SkładQA #3

Kraków, 2018 - Mateusz Bryła
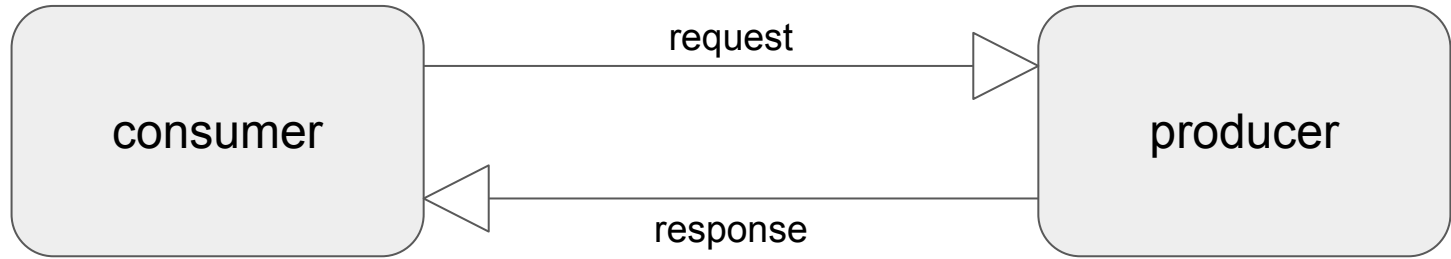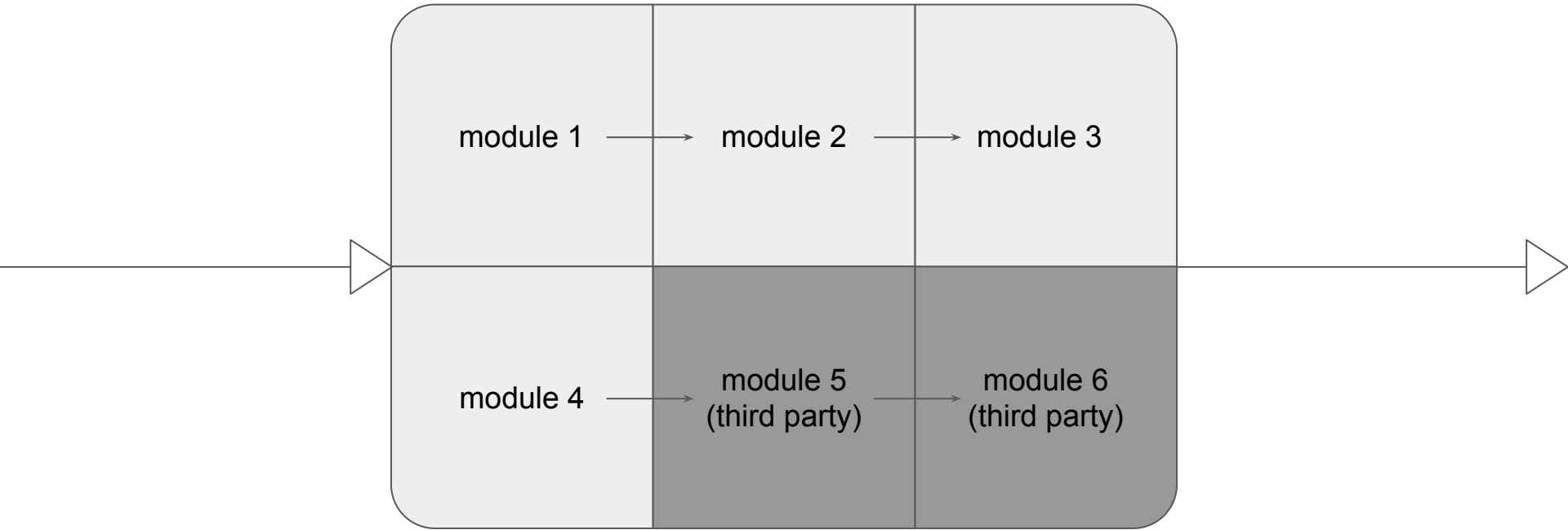
Contract Testing

Agenda

- Baseline

- Contract
- Contract Testing

- Why (not)?

- Tools
- Example
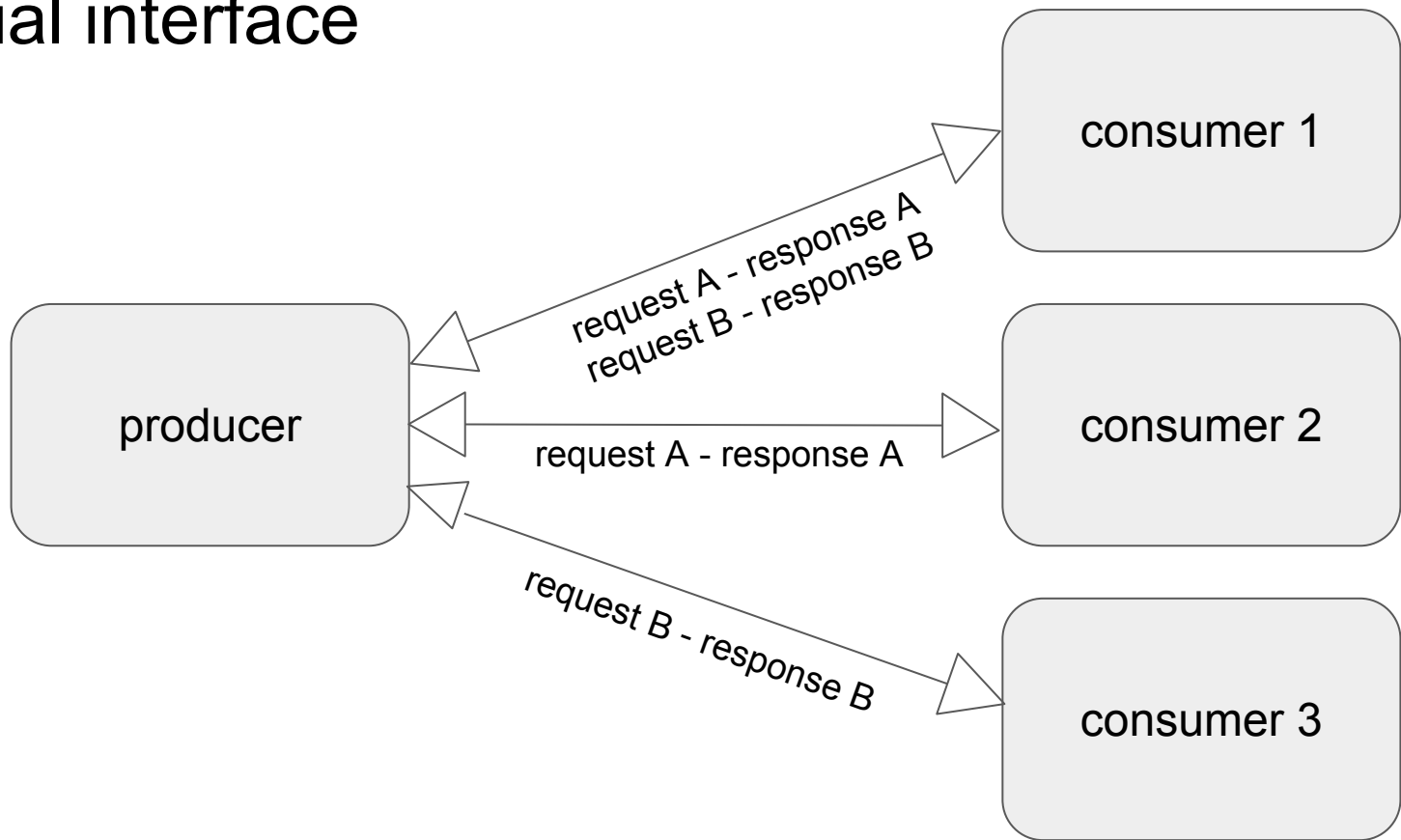
Baseline

Interface

Interfaces within a service

Partial interface
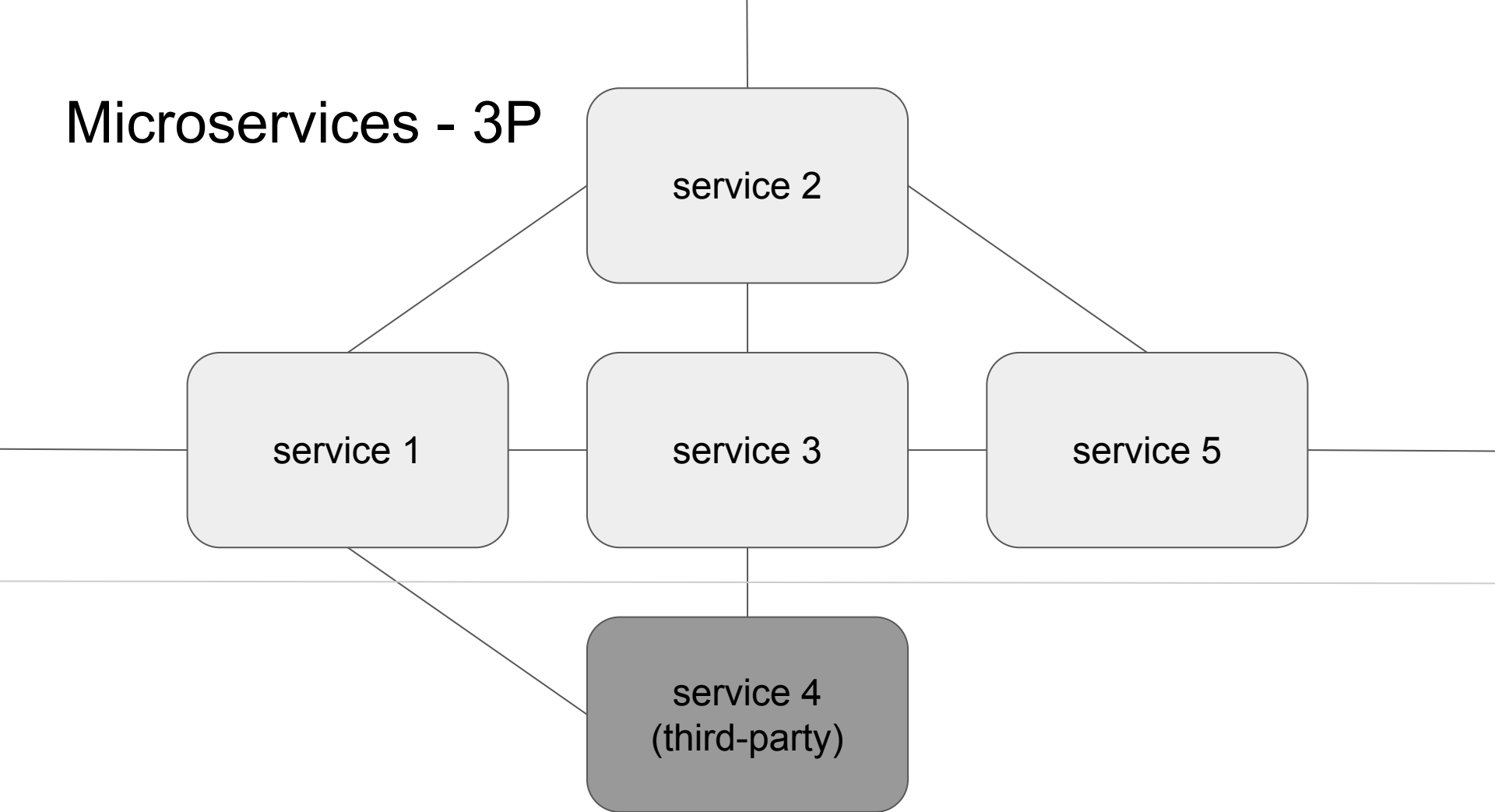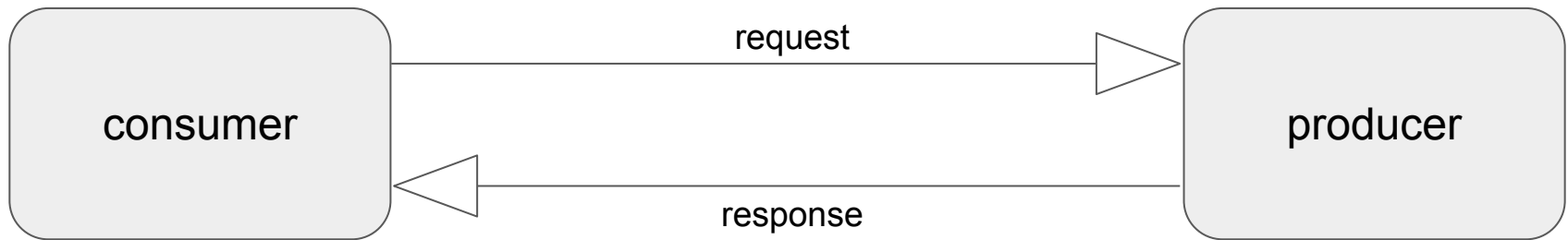
Microservices

service 2

service 1

service 3

service 5

service 4

Microservices - 3P

service 2

service 1

service 3
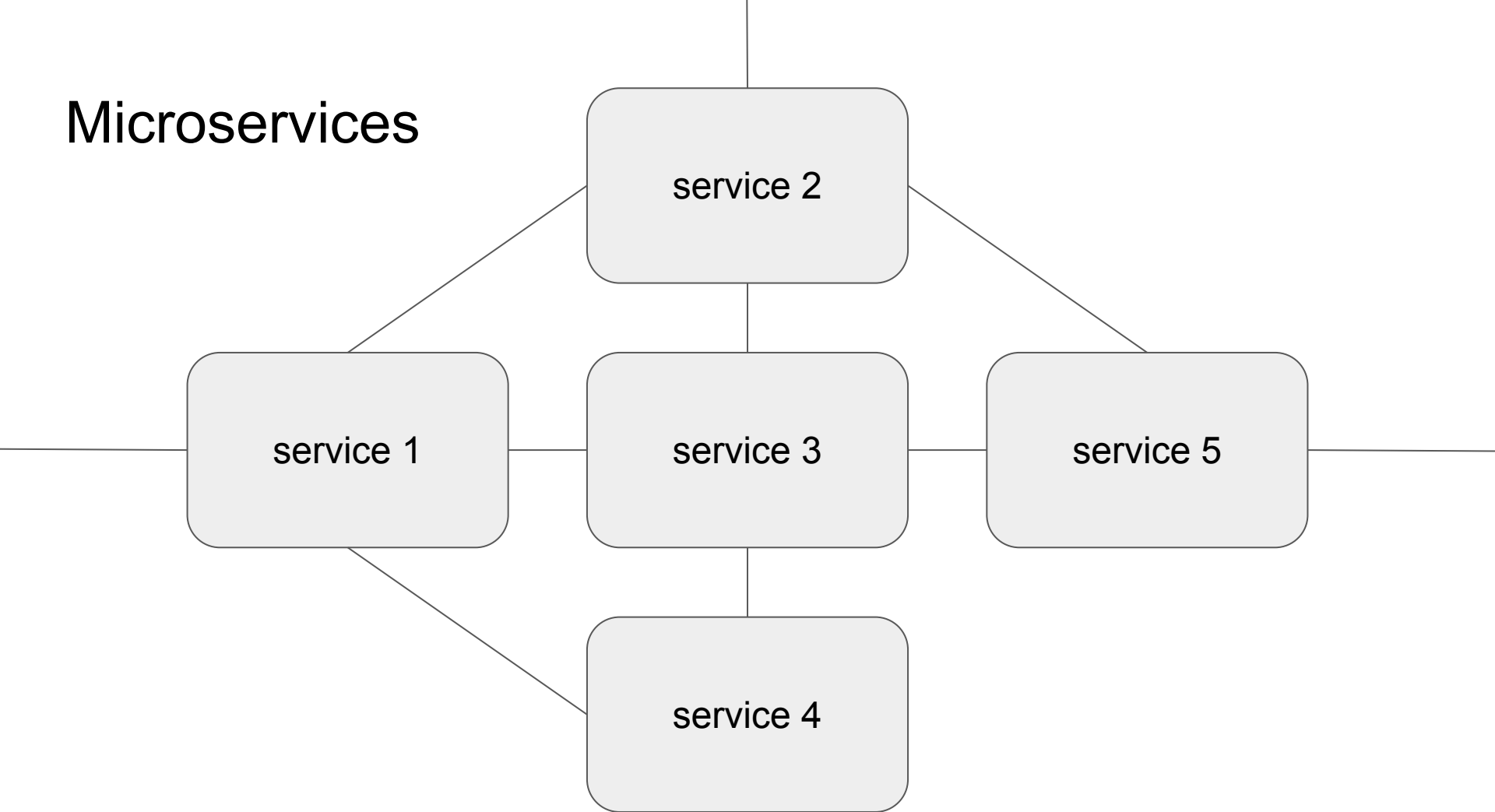
service 5

service 4
(third-party)

Contract

Interface

Contract

```
when request:

HTTP GET /echo/hello



then response:

HTTP 200
{
    "echo": "hello"
}
```
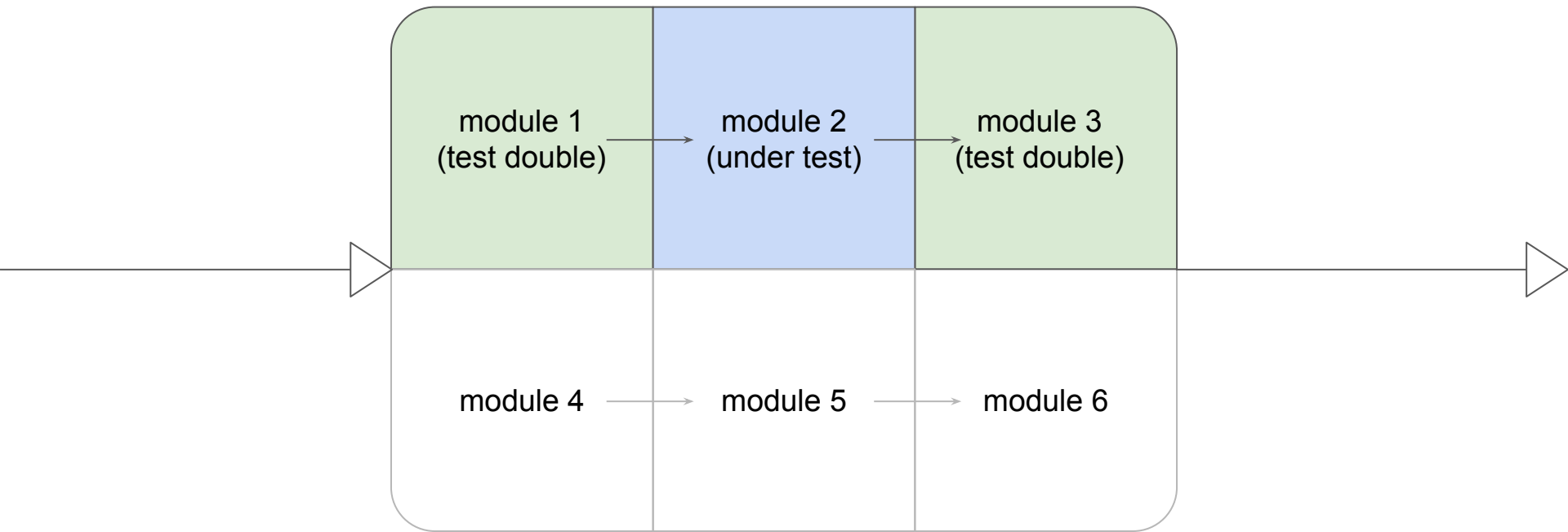
consumer

producer
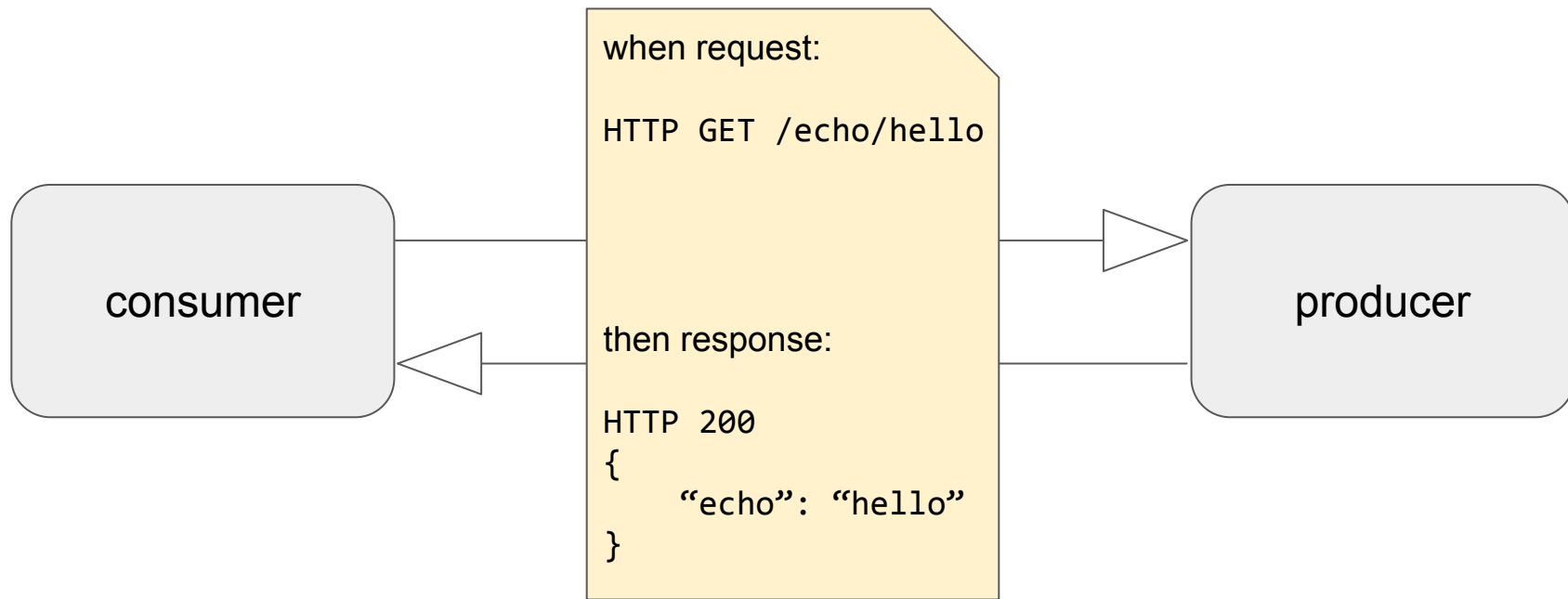
Contract Testing

Unit Tests

Integration Test

service 2
(test double)

service 1
(test double)

service 3
(under test)

service 5
(test double)

service 4
(test double)

Single integration

service 2

service 1

service 3
(under test)

service 5

service 4
(test double)
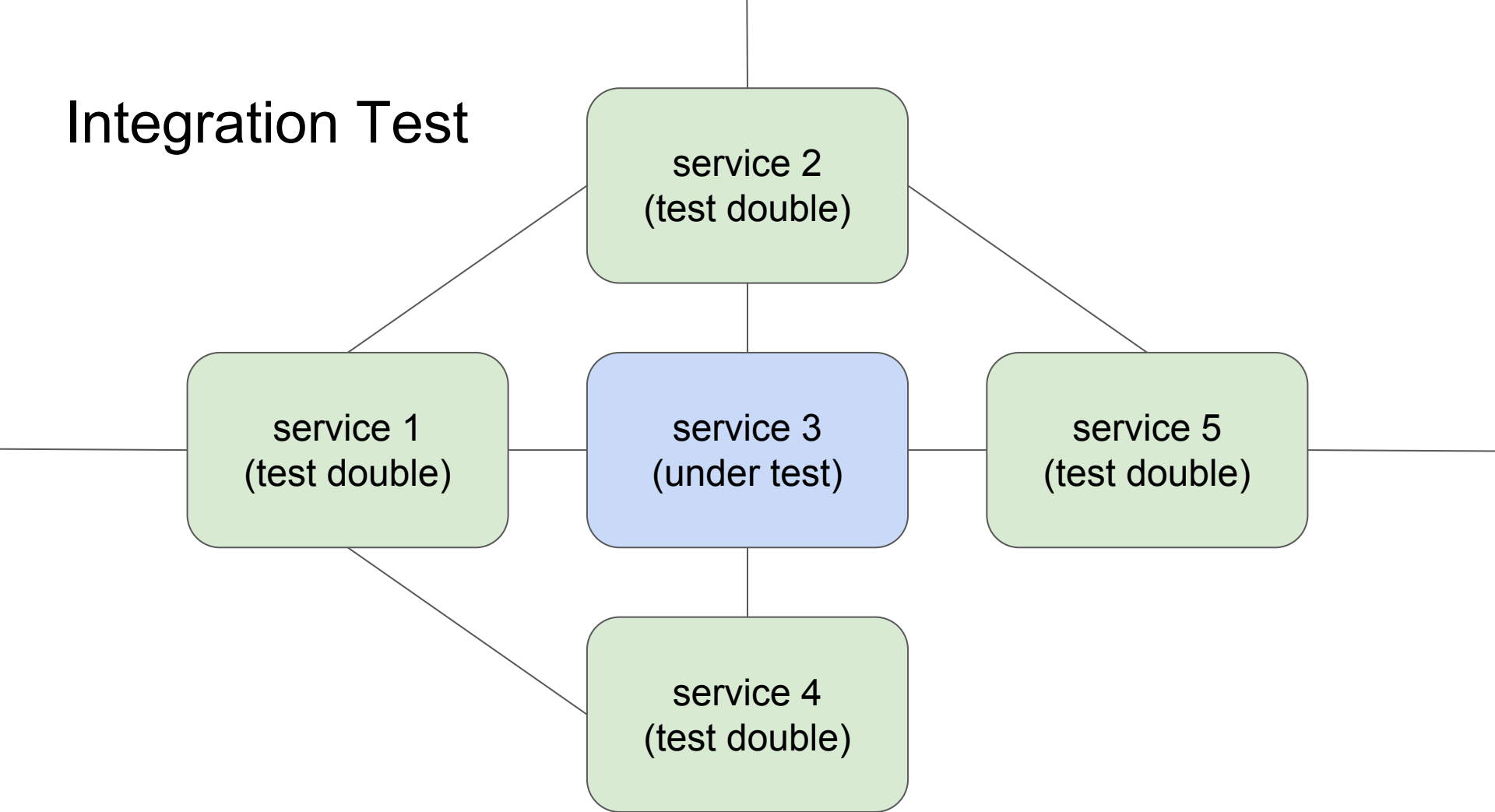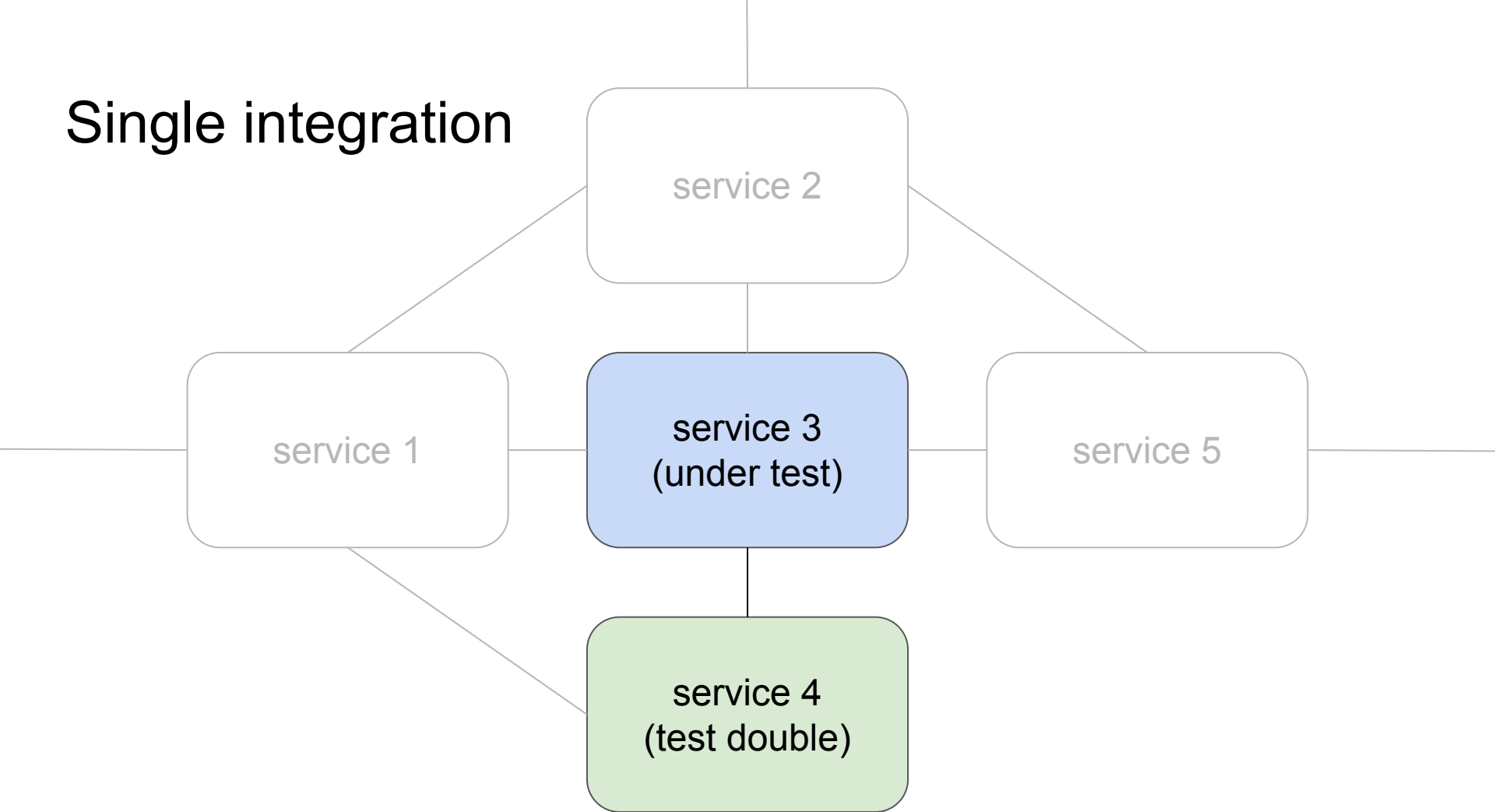
Single integration - sideways ;)

Contract test

service
(under test)

tested against ▷

when request:

HTTP GET /echo/hello

then response:

HTTP 200
{
    "echo": "hello"
}

service

Contract Test

```
when request:

HTTP GET /echo/hello




then response:

HTTP 200
{
    "echo": "hello"
}
```
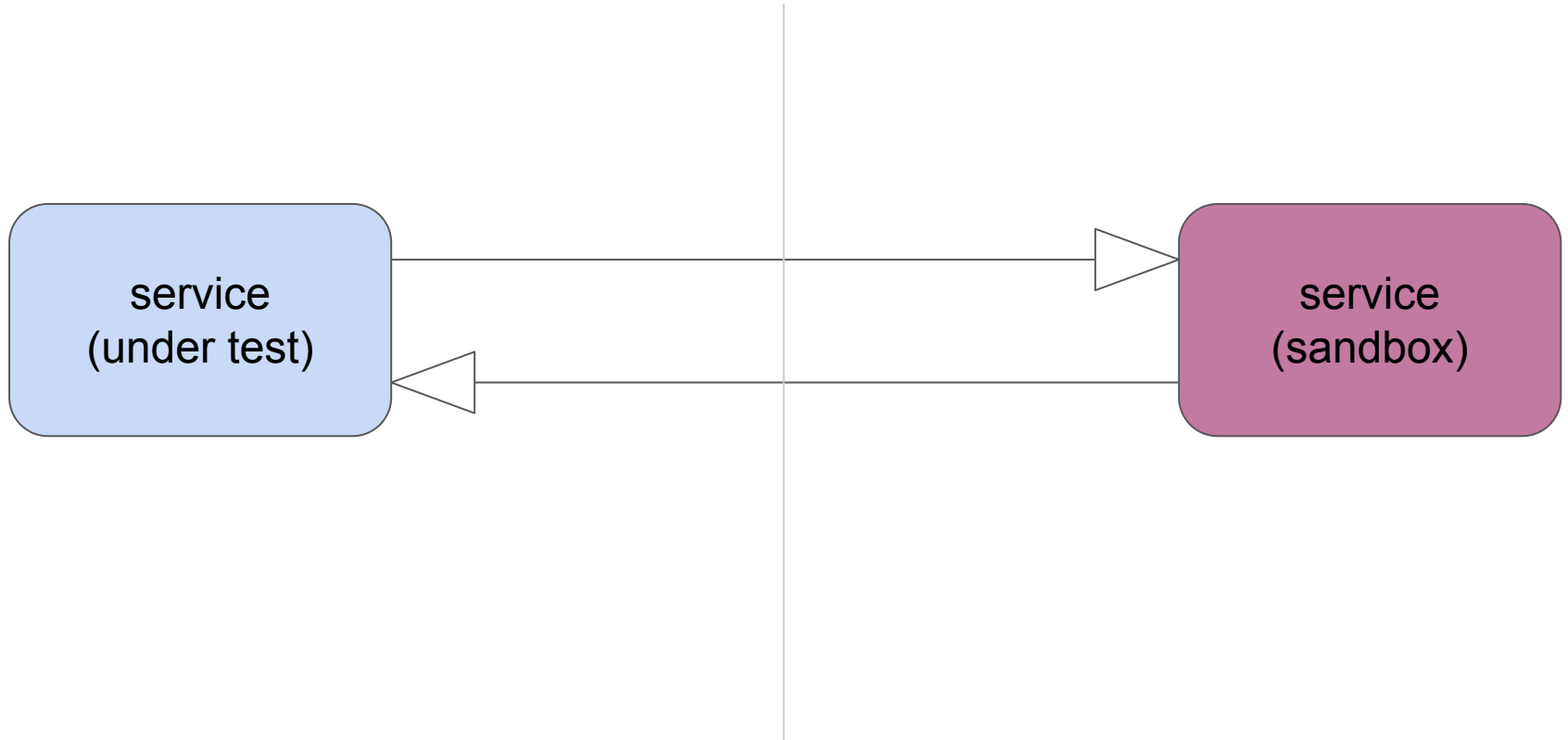
service

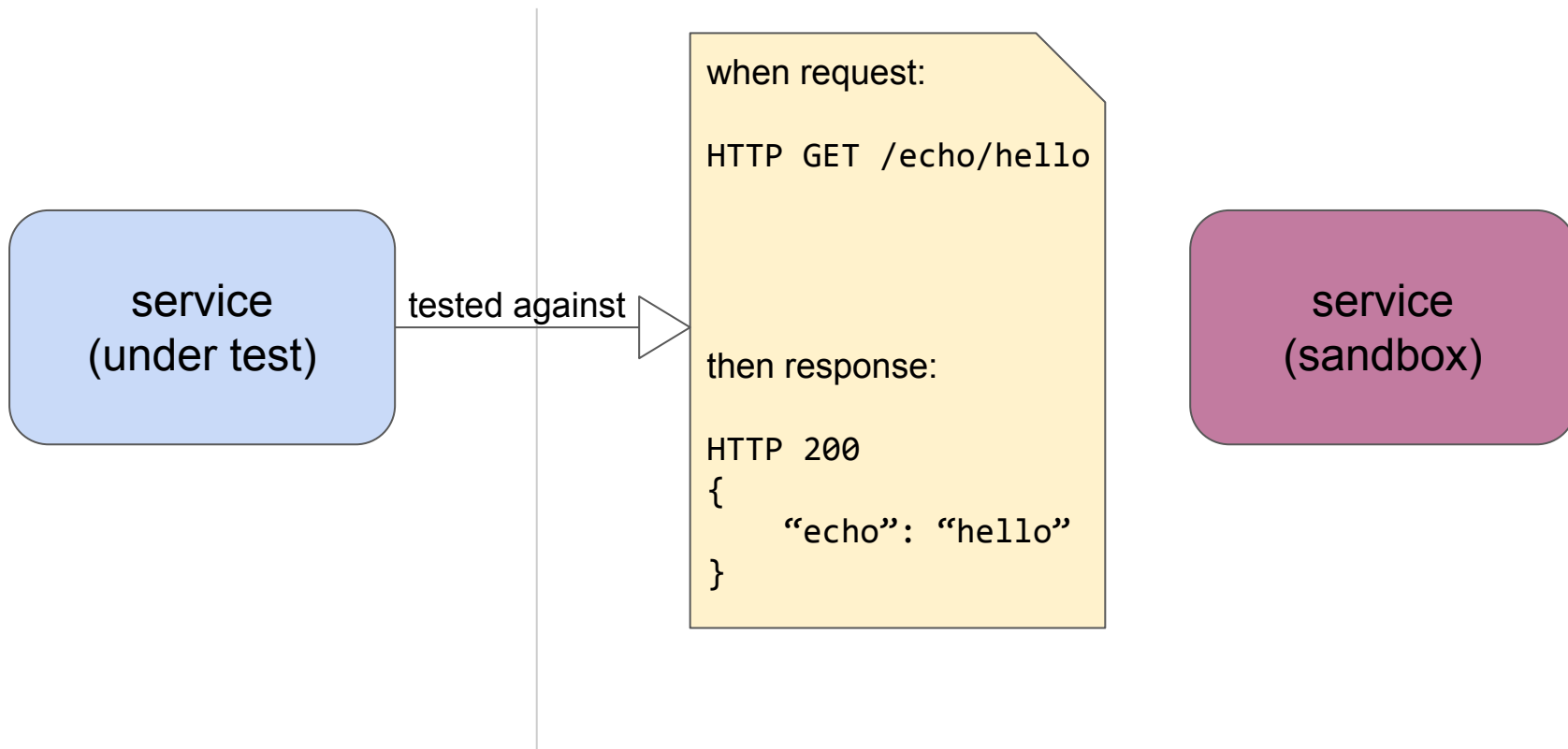tested against
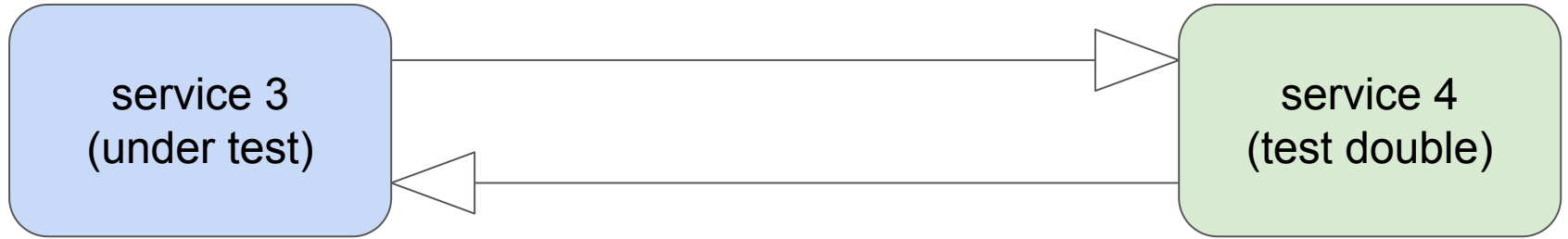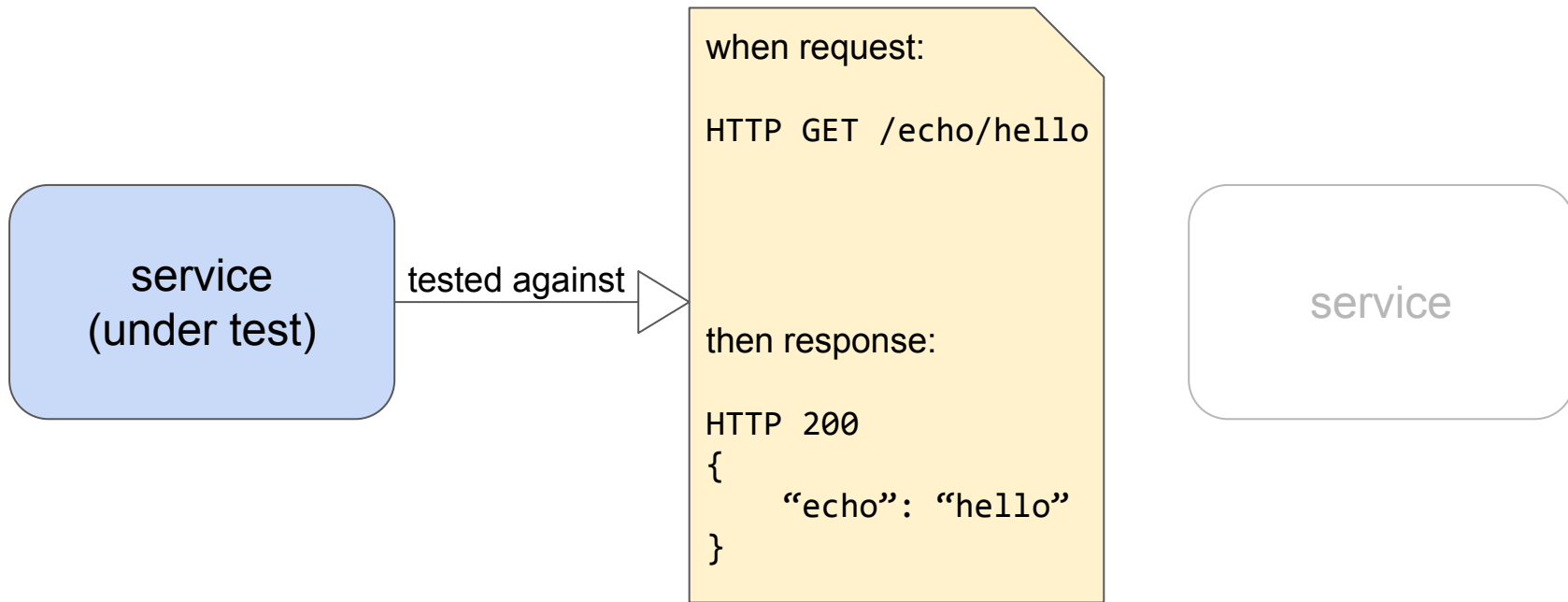
service
(under test)
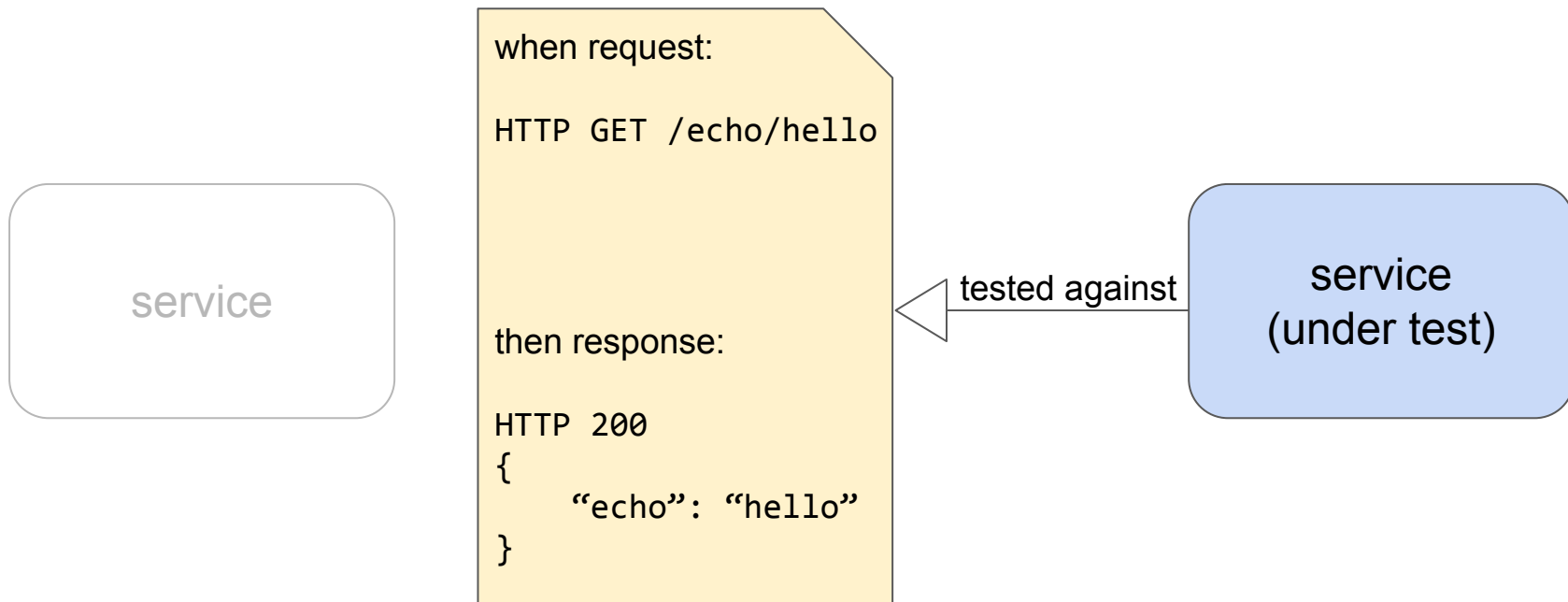
Single integration with third party - test double
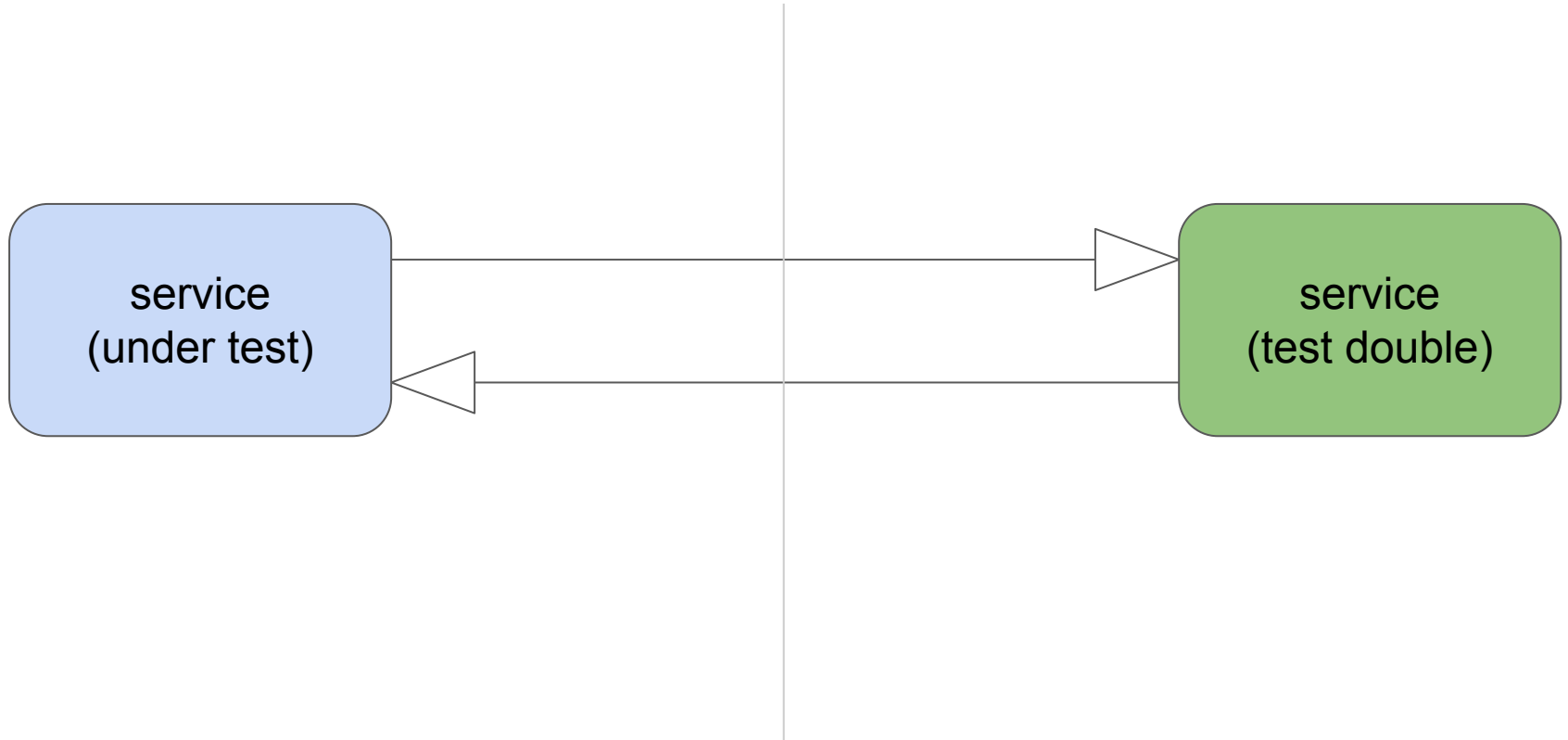
Single integration with third party - sandbox

Contract test - third party

service
(under test)

tested against

when request:

HTTP GET /echo/hello

then response:

HTTP 200
{
    "echo": "hello"
}

service
(sandbox)

Contract test - third party

service
(under test)

```
when request:

HTTP GET /echo/hello



then response:

HTTP 200
{
    "echo": "hello"
}
```

tested against →

service
(sandbox)

Why (not)?

Integration Test
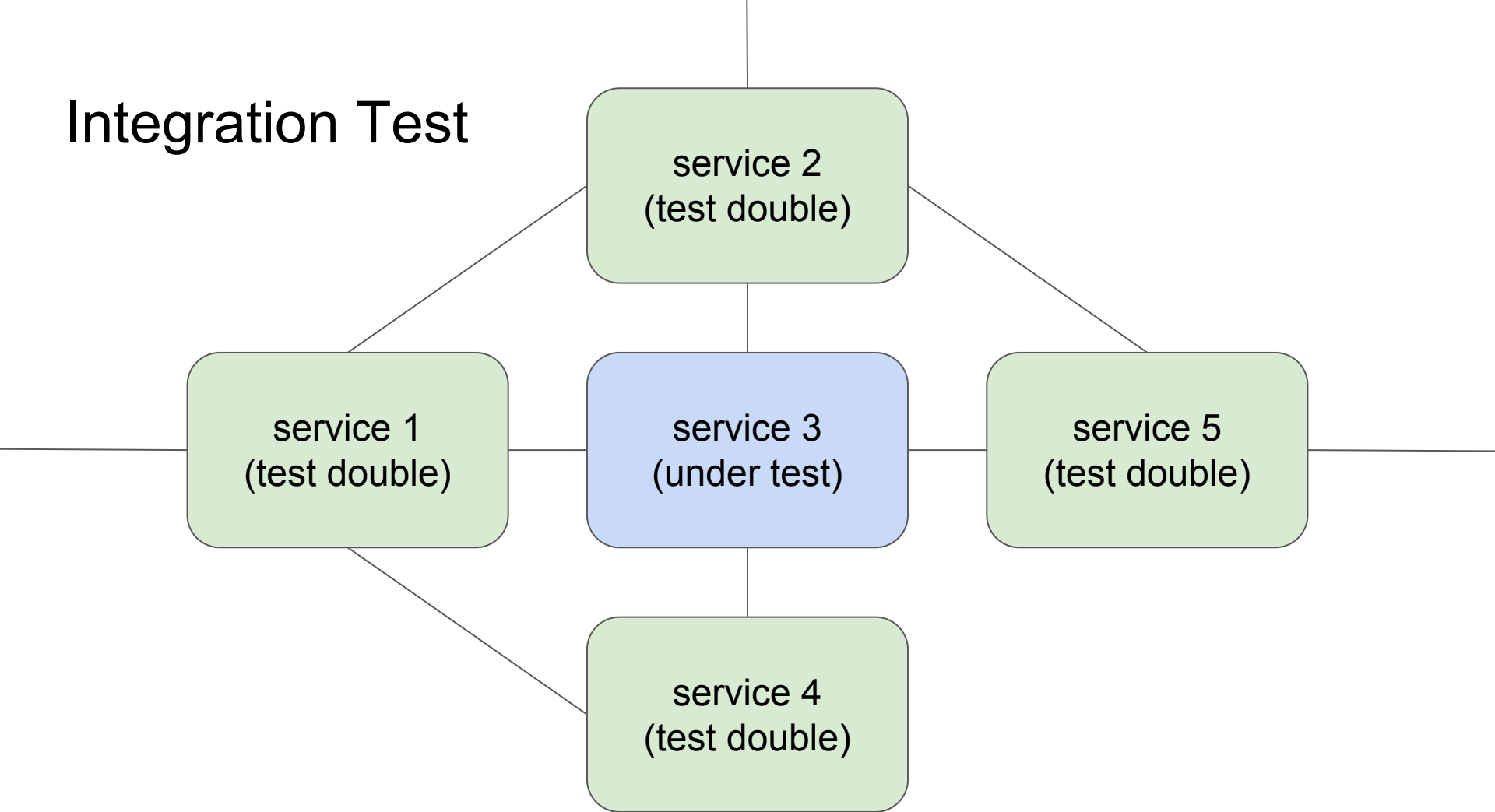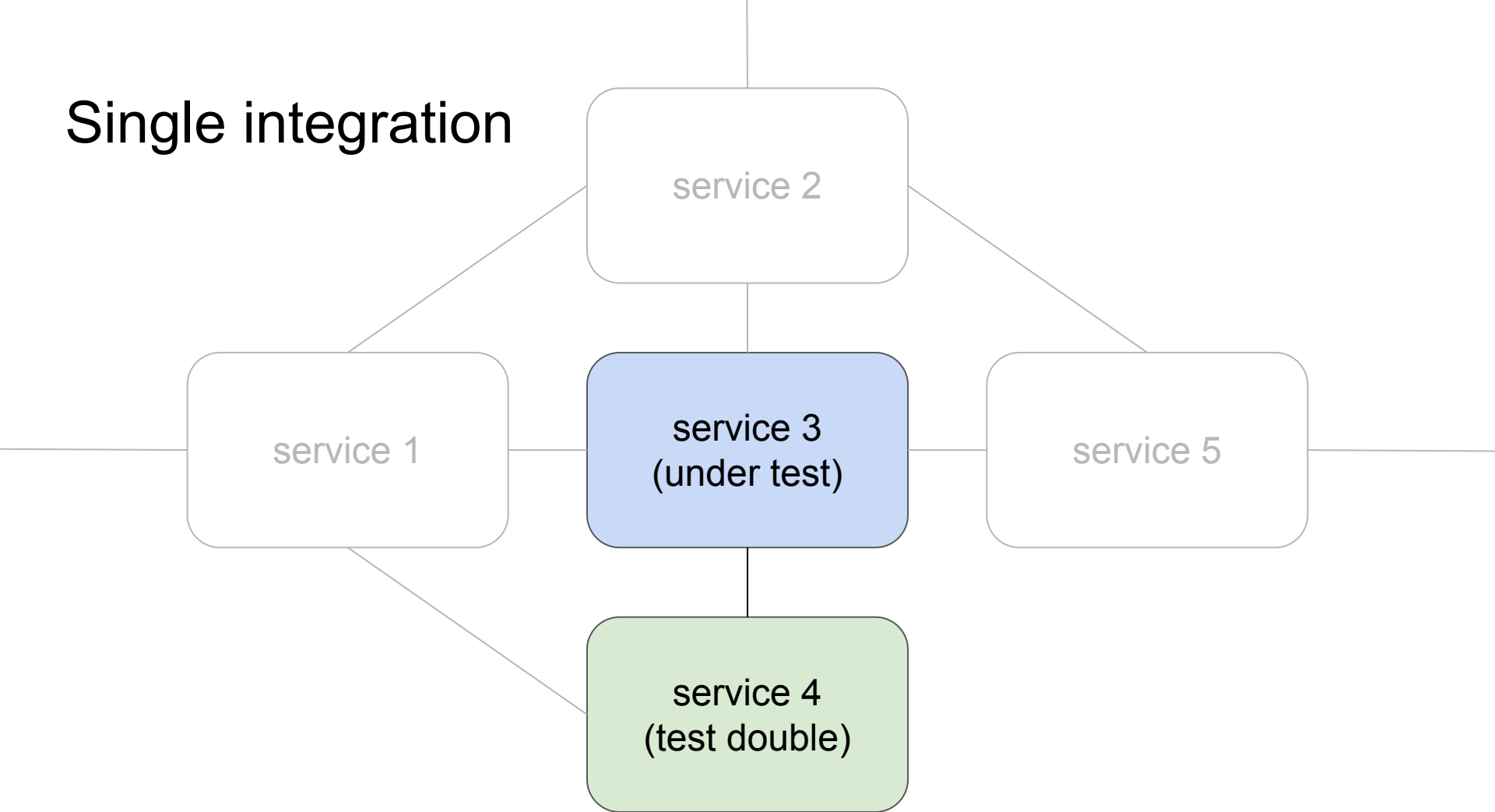
service 2
(test double)

service 1
(test double)

service 3
(under test)

service 5
(test double)

service 4
(test double)

Single integration

service 2

service 1

service 3
(under test)

service 5

service 4
(test double)

Single integration with third party - test double

service
(under test)

service
(test double)

Single integration with third party - sandbox

service
(under test)
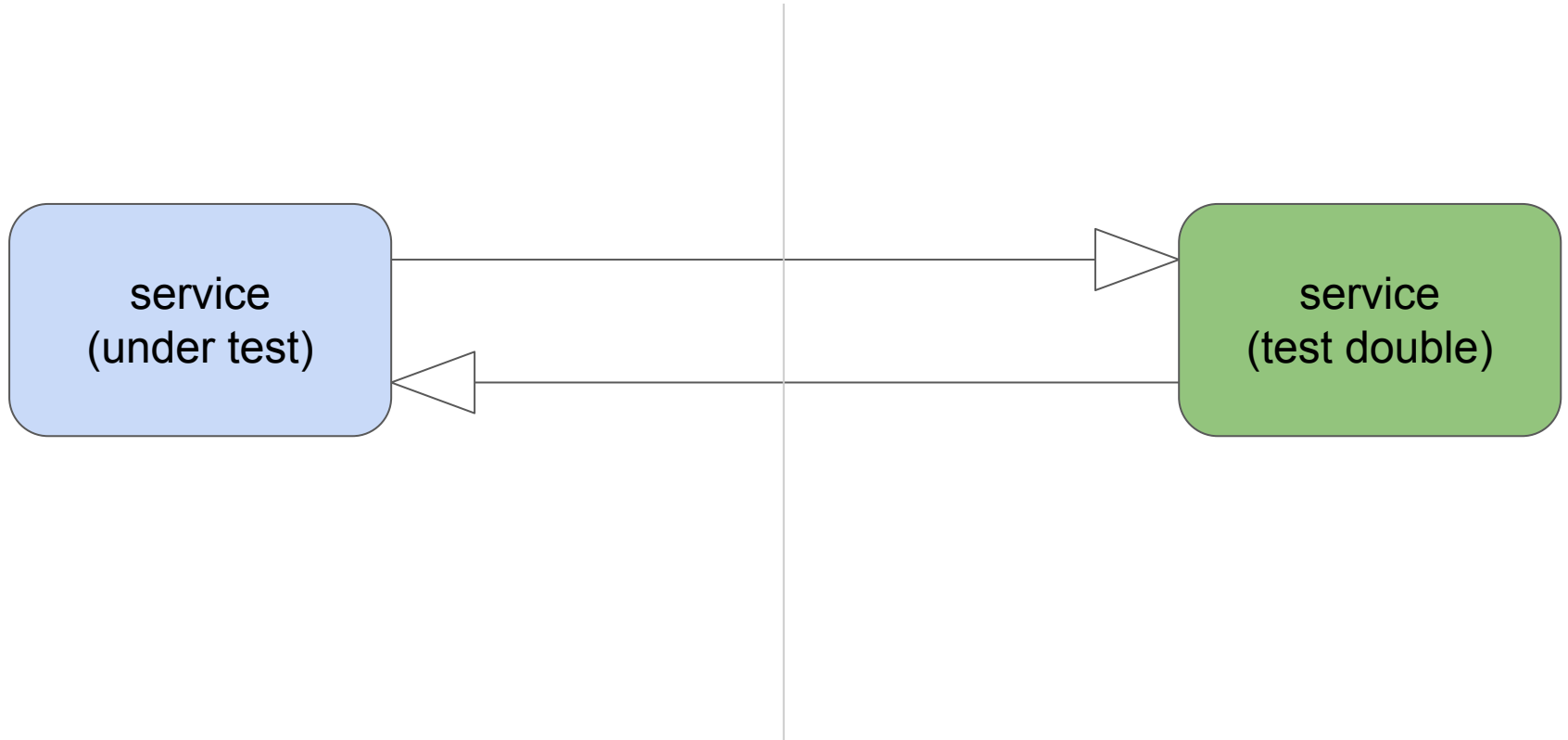
service
(sandbox)

Partial interface

Contract test - third party

service
(under test)

when request:

HTTP GET /echo/hello

then response:

HTTP 200
{
    "echo": "hello"
}

tested against

service
(sandbox)
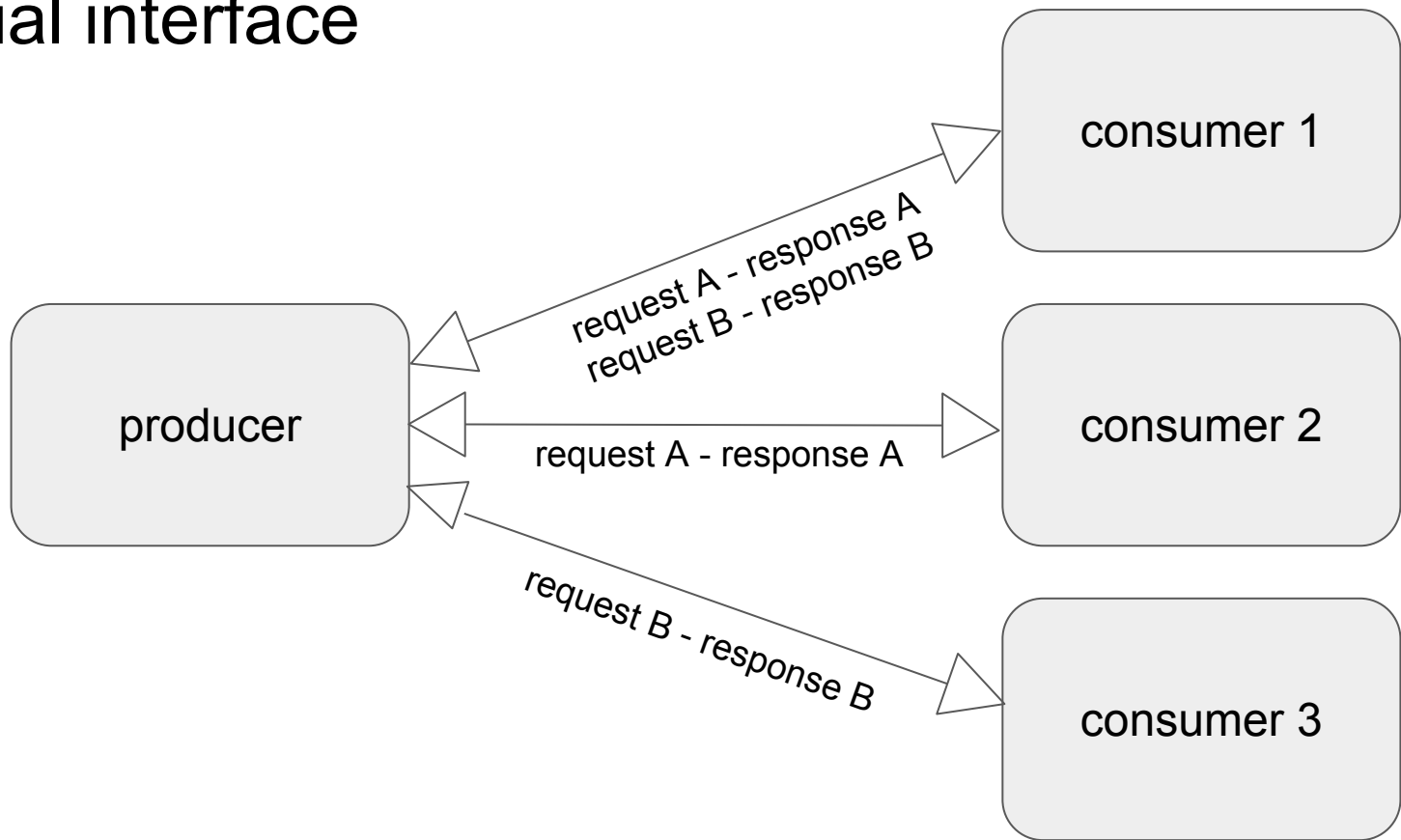
Tools

Spring Cloud Contract
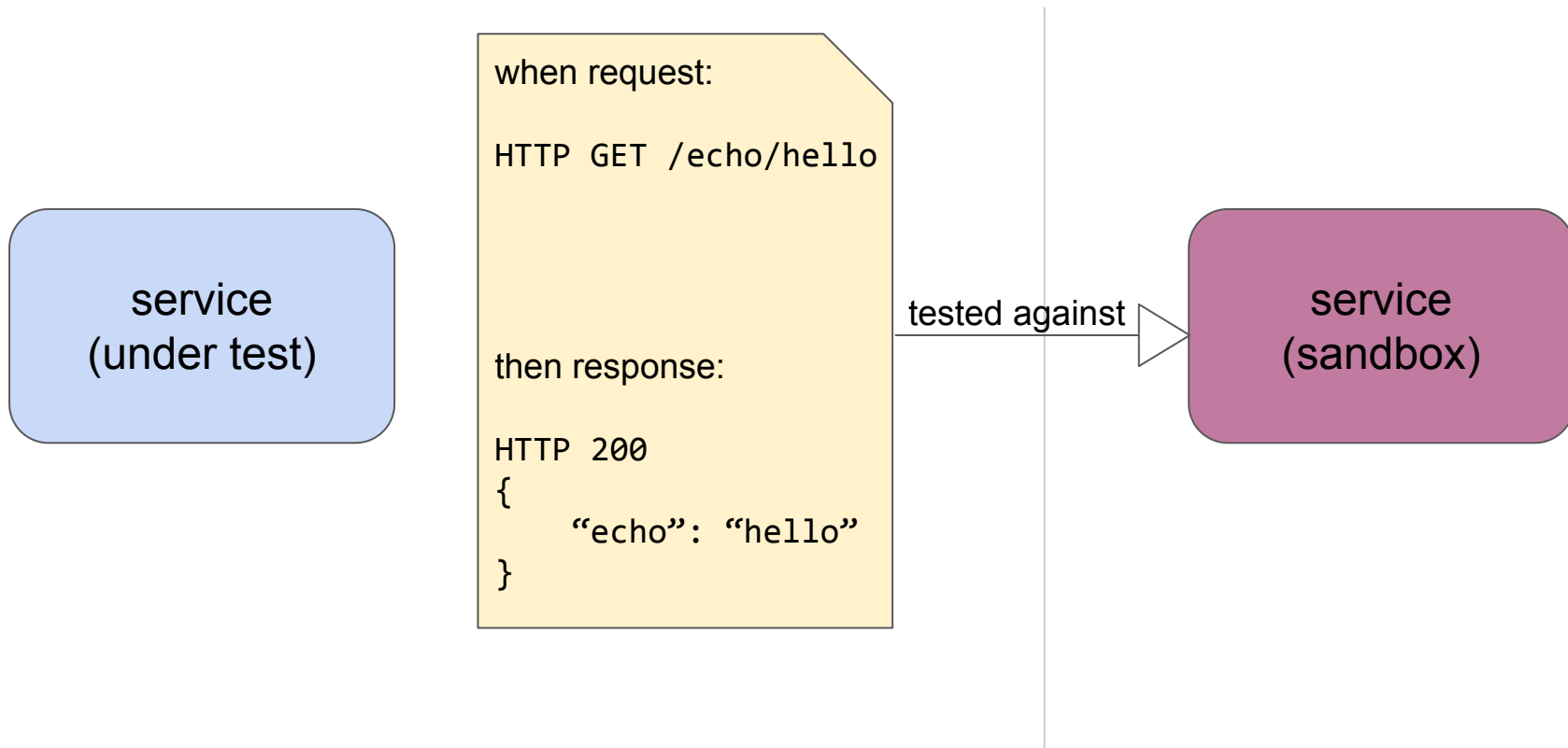
https://cloud.spring.io/spring-cloud-contract/

Pact

https://docs.pact.io/

Example

Java - shared contract

https://github.com/mbryla/contract-testing

Should we do it?

Q / A