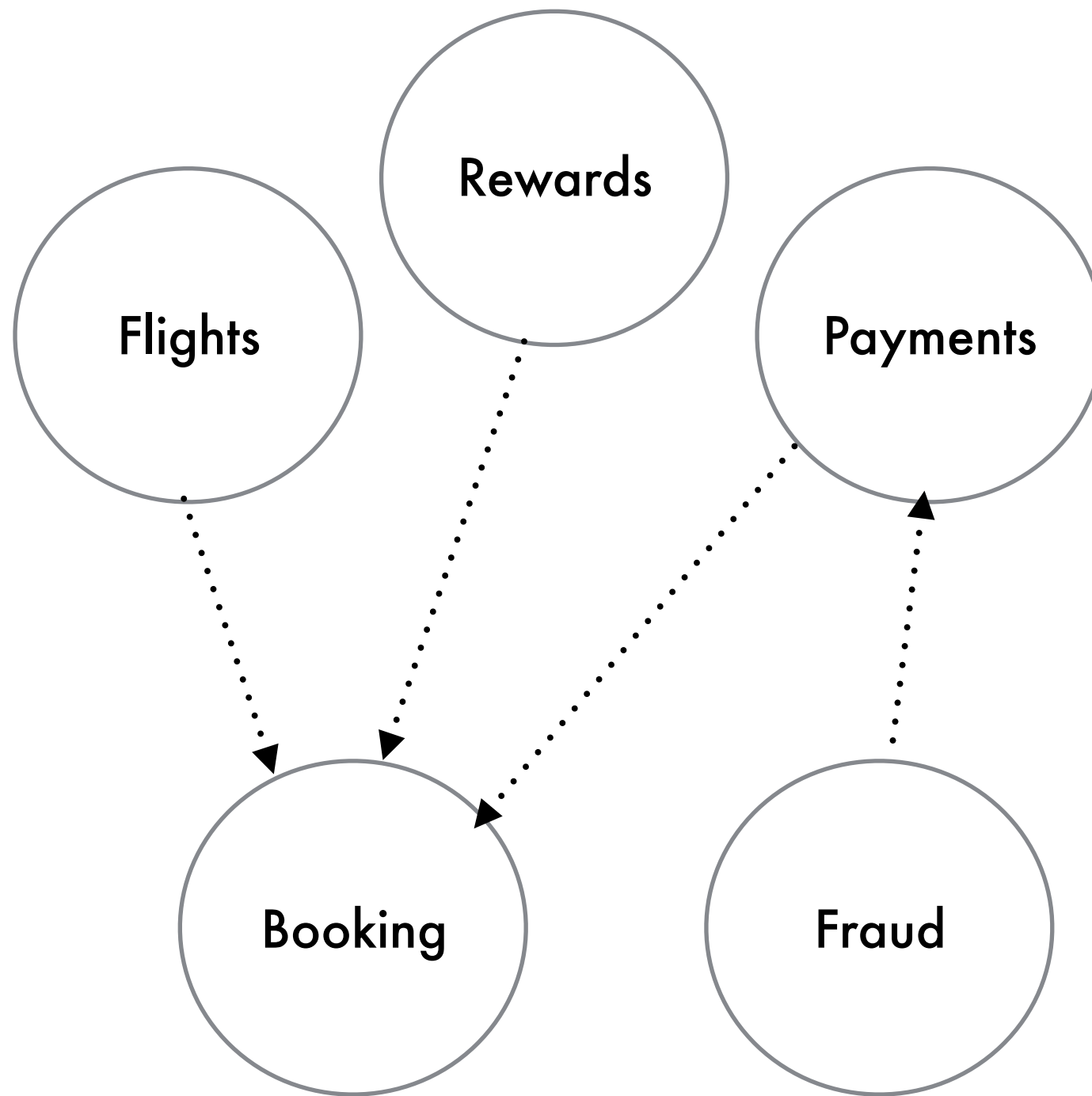
Booking





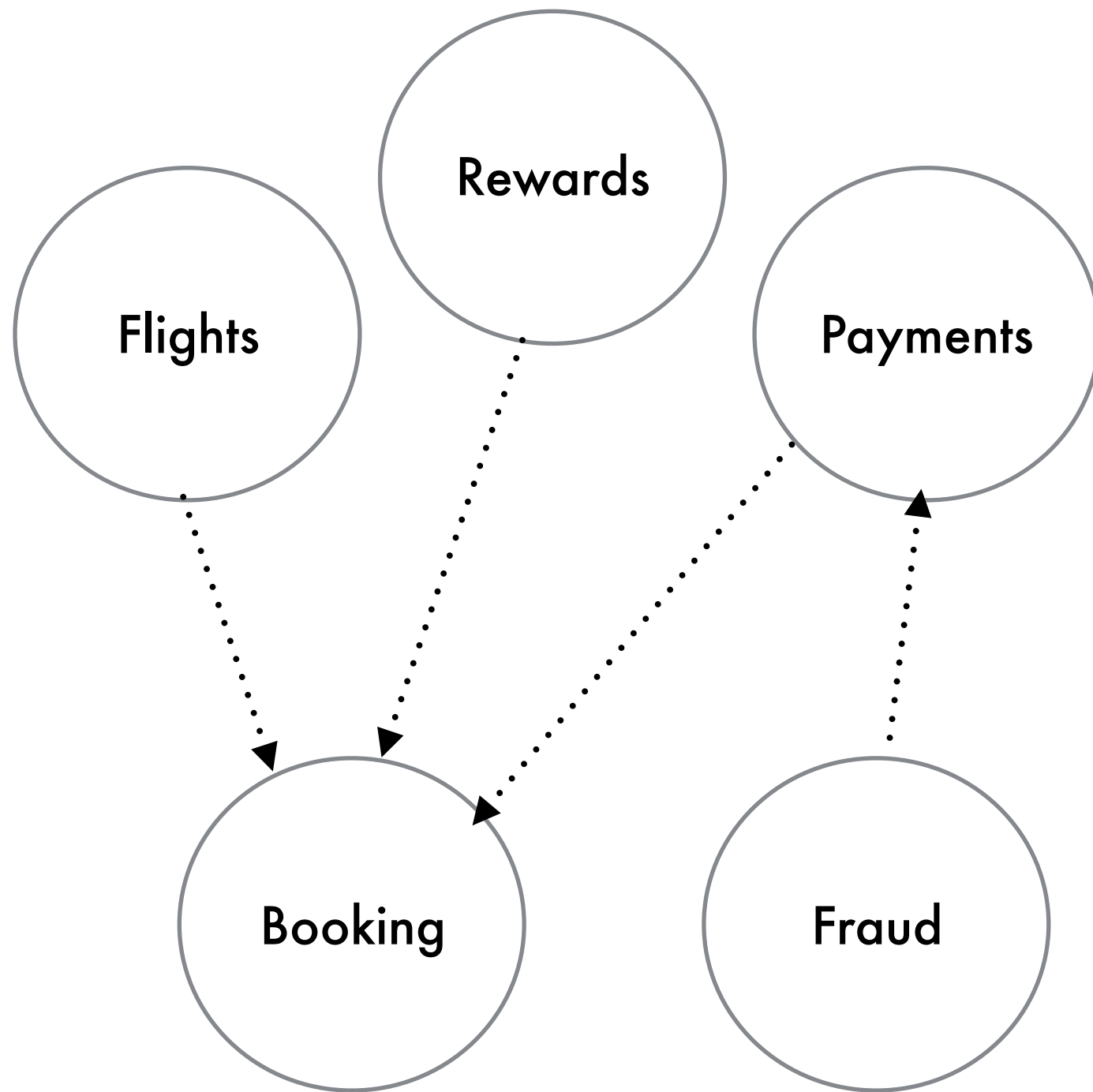






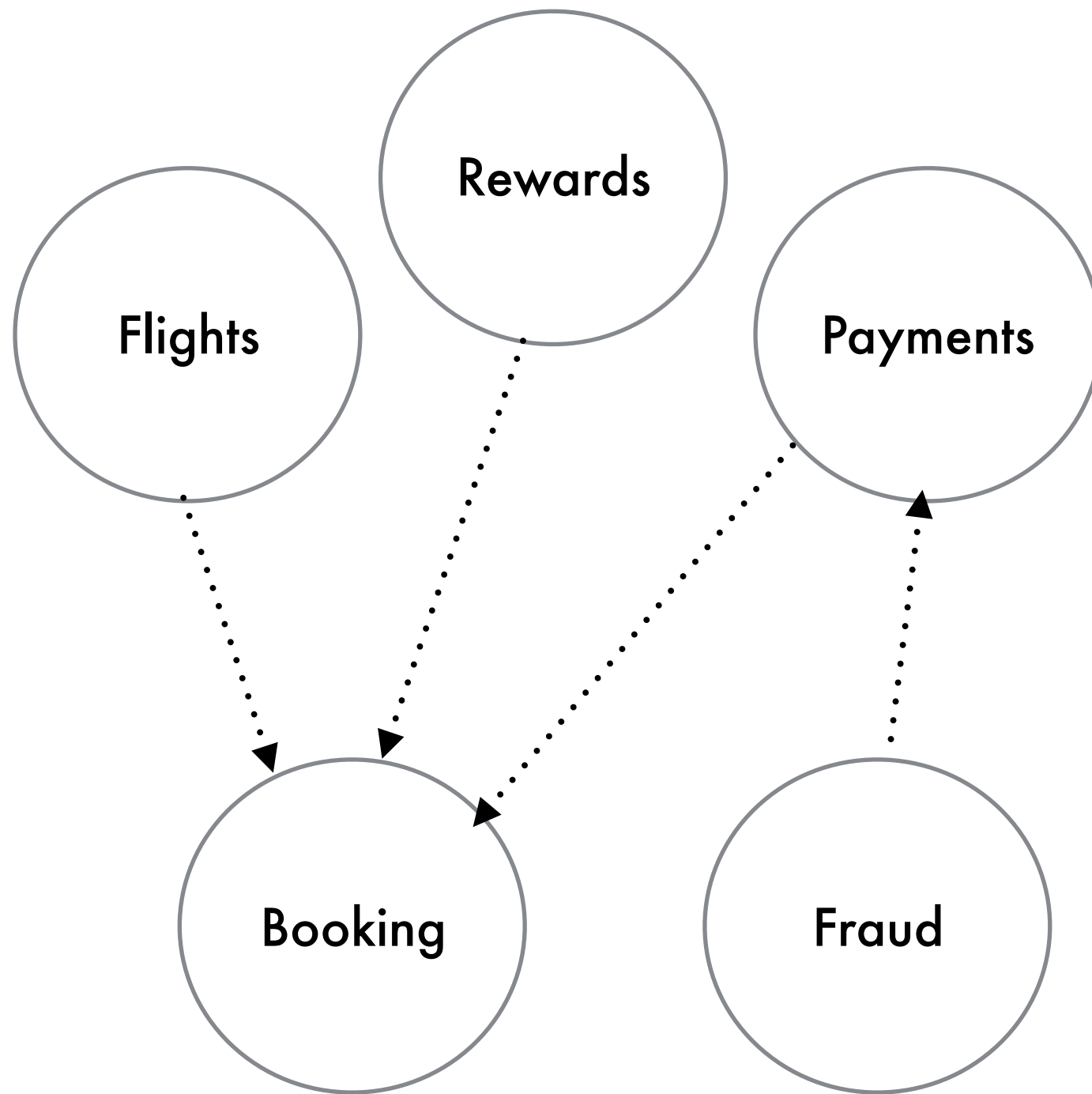
Resources

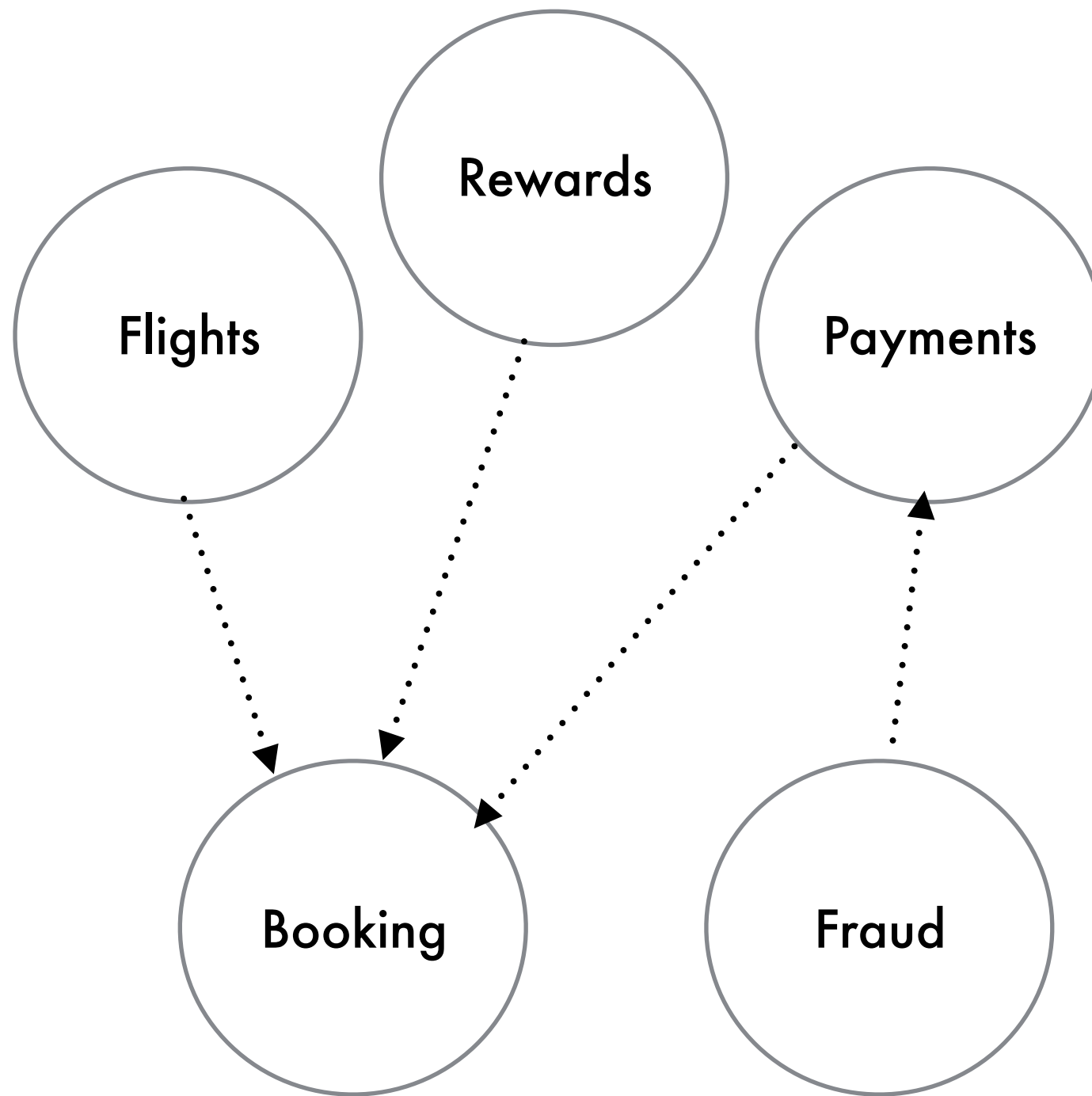
SpectoLabs



Resources

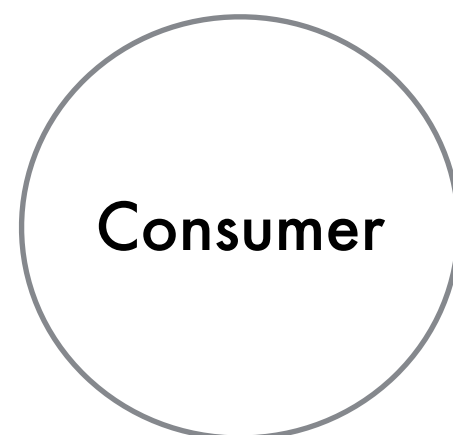
SpectoLabs

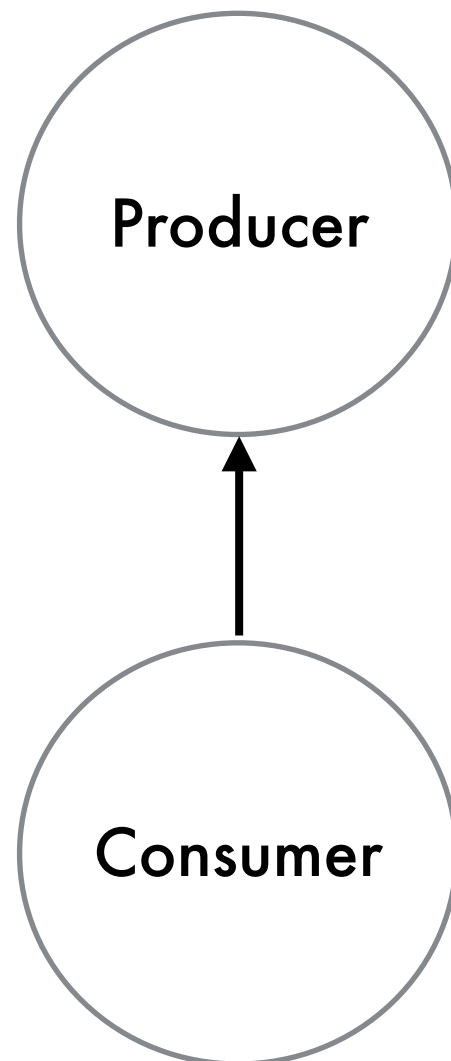


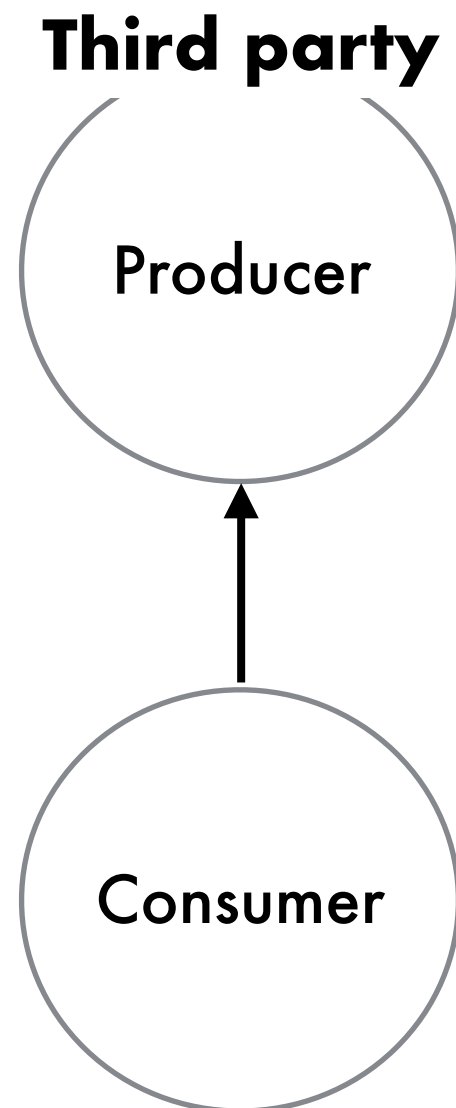


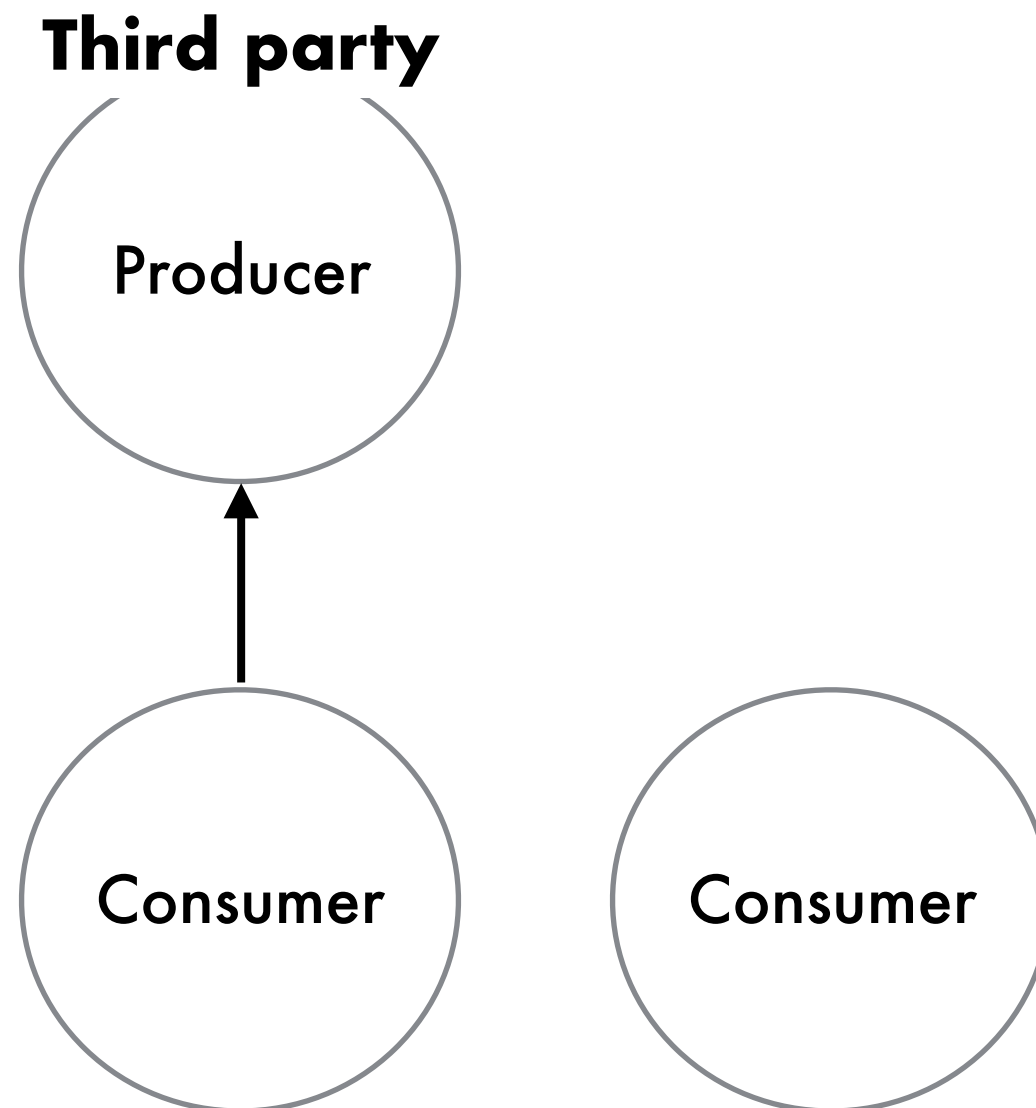
Cost

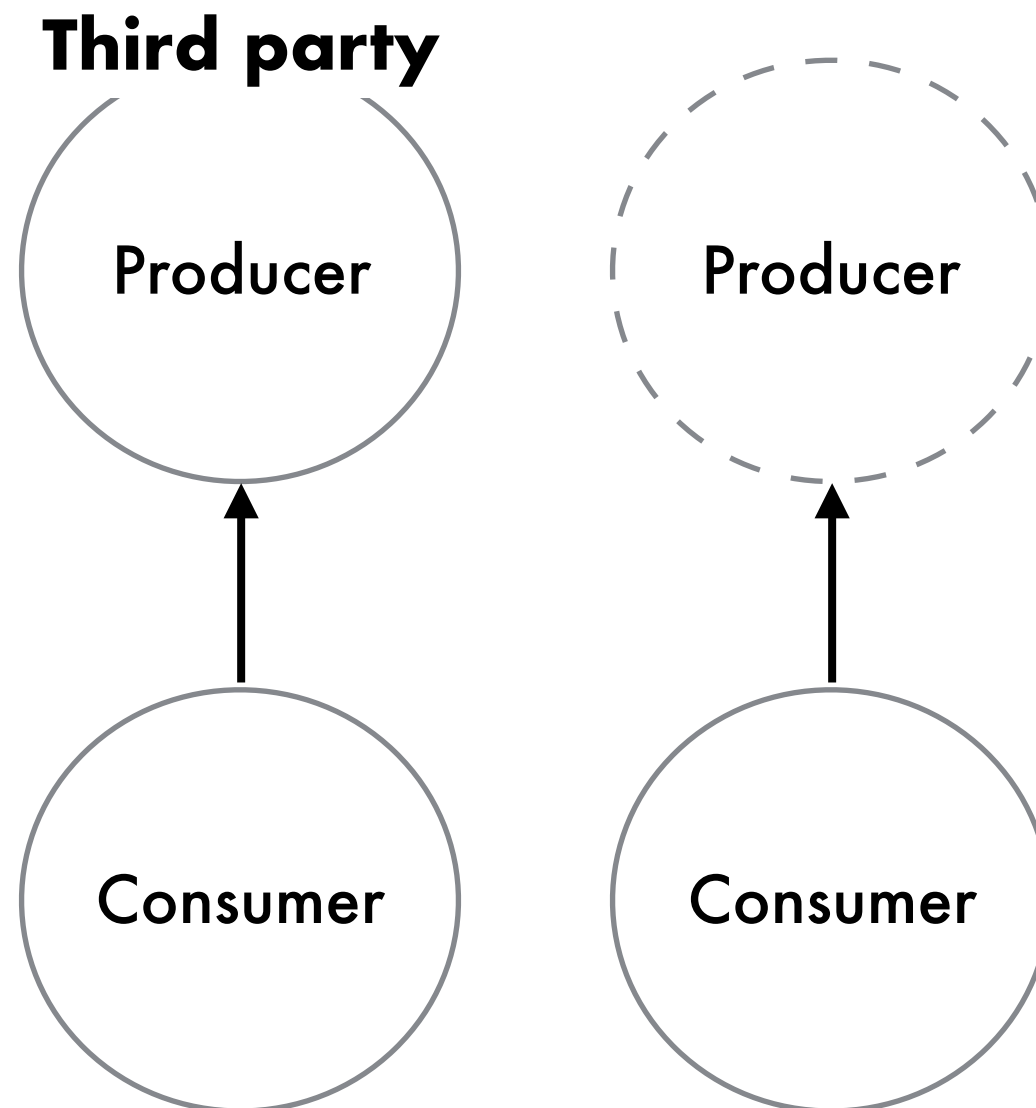


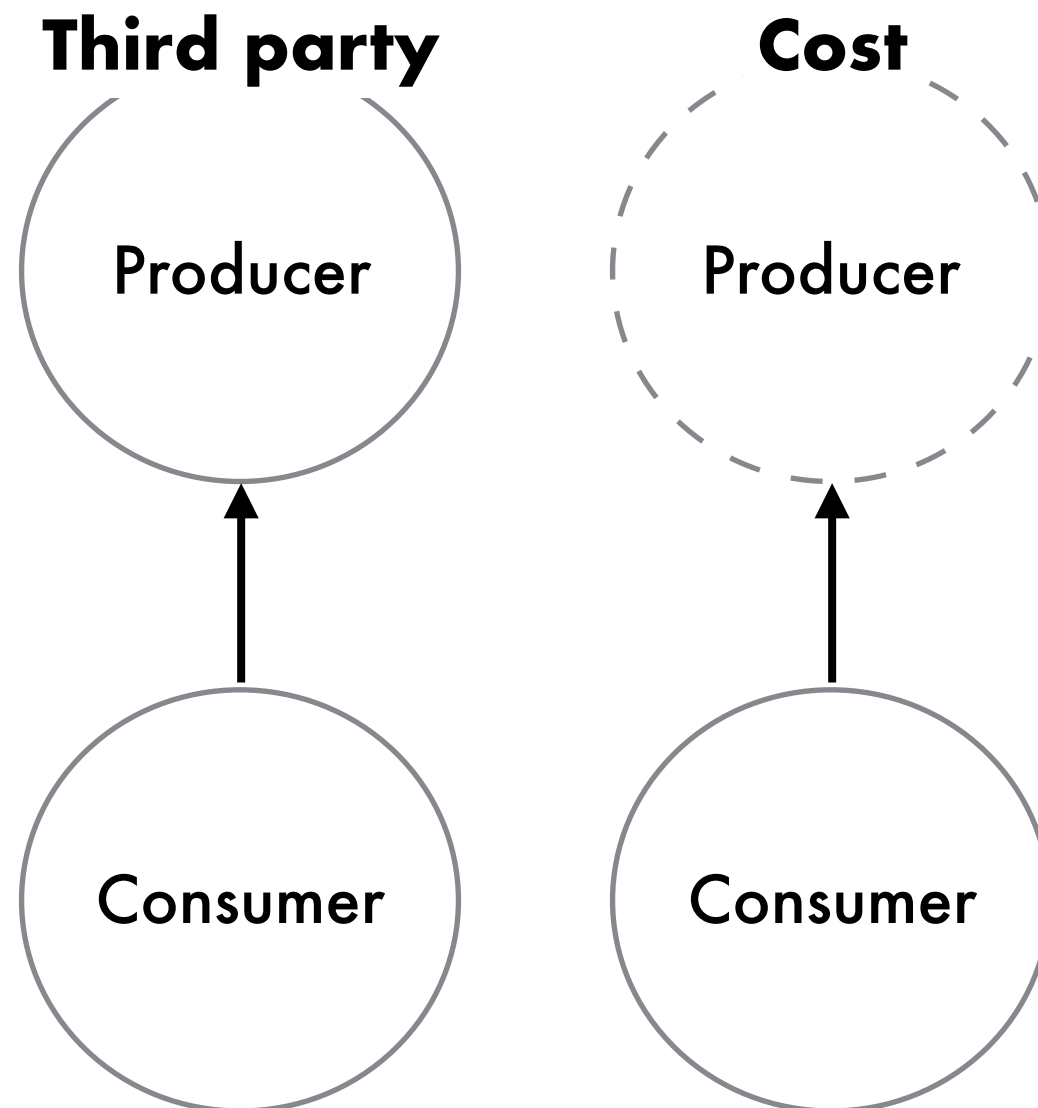






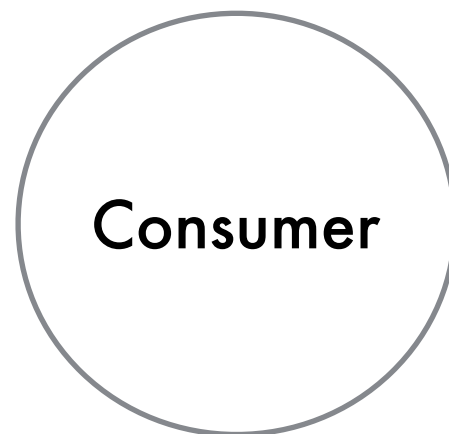


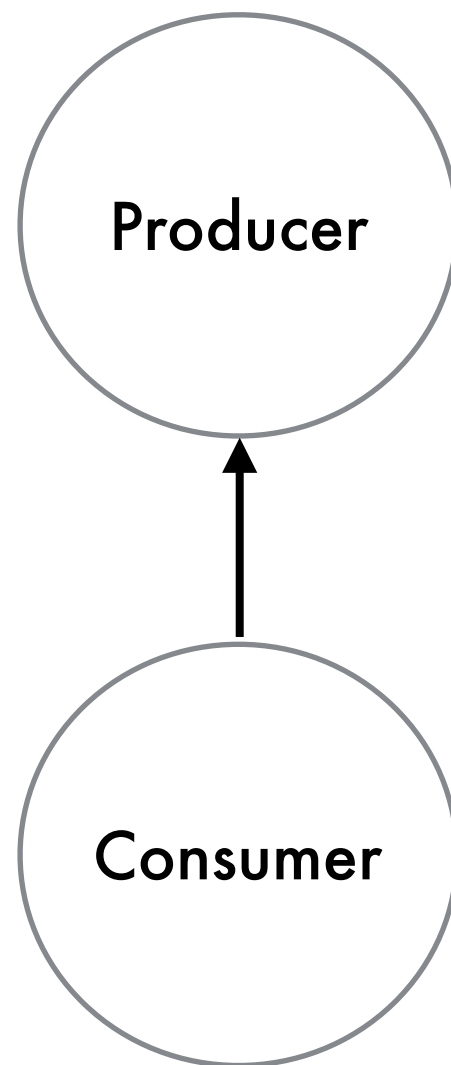




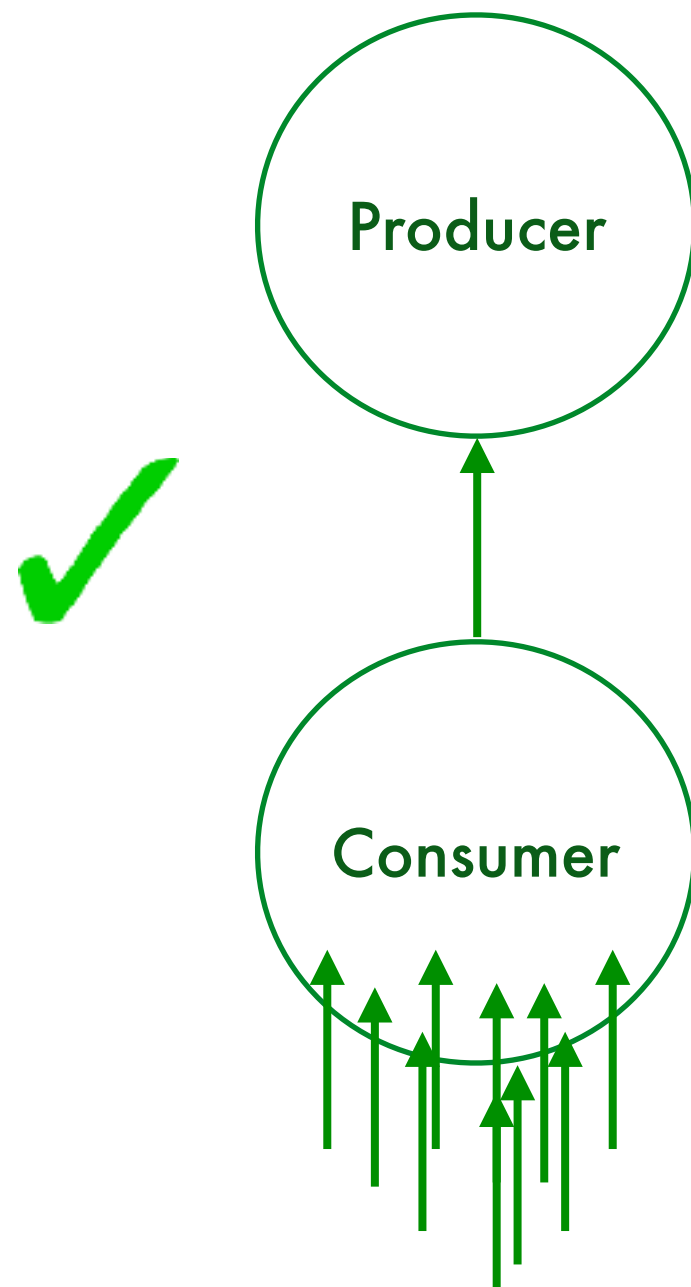
Rate Limiting

SpectoLabs

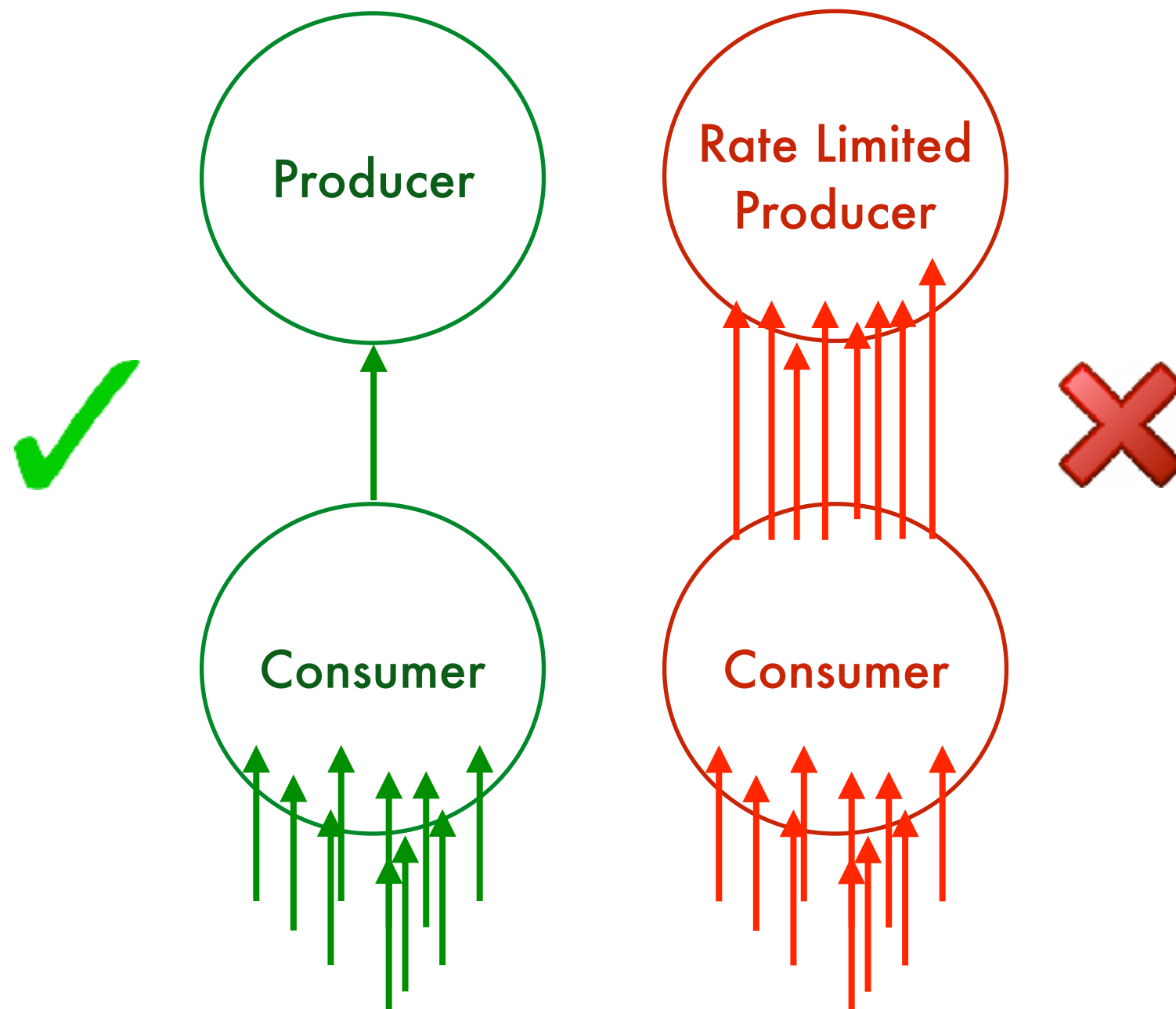




Rate Limiting



Rate Limiting

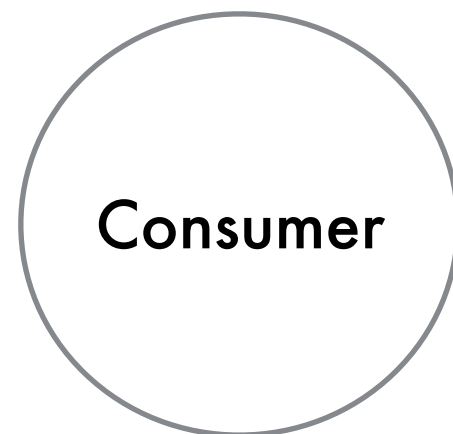


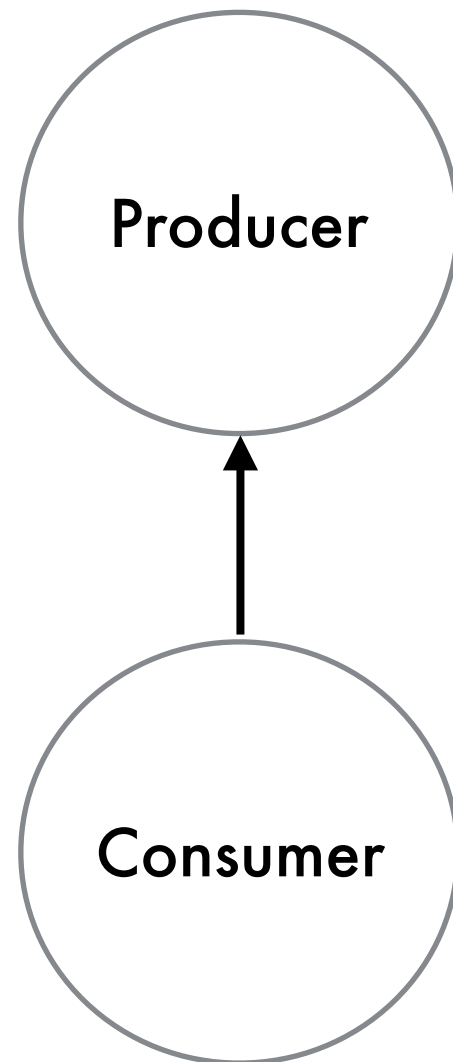
Flakey Dependencies

SpectoLabs

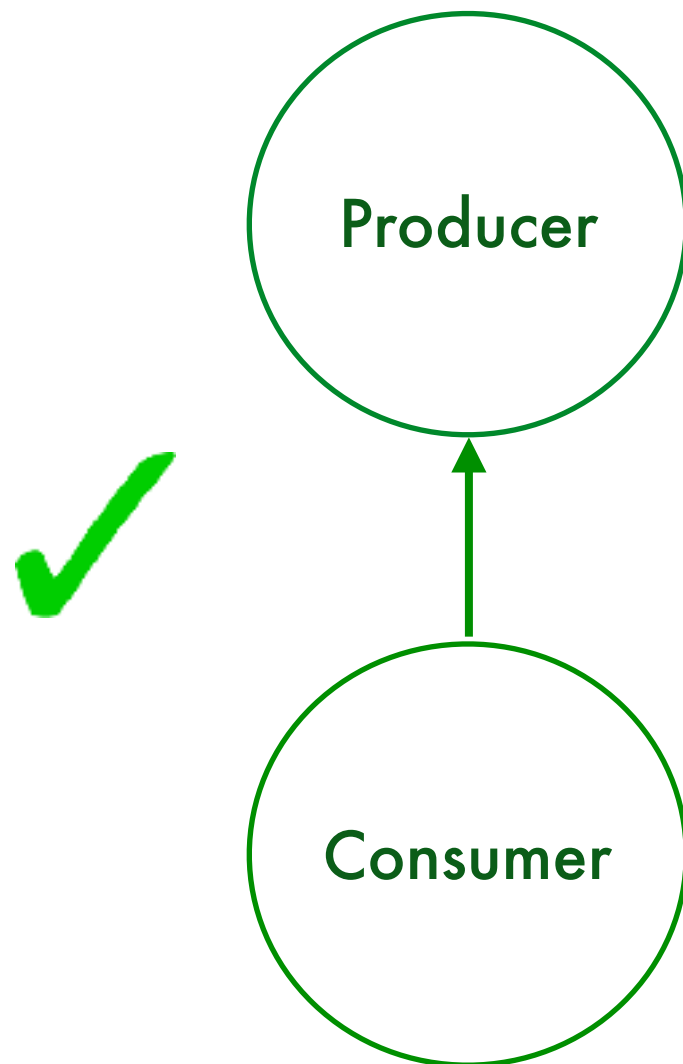
Flakey Dependencies

SpectoLabs

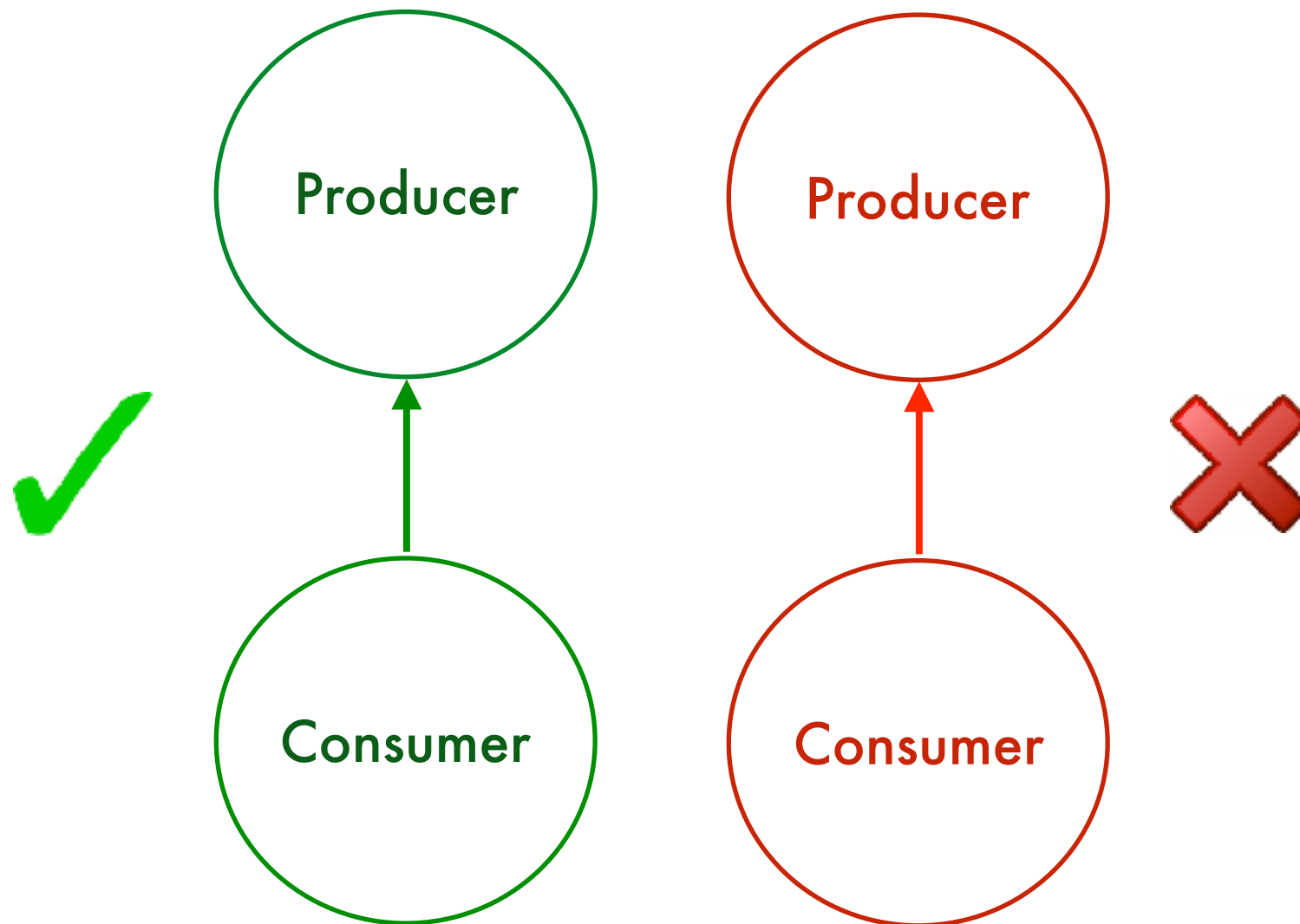




Flakey Dependencies



Flakey Dependencies

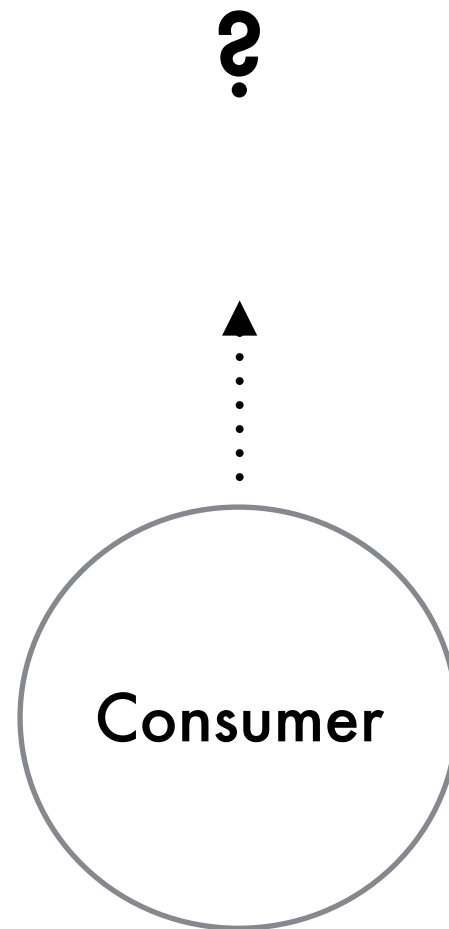


Doesn't exist

SpectoLabs

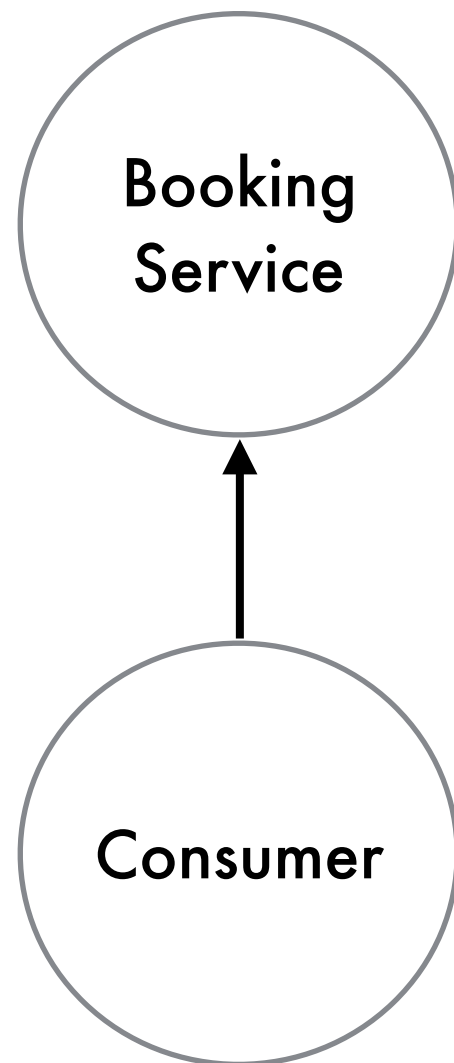
Doesn't exist

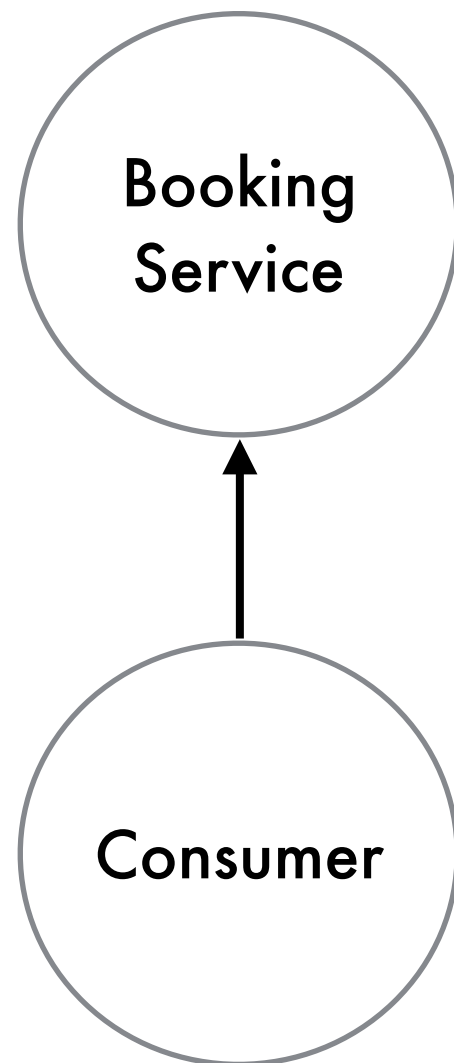
SpectoLabs



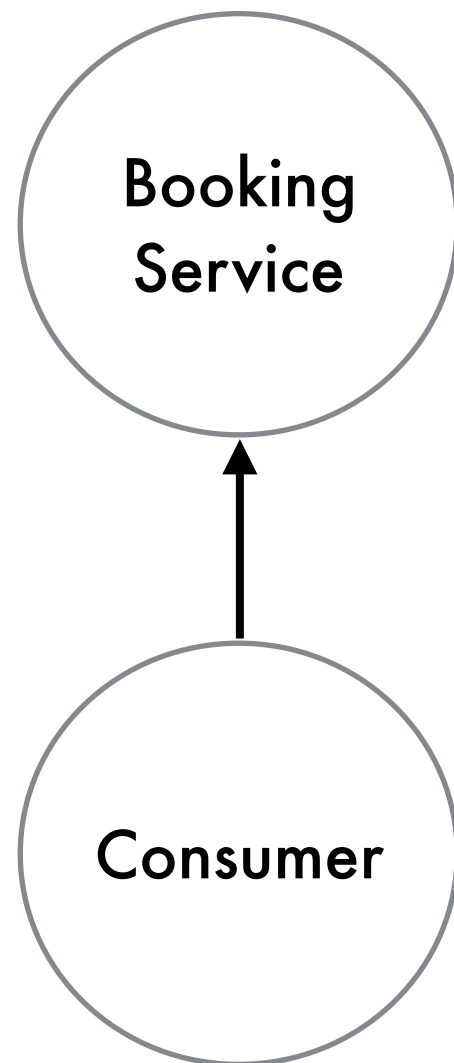
Non Deterministic

SpectoLabs



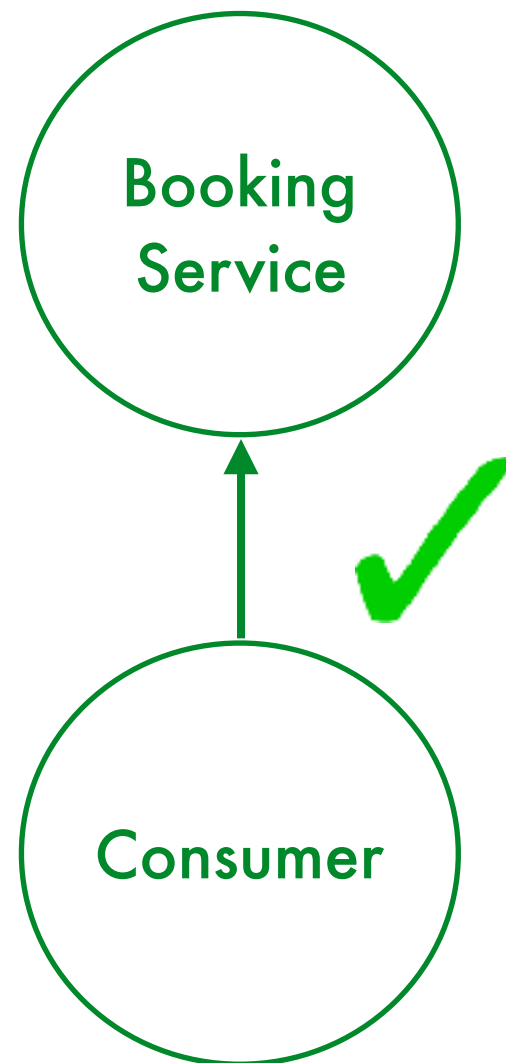


```
{  
  "date" : "2013-01-08",  
  ...  
}
```



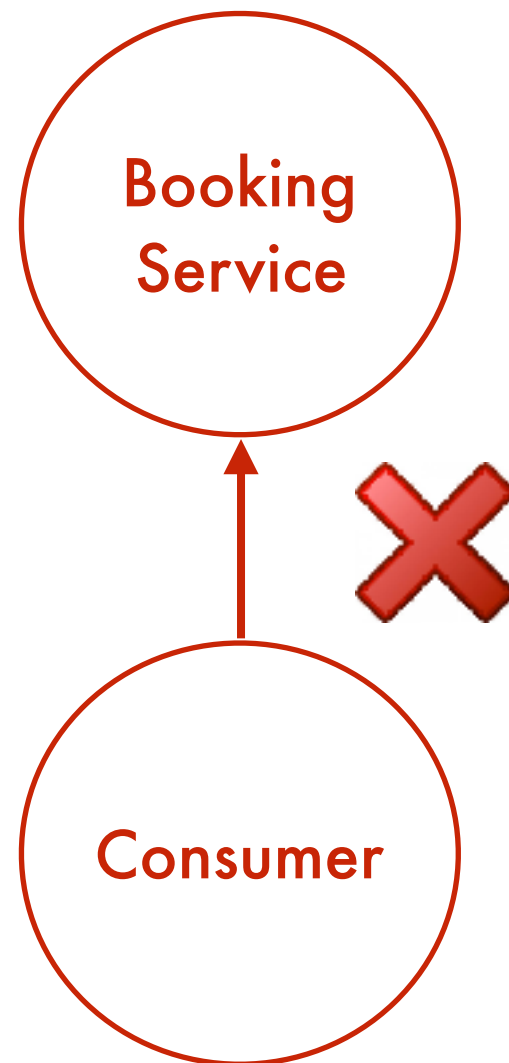
```
{  
    "date" : "2013-01-08",  
    ...  
}
```

```
assertThat(booking.getDate())  
    .isEqualTo(LocalDate.of(2013,1,8));
```



```
{  
    "date" : "2013-01-08",  
    ...  
}
```

```
assertThat(booking.getDate())  
    .isEqualTo(LocalDate.of(2013,1,8));
```



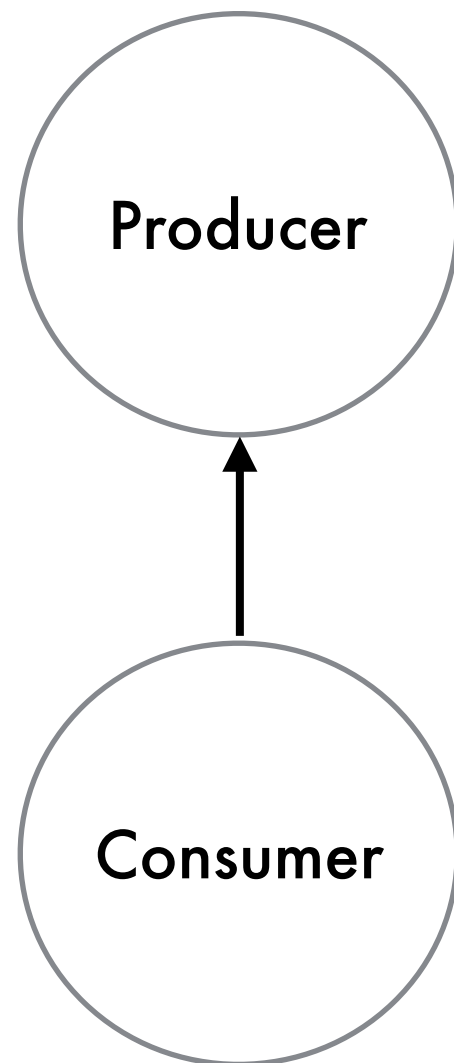
```
{  
    "date" : "2016-02-07",  
    ...  
}
```

```
assertThat(booking.getDate())  
    .isEqualTo(LocalDate.of(2013,1,8));
```

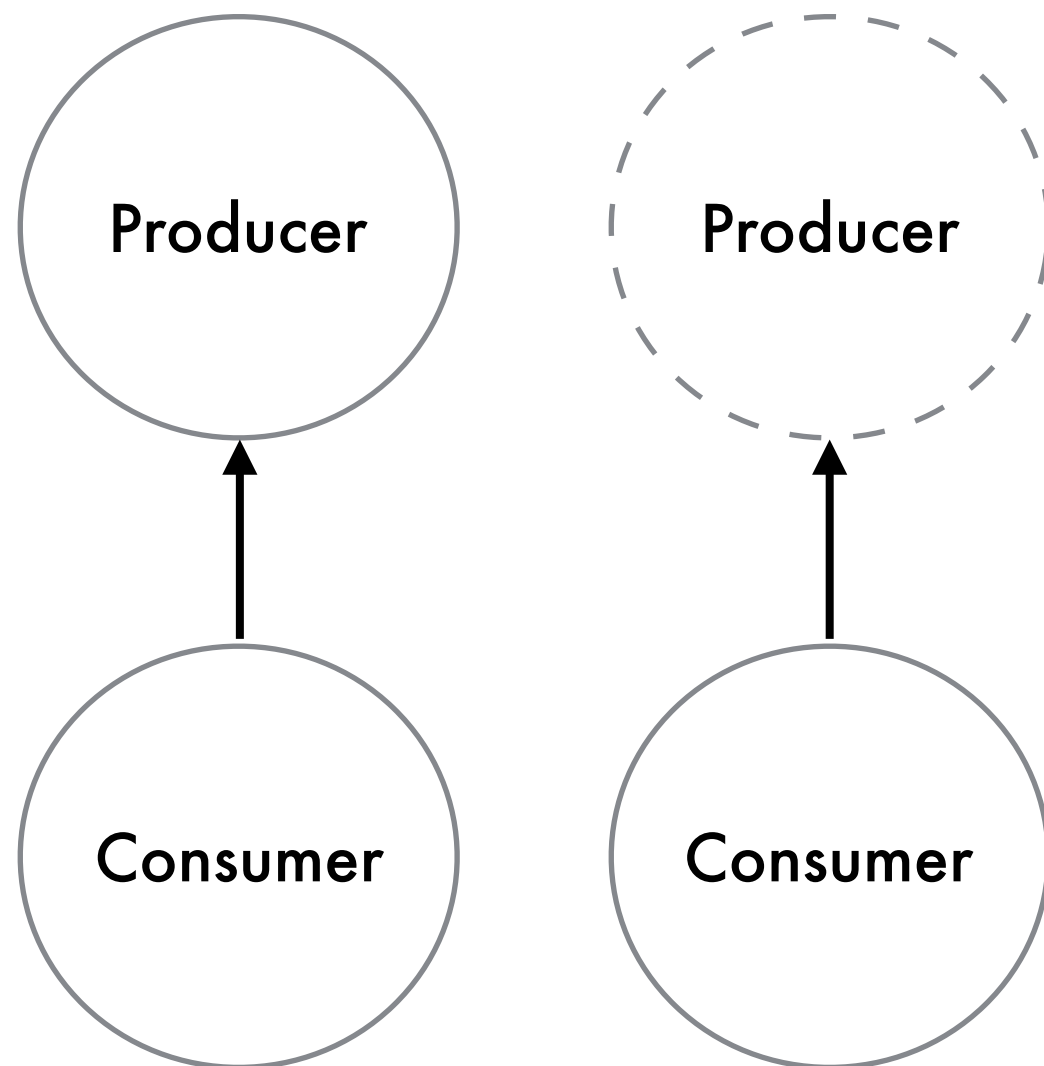
Fault Injection

SpectoLabs

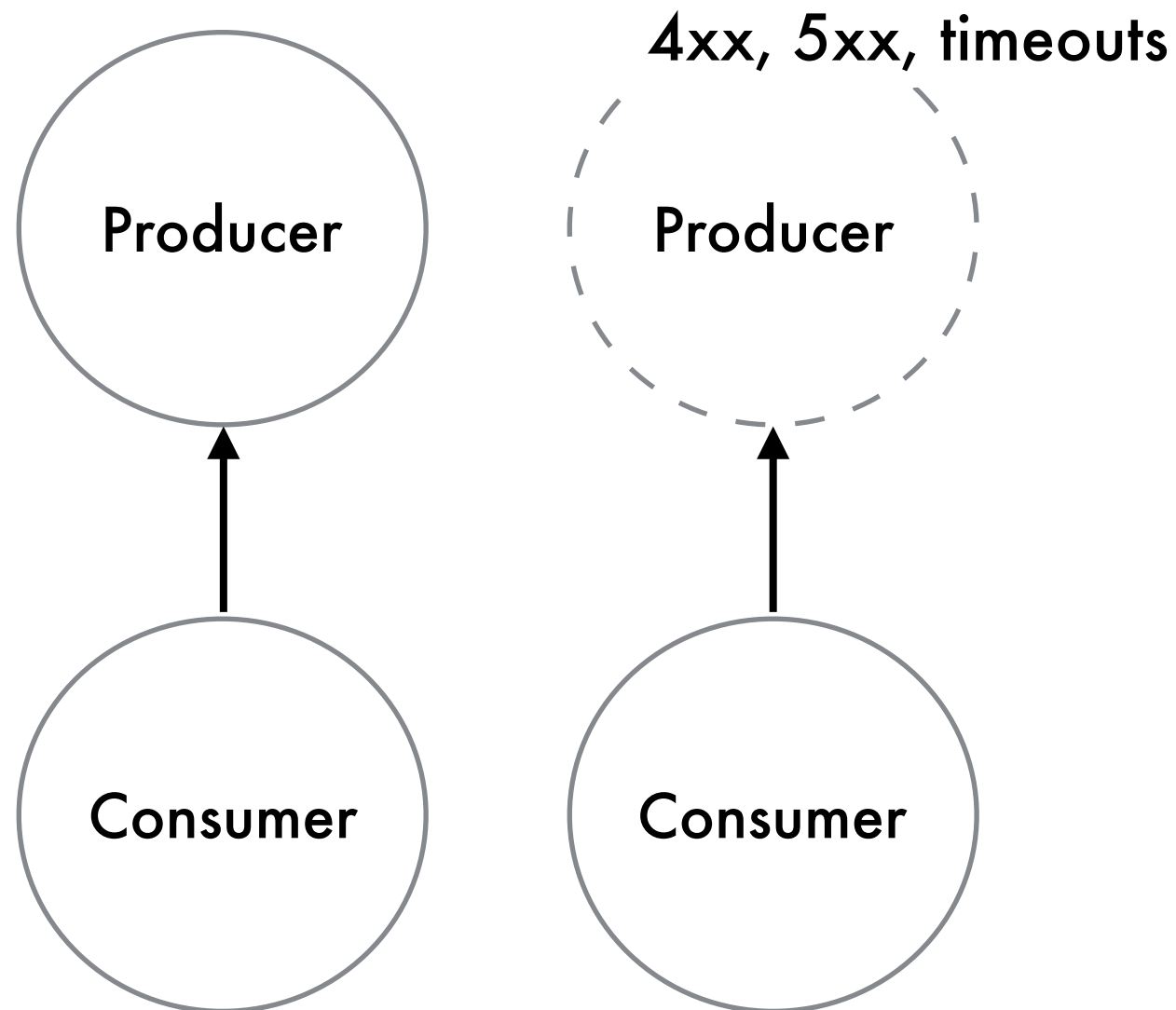
Fault Injection

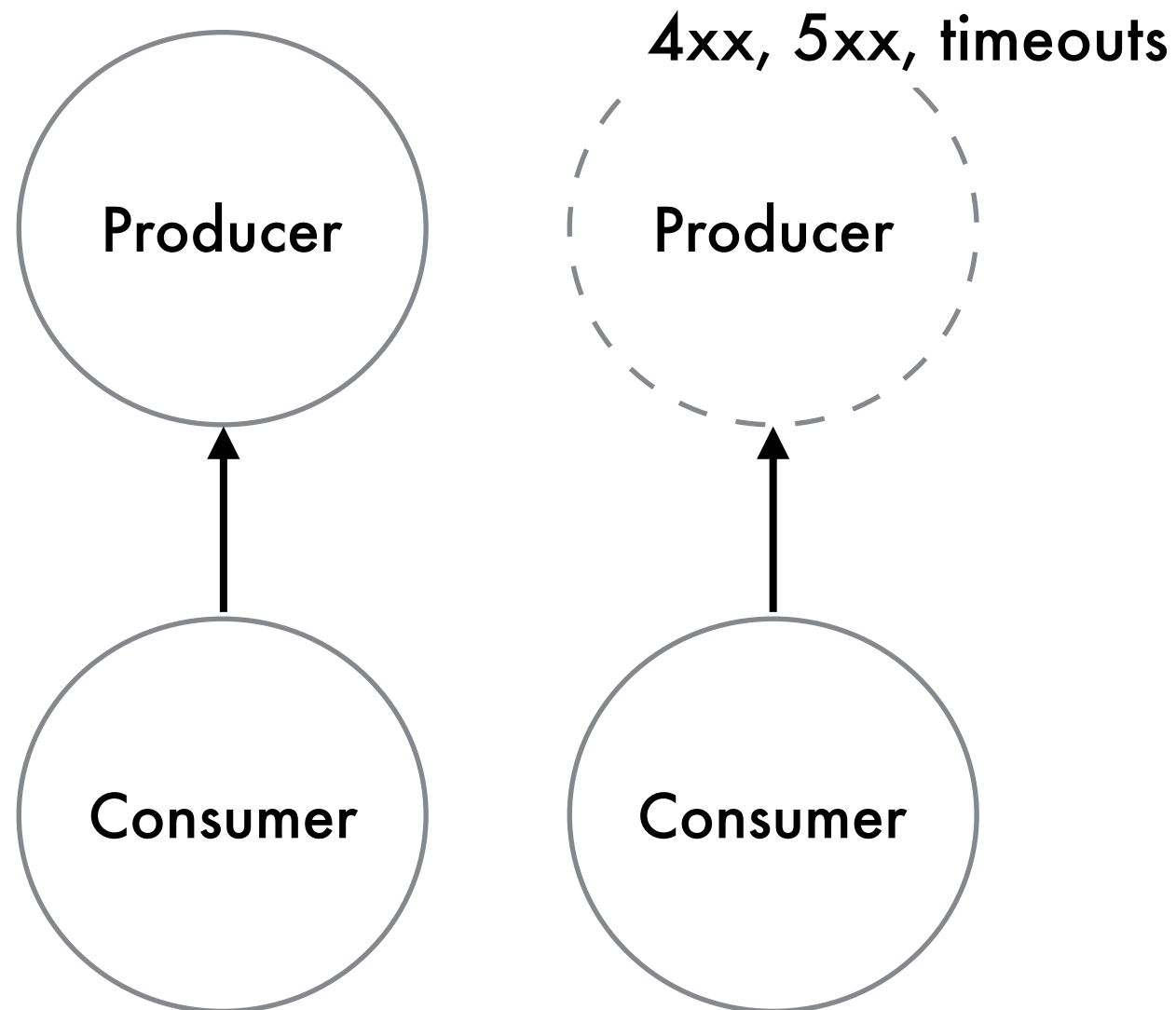


Fault Injection



Fault Injection





Operationally
challenging to trigger
faults

Agenda

- Why we need API simulation
- Introduction to API simulation
- Hoverfly Demonstration
- Simulating Your First API

- Emulation of real API

API Simulation

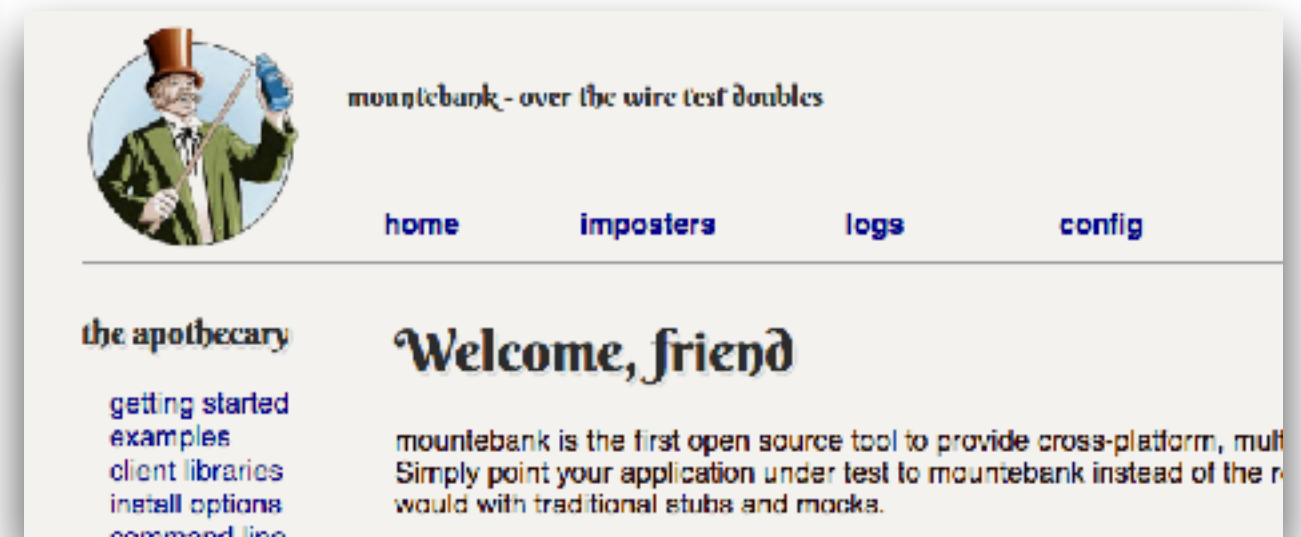
- Emulation of real API
- Non-intrusive

API Simulation

- Emulation of real API
- Non-intrusive
- Lightweight - or should be

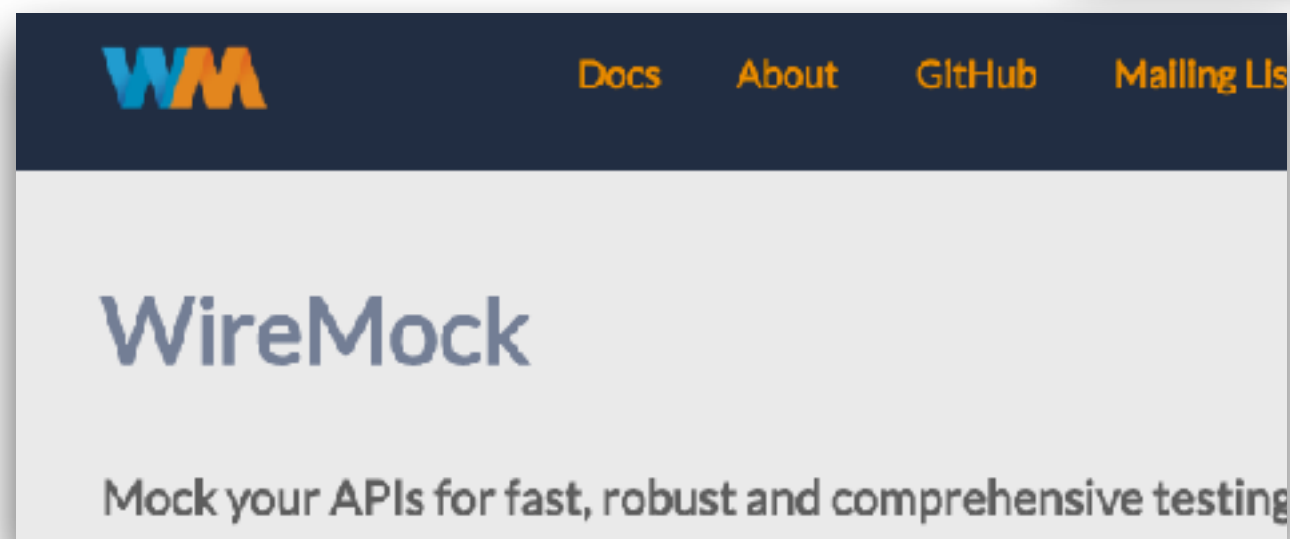
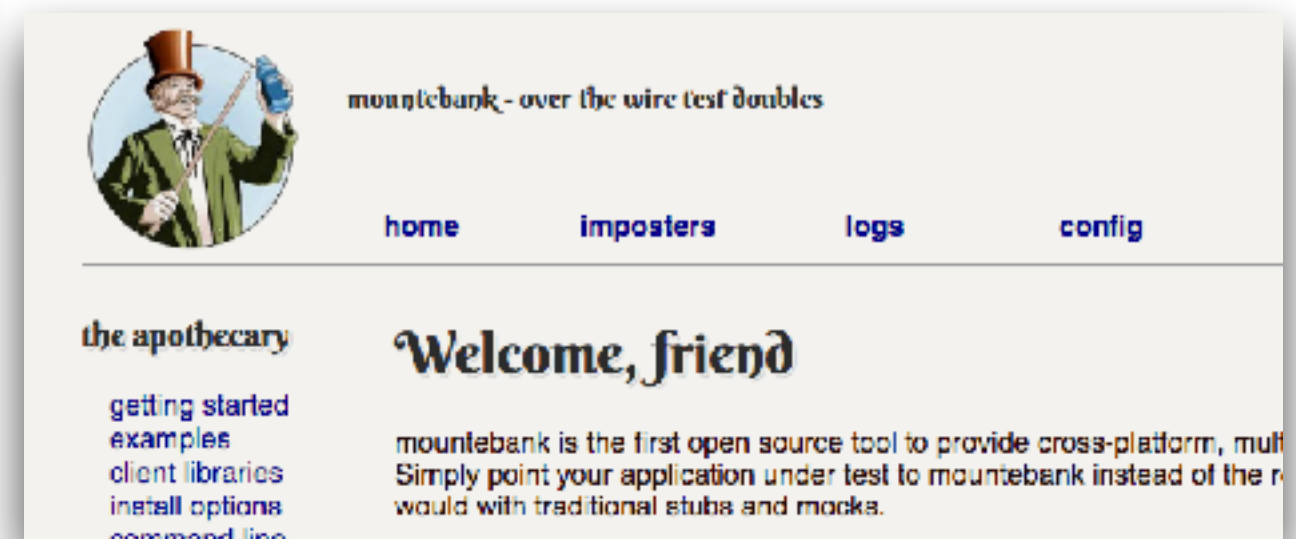
Open Source Tooling



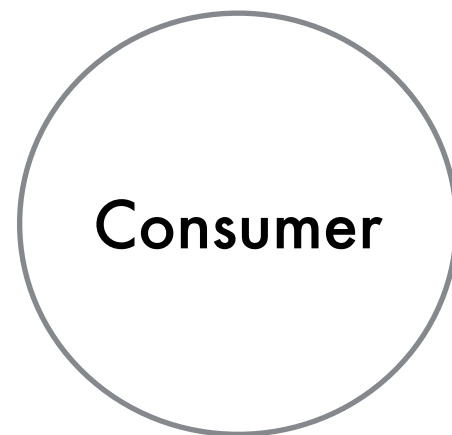


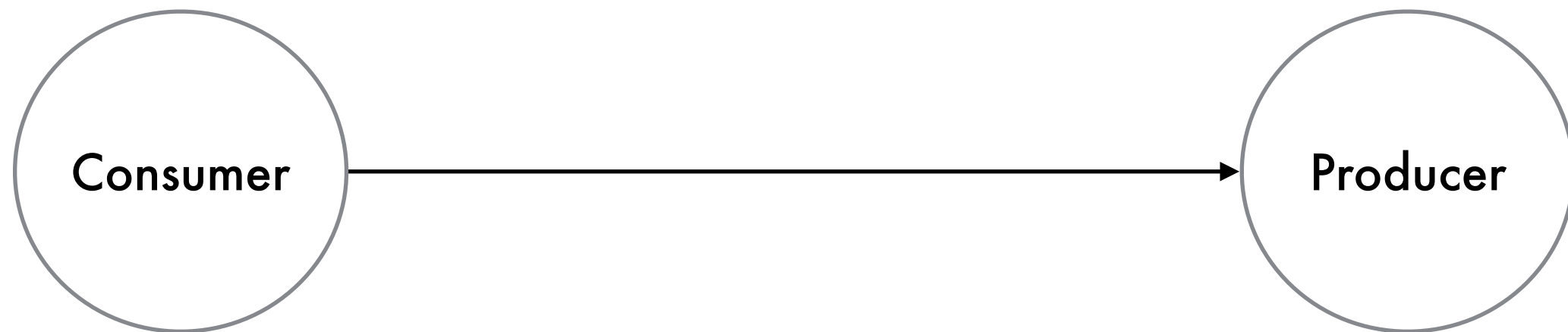
Open Source Tooling

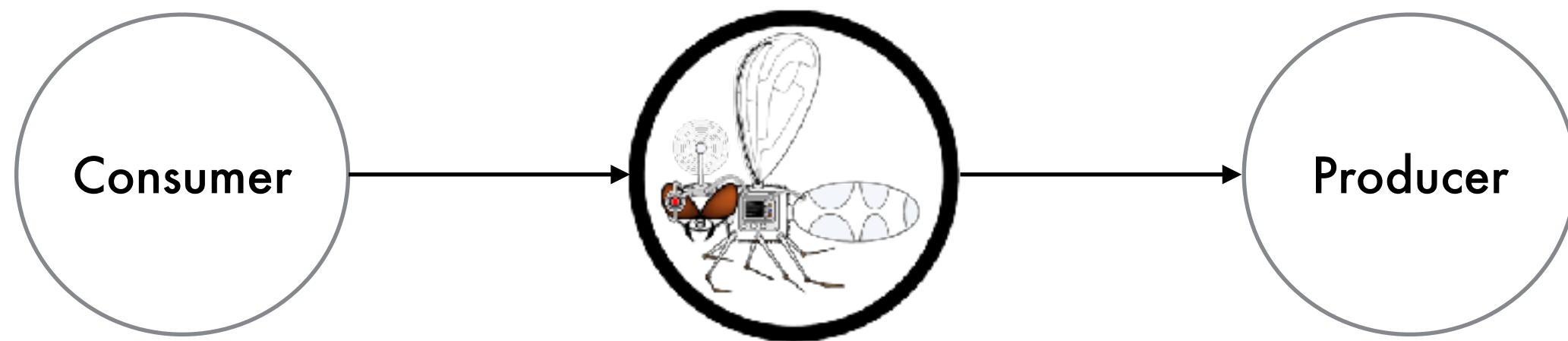
SpectoLabs



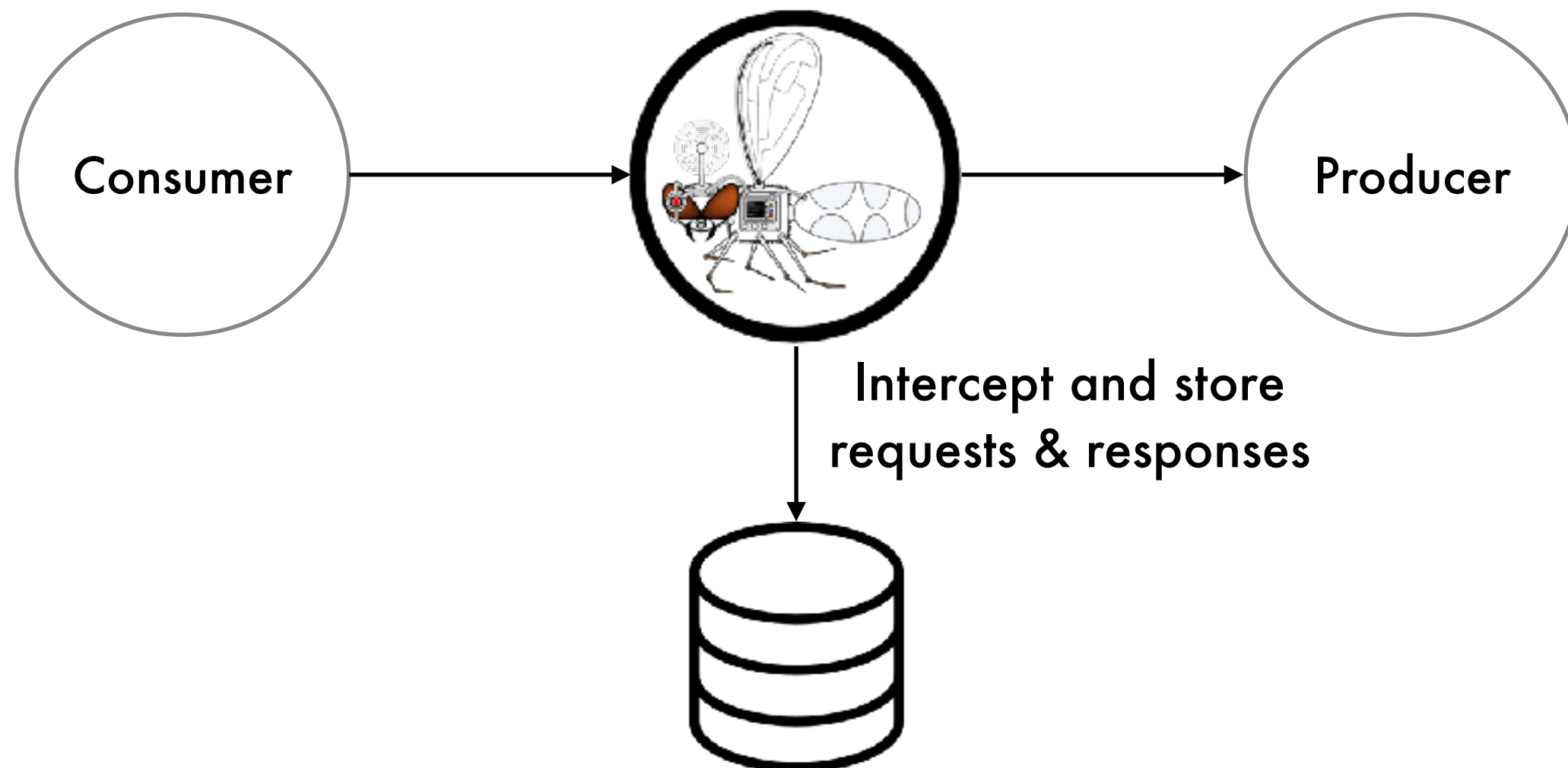




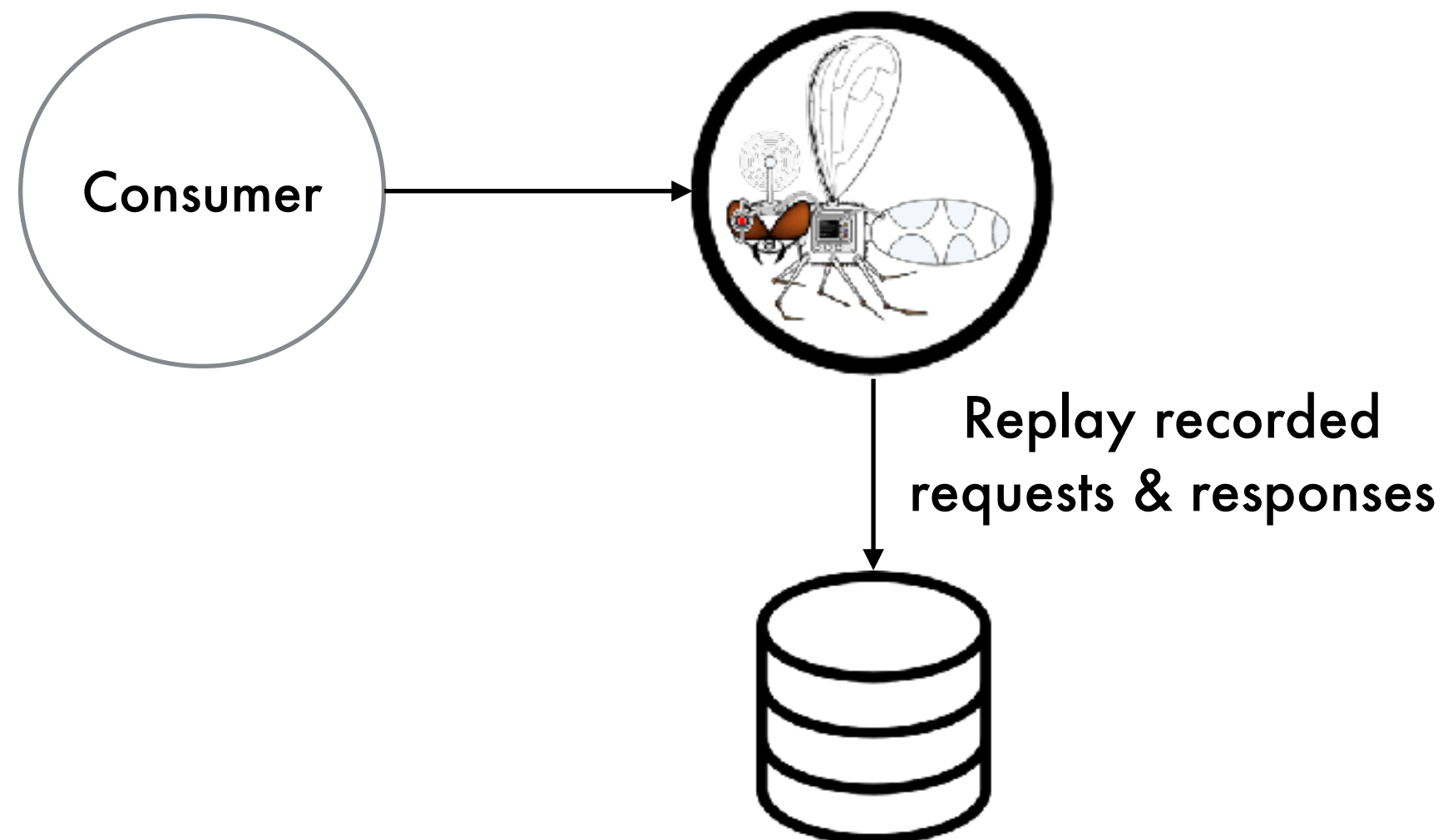




Capture Mode



Simulate Mode



Creating Data

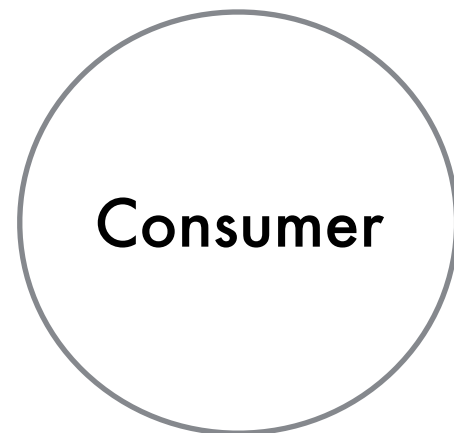
```
@ClassRule
public static HoverflyRule hoverflyRule = HoverflyRule.inSimulateMode(dsl(
    service("www.booking-service.com")
        .get("/api/bookings/1")
        .willReturn(success(json(new Booking(1))))
        .post("/api/bookings")
        .body(json(new Booking("myBooking")))
        .willReturn(created("http://www.booking-service.com/api/bookings/1")),
    service("www.payment-service.com")
        .get("/api/payments/1")
        .willReturn(success("{\"amount\": \"1.25\"\"}")),
));
```

Creating Data

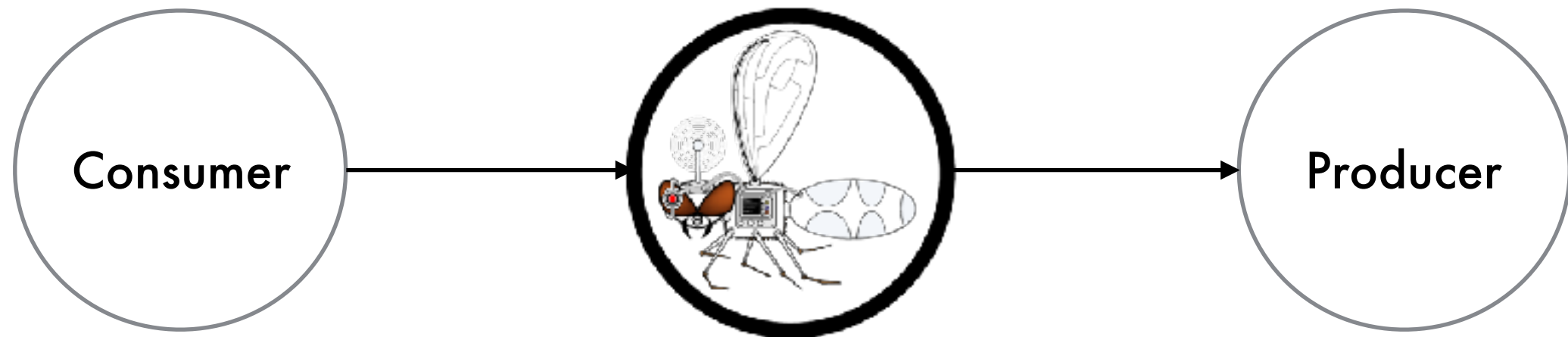
```
{
  "data": {
    "pairs": [
      {
        "response": {
          "status": 200,
          "body": "{\"stats\":{\"counters\":{\"capture\":0,\"modify\":0,\"simulate\""},
        },
        "request": {
          "requestType": "recording",
          "path": "/api/stats",
          "method": "GET",
          "destination": "localhost:60603",
          "scheme": "http",
          "query": "",
          "body": "",
          "headers": {
            "Accept": [
              "text/html, image/gif, image/jpeg, *; q=.2, */*; q=.2"
            ],
            "Proxy-Connection": [
              "keep-alive"
            ],
            "User-Agent": [
              "Java/1.8.0_40"
            ]
          }
        }
      }
    ]
  }
}
```

Forward Proxy

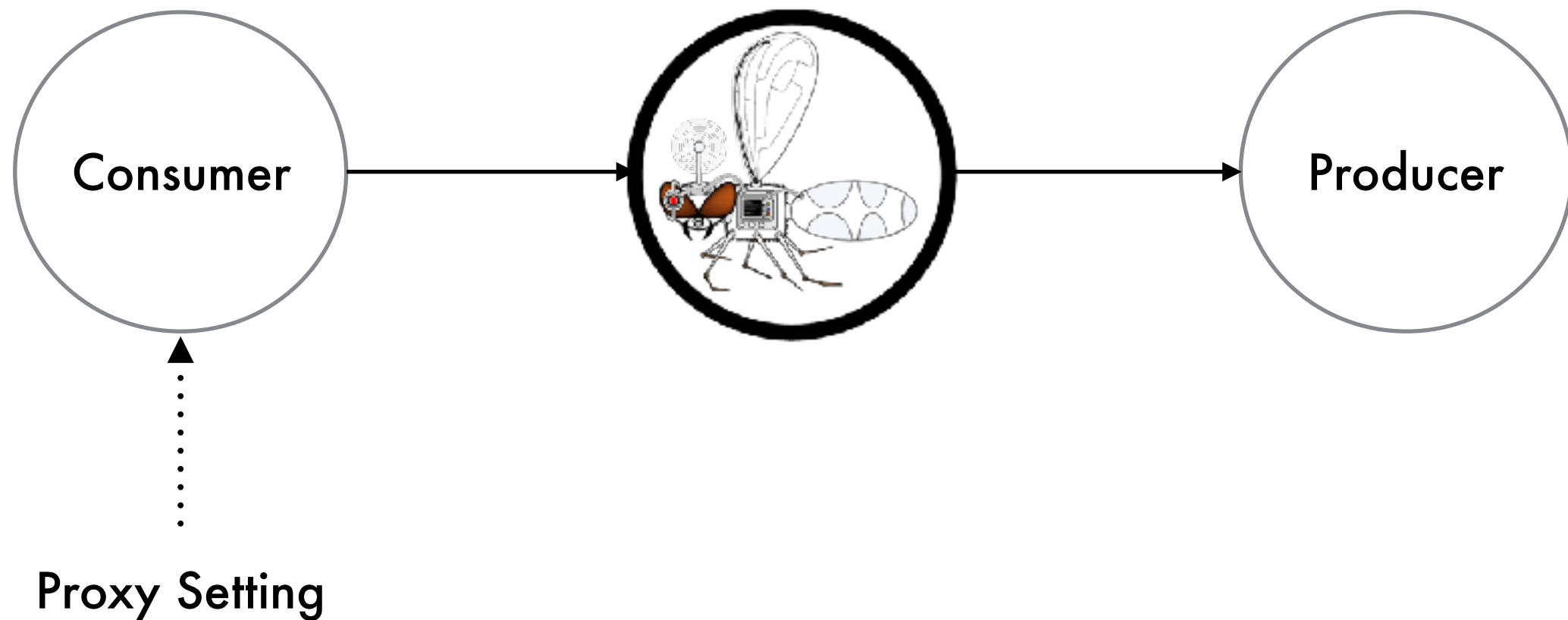
SpectoLabs



Forward Proxy



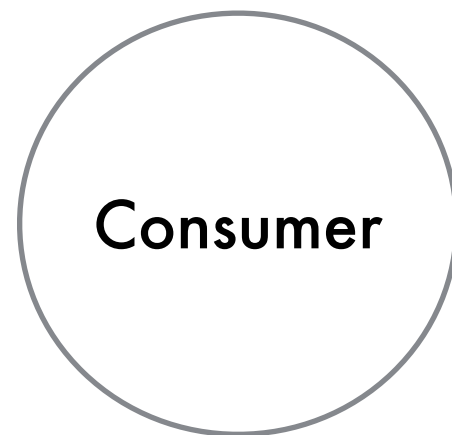
Forward Proxy

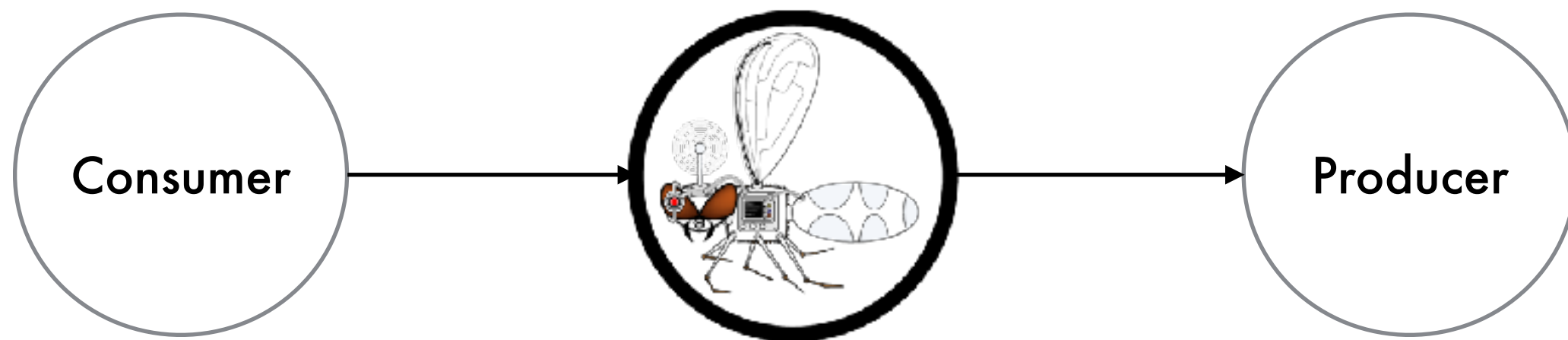


Reverse Proxy

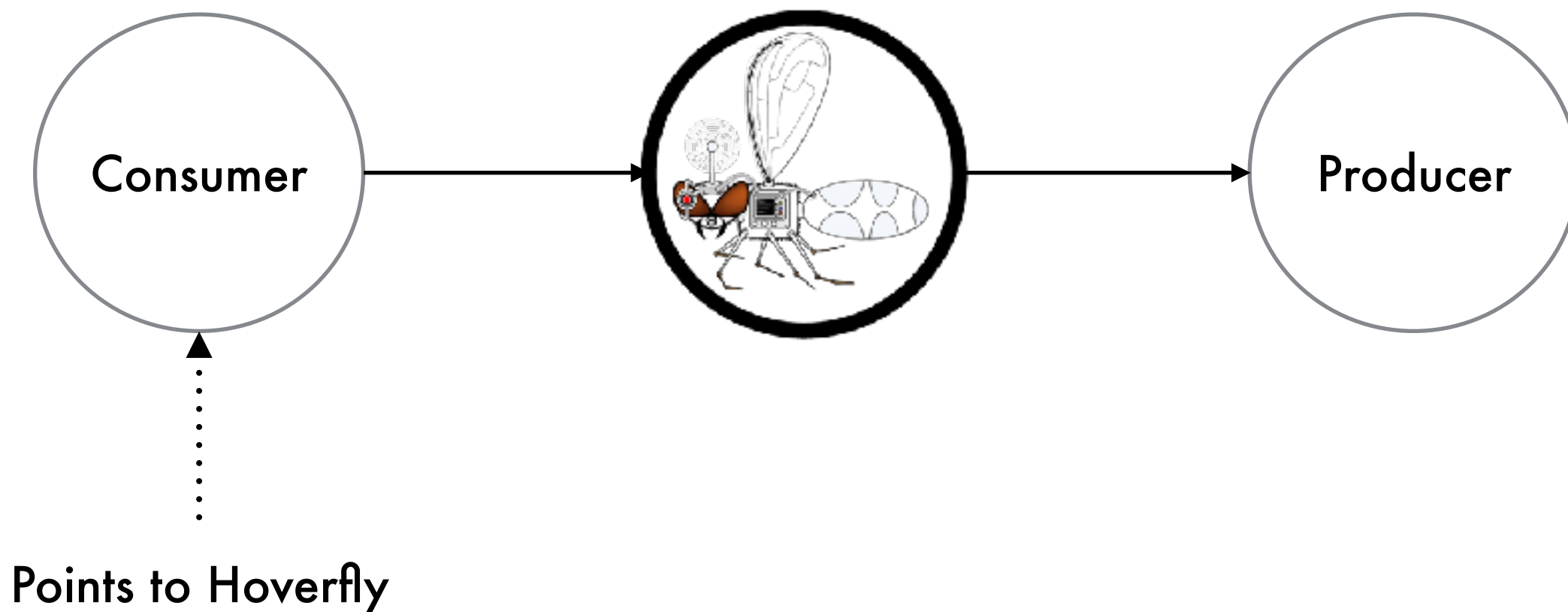
SpectoLabs

Reverse Proxy

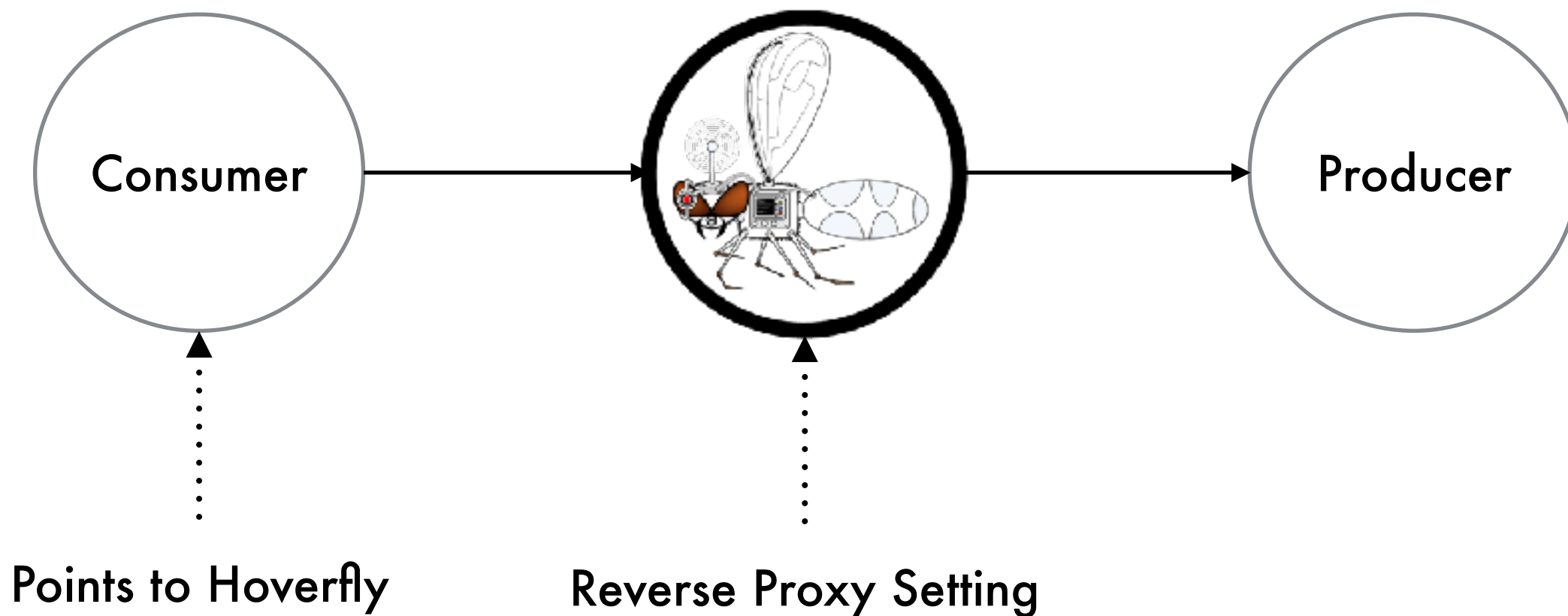




Reverse Proxy



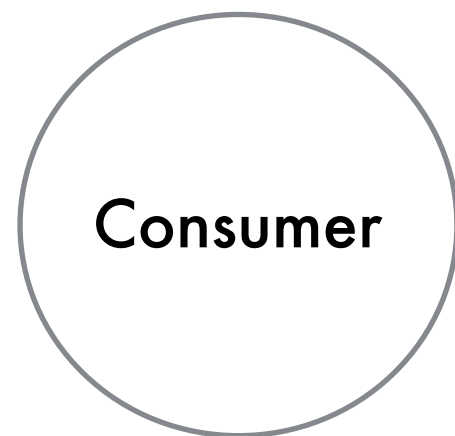
Reverse Proxy



Record and replay

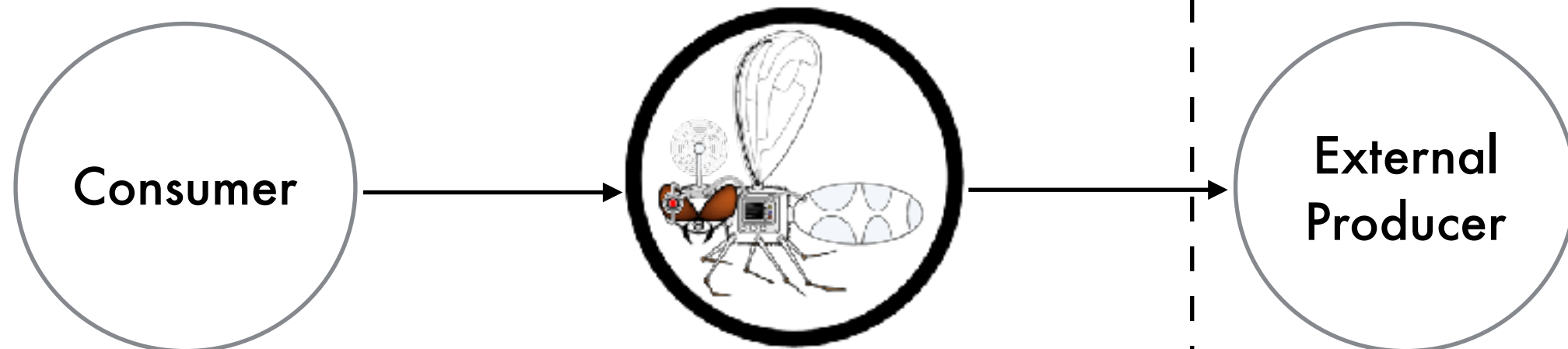
SpectoLabs

Record and replay



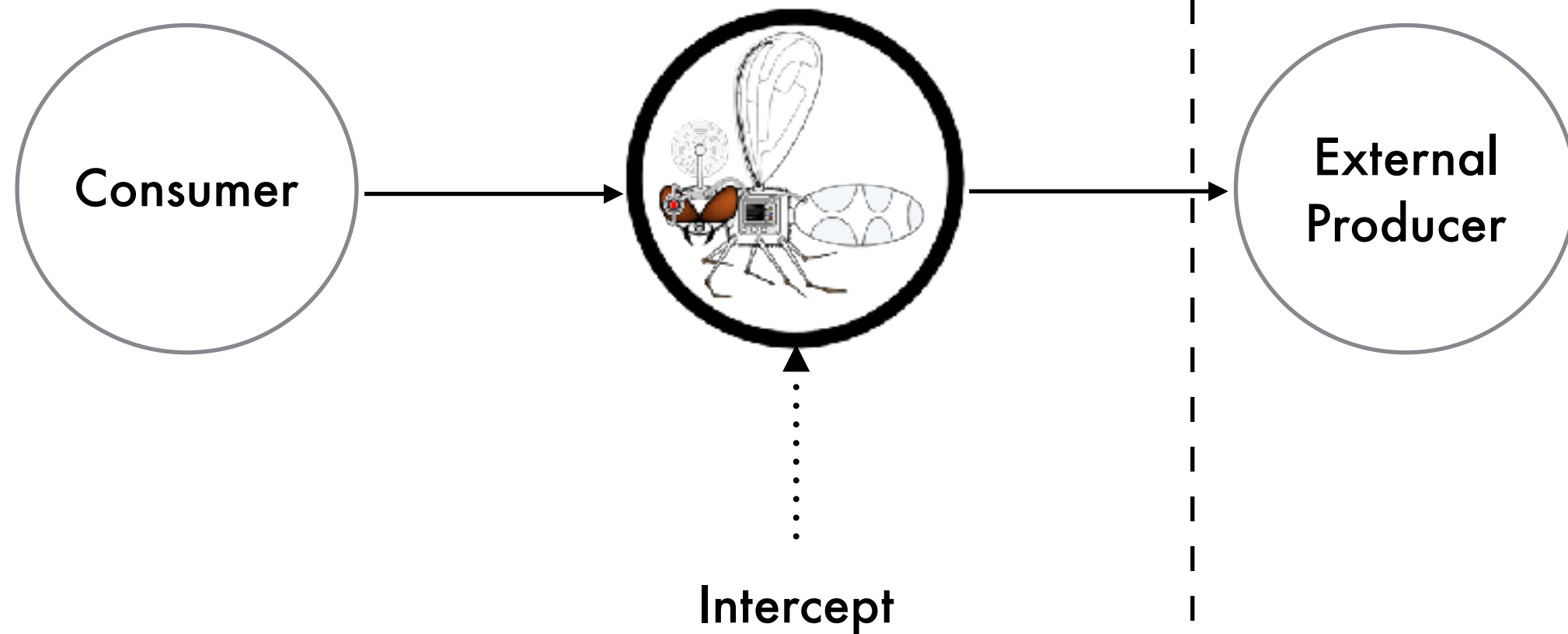
Record and replay

SpectoLabs



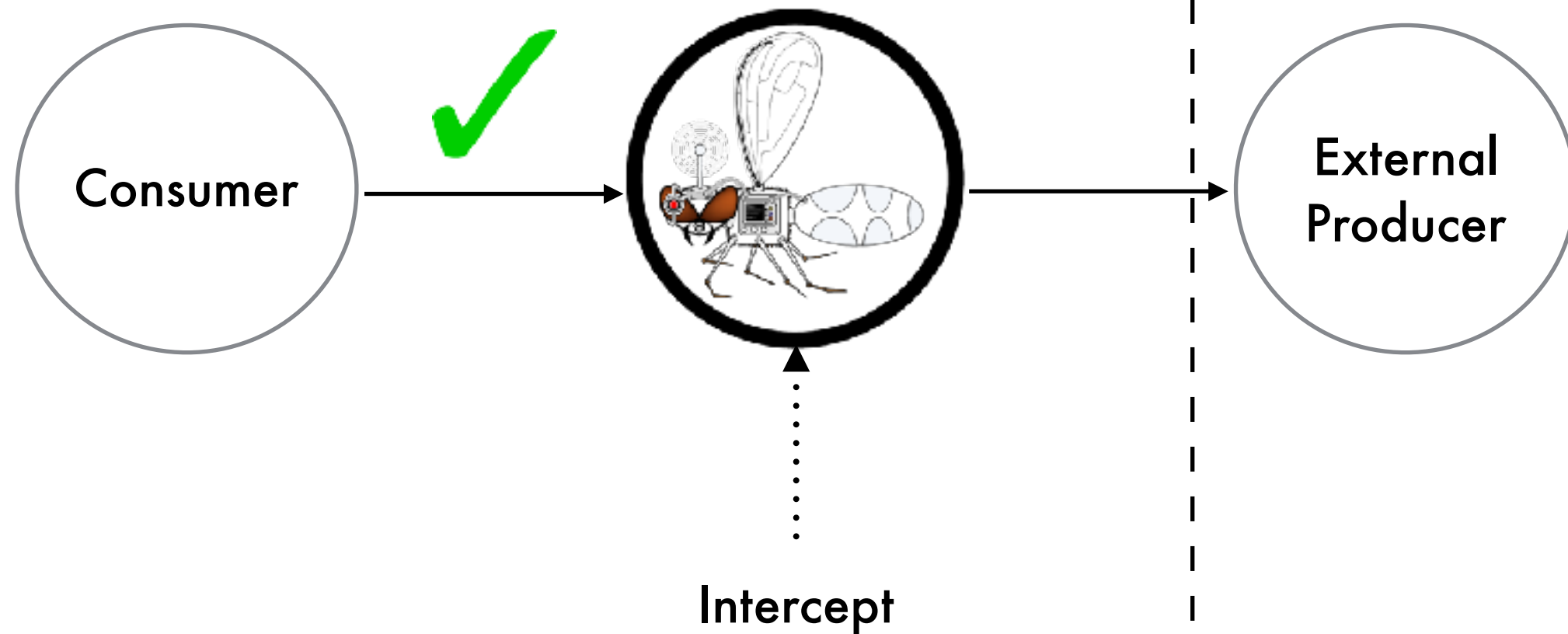
Record and replay

SpectoLabs

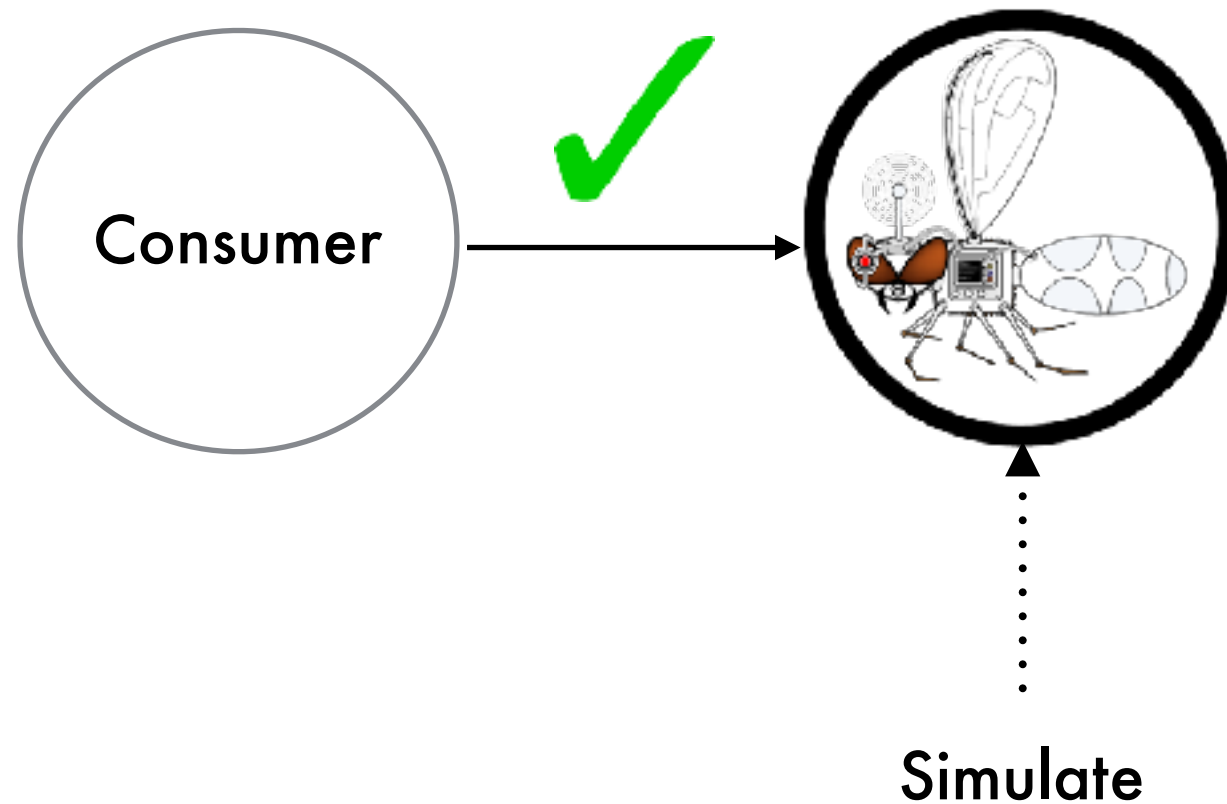


Record and replay

SpectoLabs



Record and replay



Advantages

SpectoLabs

Advantages

- Isolation

Advantages

- Isolation
- Flaky dependencies

Advantages

- Isolation
- Flaky dependencies
- Rate limiting

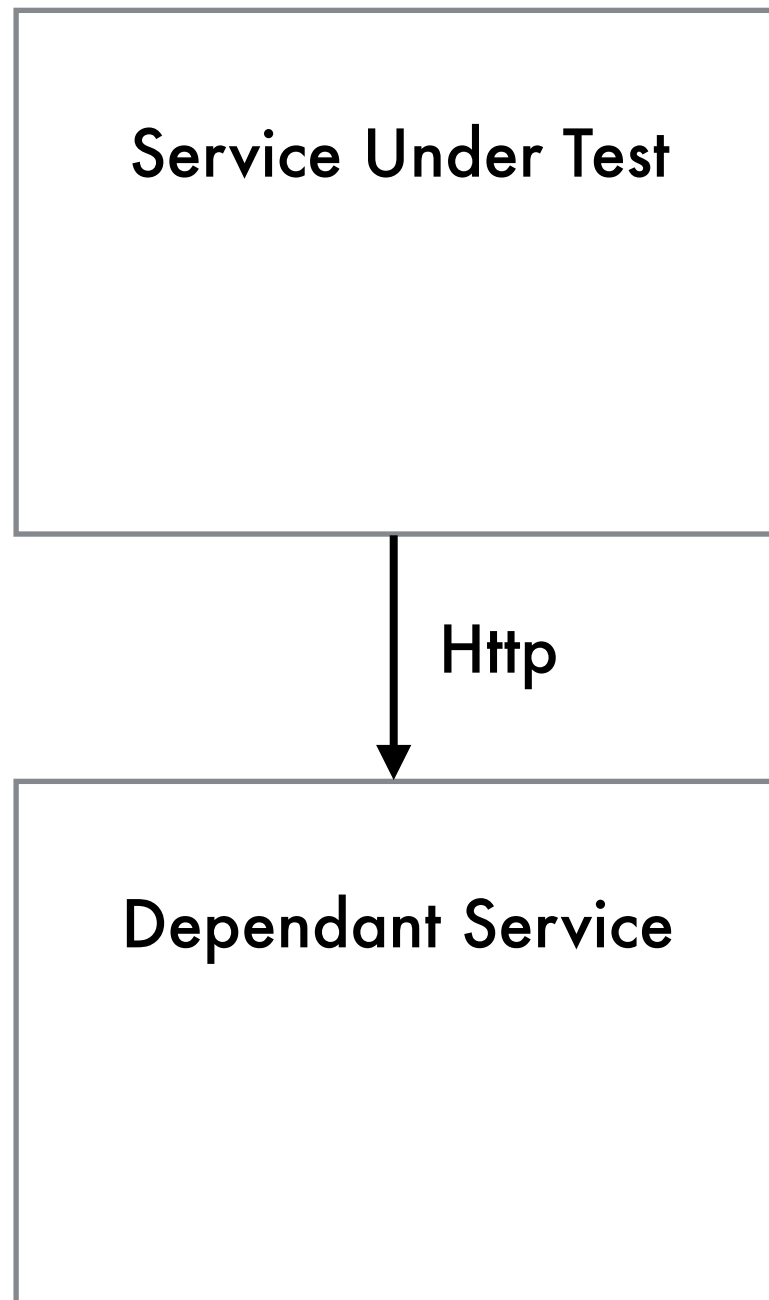
Advantages

- Isolation
- Flaky dependencies
- Rate limiting
- Deterministic

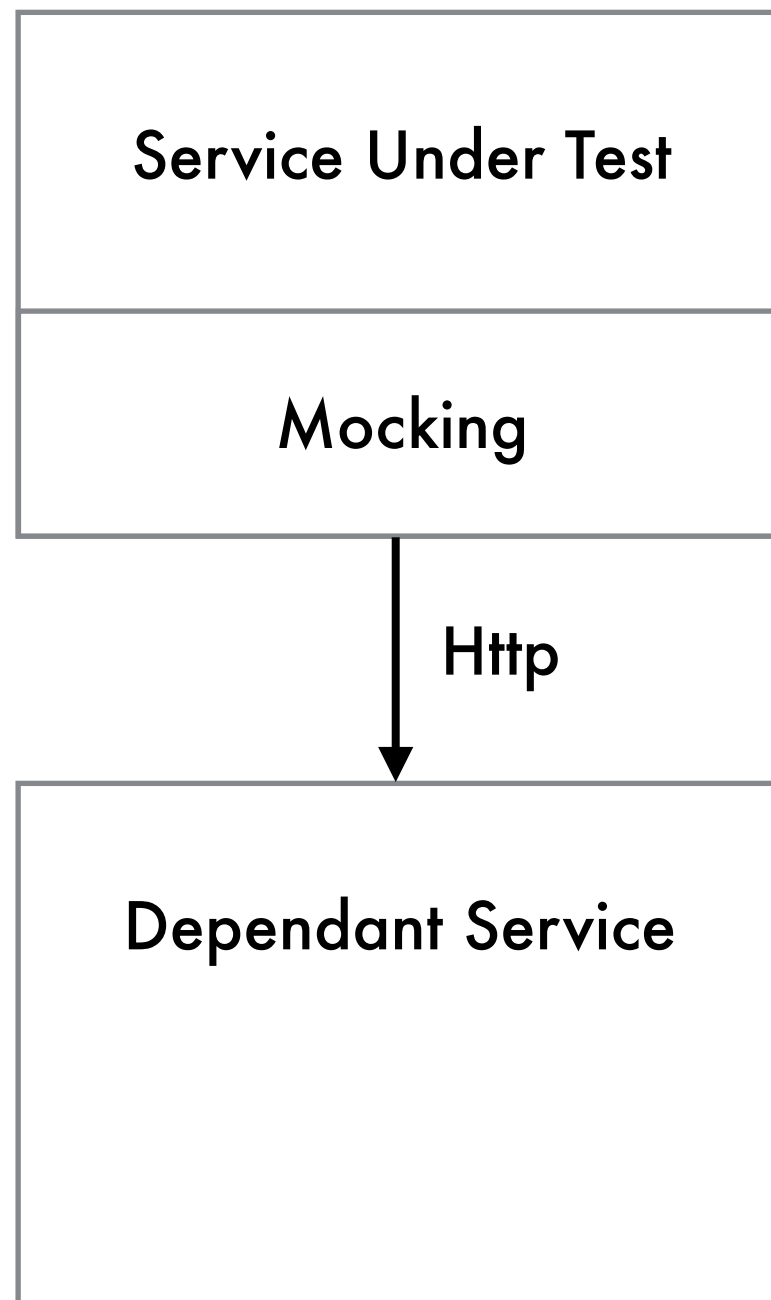
Advantages

- Isolation
- Flaky dependencies
- Rate limiting
- Deterministic
- Licensing

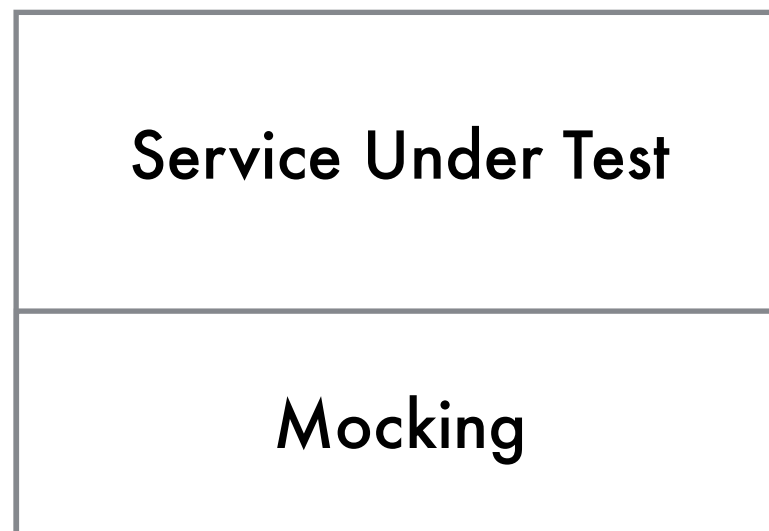
Not Mocking



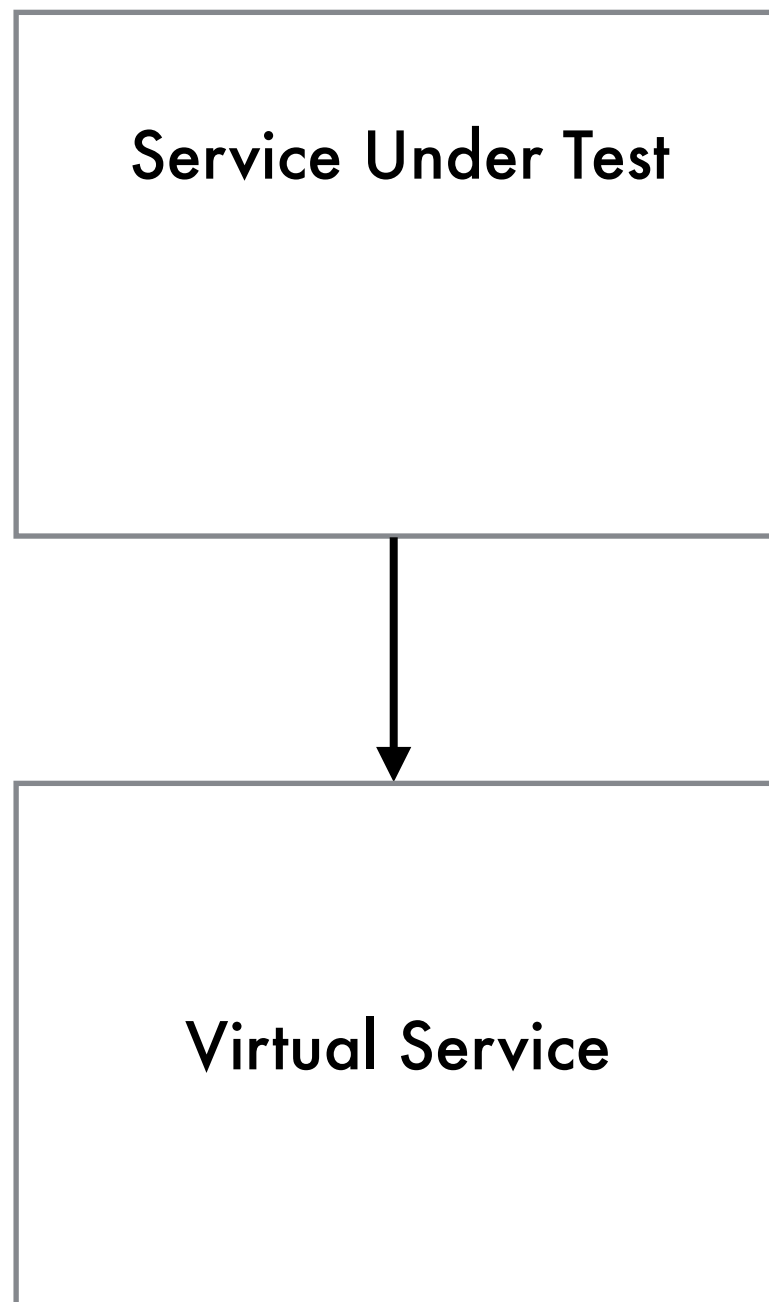
Not Mocking

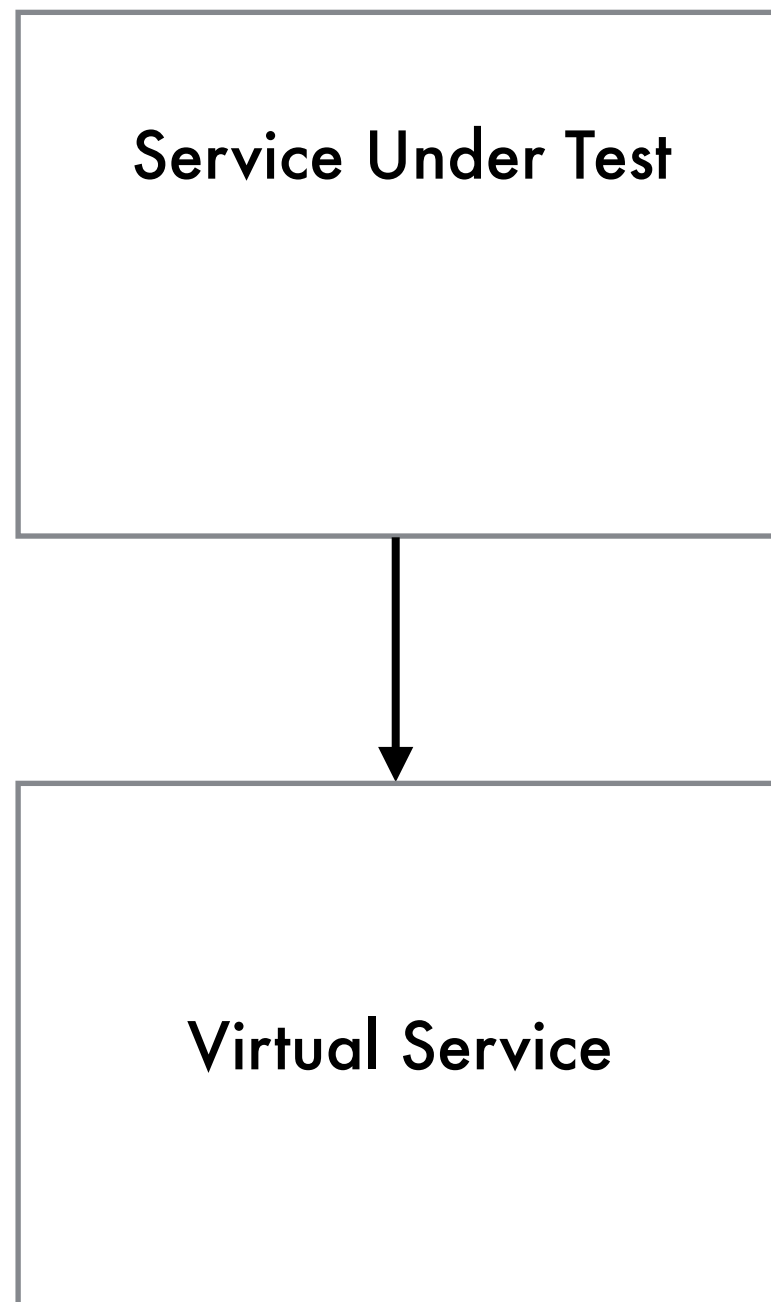


Not Mocking



Not Mocking





**Mocks bleed into the
architectural boundary of
your application**

Agenda

- Why we need API simulation
- Introduction to API simulation
- **Hoverfly Demonstration**
- Exercise One: Simulating Your First API

Agenda

- Why we need API simulation
- Introduction to API simulation
- Hoverfly Demonstration
- Exercise One: Simulating Your First API