

# Microservices. Test smarter, not harder.

Beth Skurrie (DiUS)  
@bethesque

A big problem

Problem:  
Long time to market

**Microservices!**

Solved a problem!  
New problem...



# Microservices

## Solved problems

- Feature time to market

## New problems

- E2E integration tests
  - Slow tests
  - Easy to break
  - Hard to fix
  - Scale badly
  - Lots of set up
  - Flakey tests ignored
  - Takes dev time away from features



To E2E test, or not to E2E test?



What if there was another way?

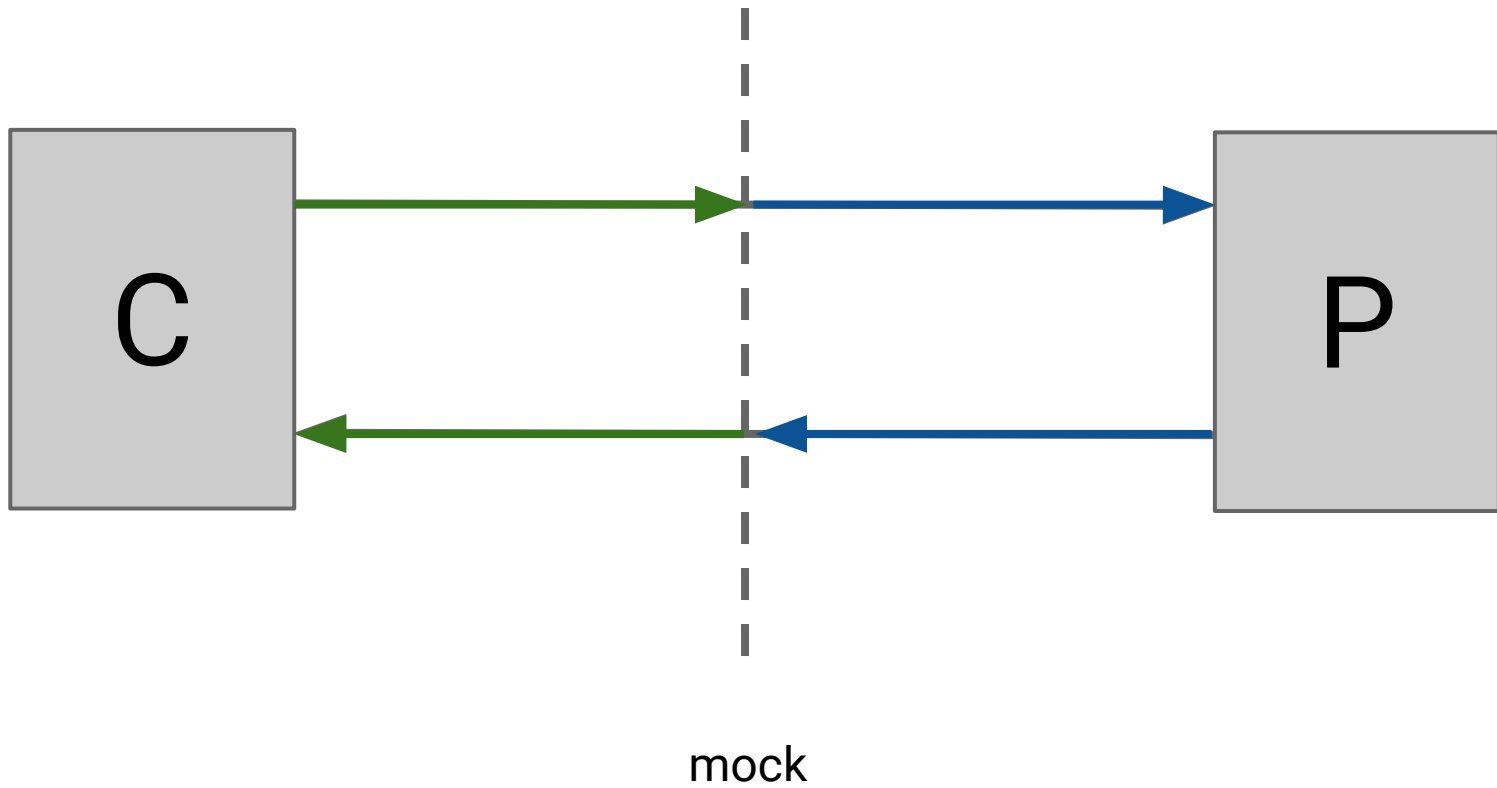




## Integration test

Consumer tests

Provider tests





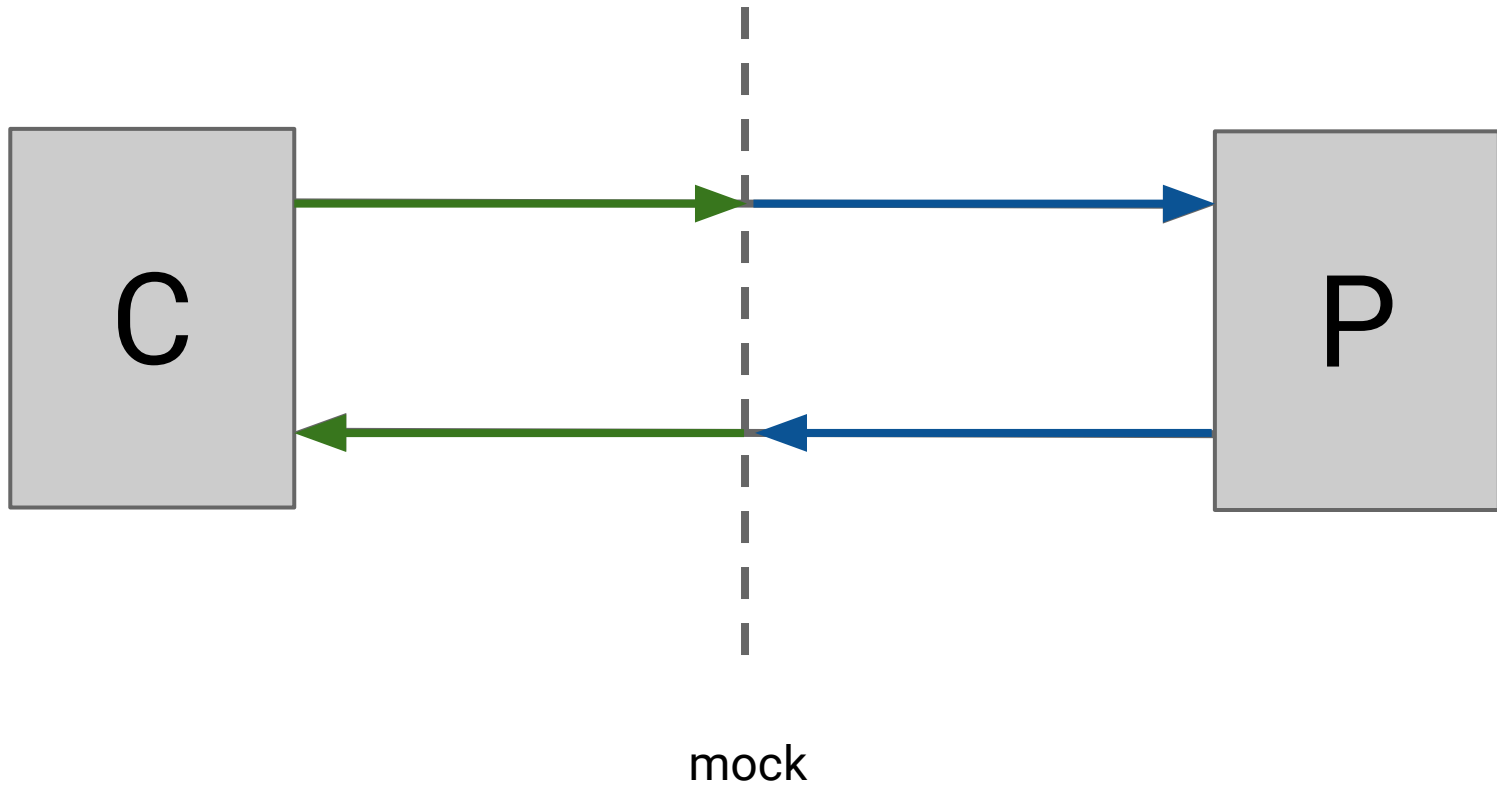
# Test symmetry

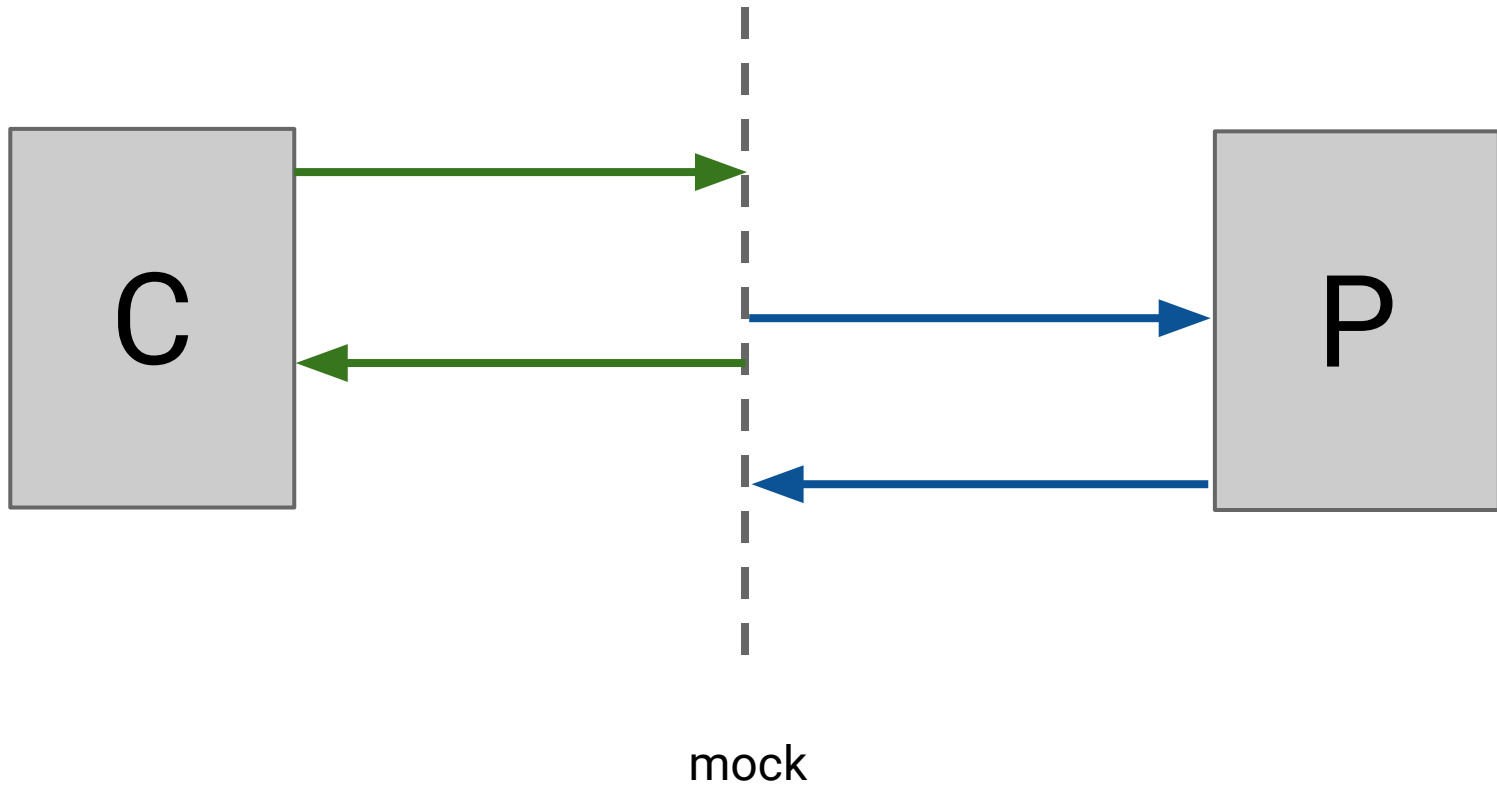
## Solved problems

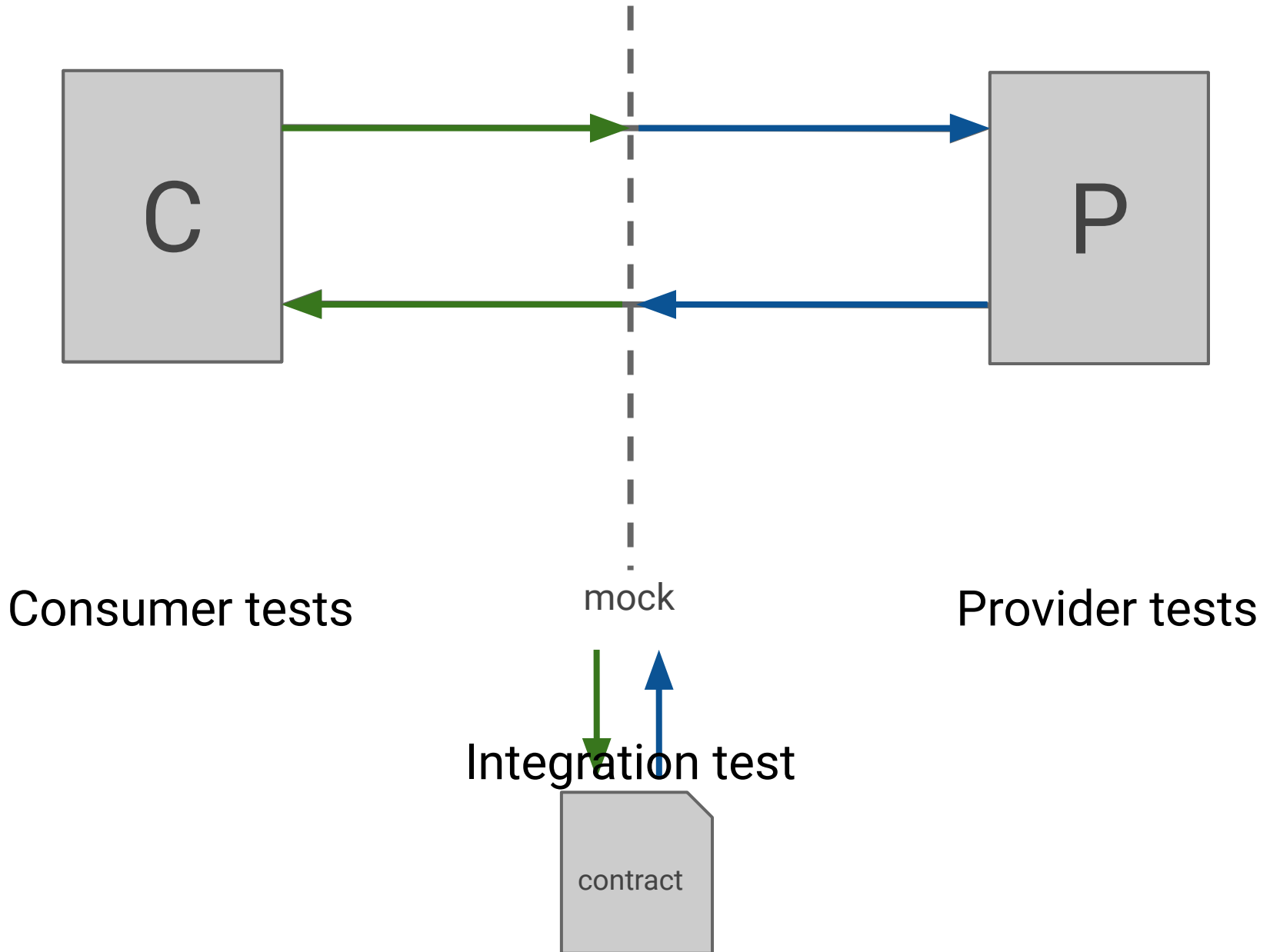
- Fast feedback
- Few dependencies
- No dedicated environment
- Reliable
- Easy to debug

## New problems

- Hard to keep both sides in sync

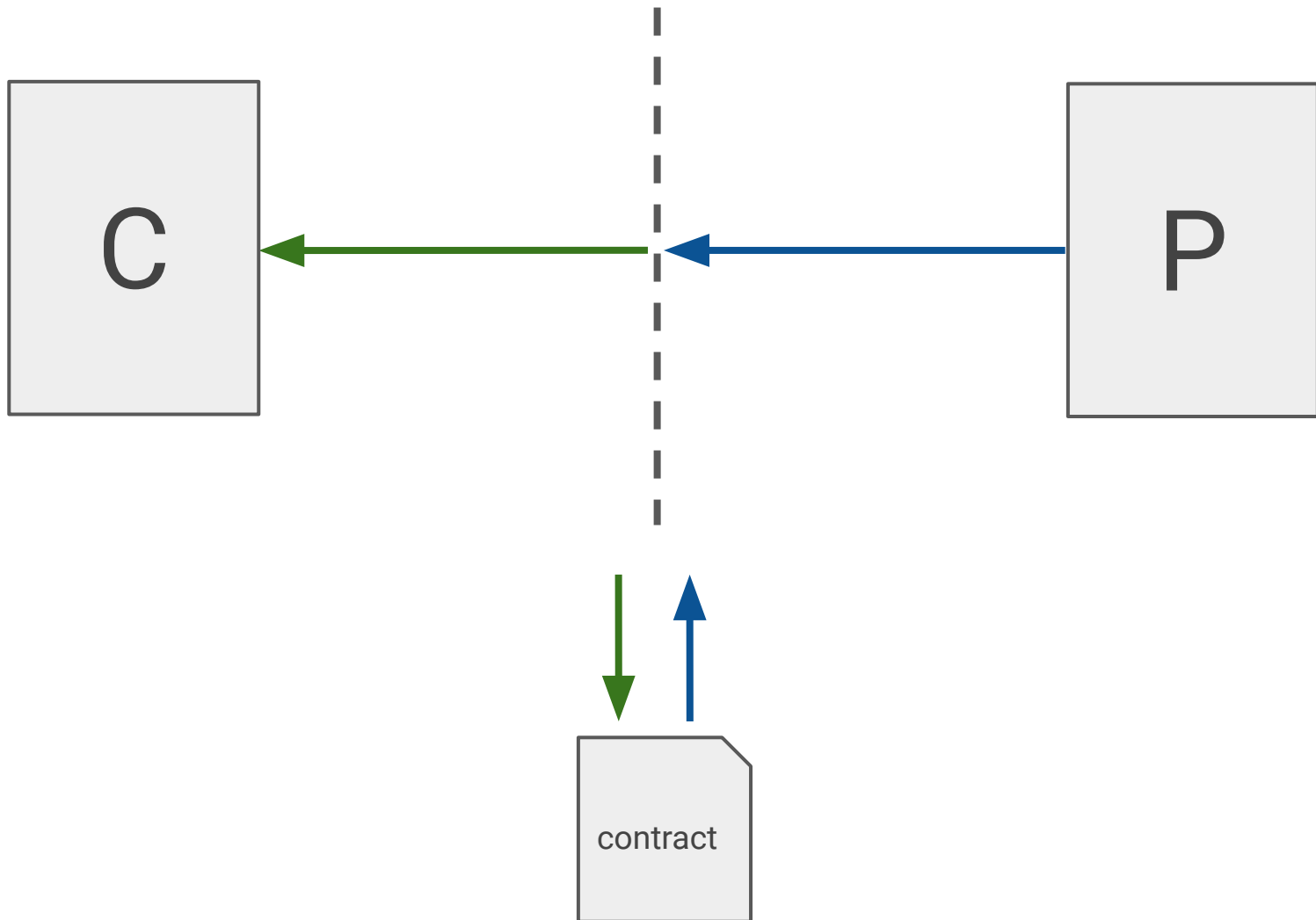








# Message Contract





# Contracts

## Solved problems

- Keeping tests in sync

## New problems

- ???

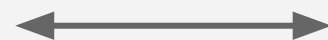
How long between writing the  
code and finding the bug?





Find  
bug  
here!

Not  
here!



5 mins

15 mins

45 mins

1 hour +



Local  
tests

Isolated CI  
tests

Deploy

E2E  
tests



# Contracts tests

Find integration bugs  
early and quickly



# Contracts tests

Know before you  
commit



# Contracts tests

Make changes with  
speed and confidence



Contracts tests

Deploy independently



(Consumer)  
Contracts tests

Better API design



# Contracts tests

Are **not** functional tests



# Contracts tests

Are **not** an API  
specification





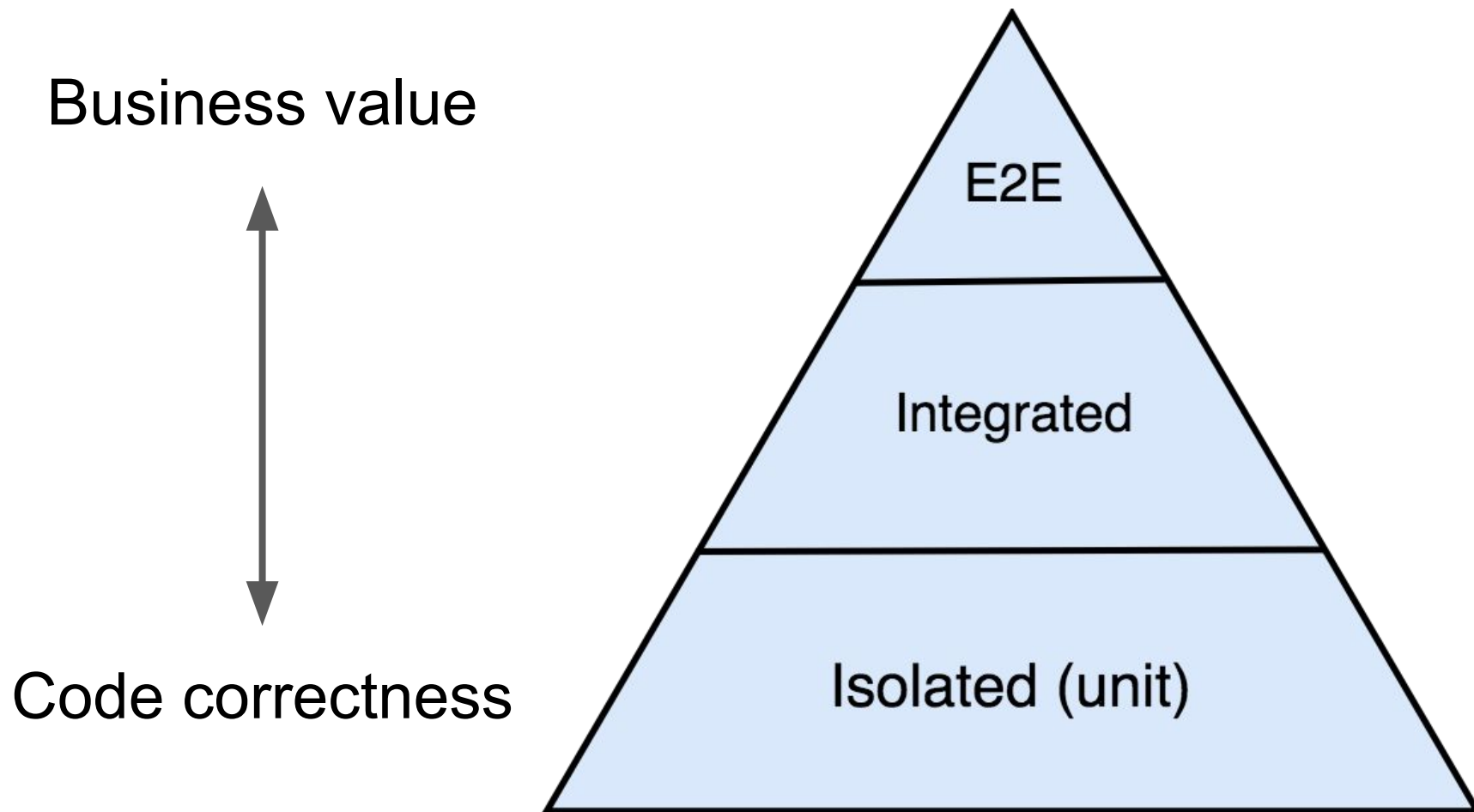
## (Consumer) Contracts tests

Are **not** good for “public”  
APIs (many, unknown  
consumers)



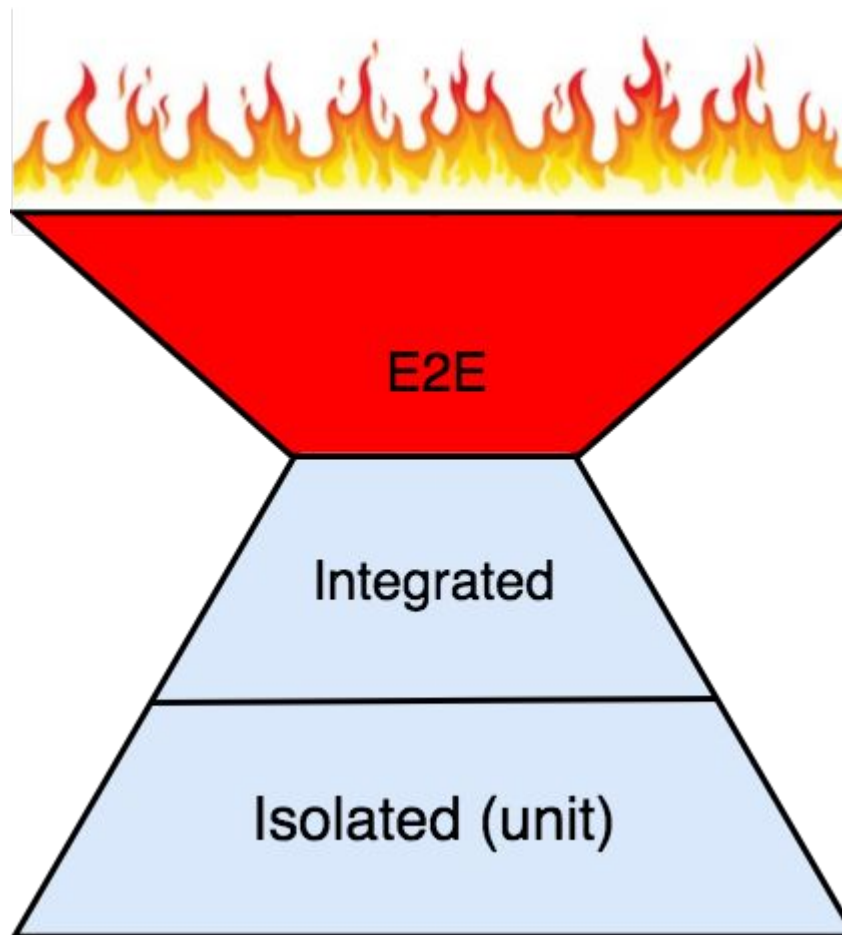
# Contracts tests

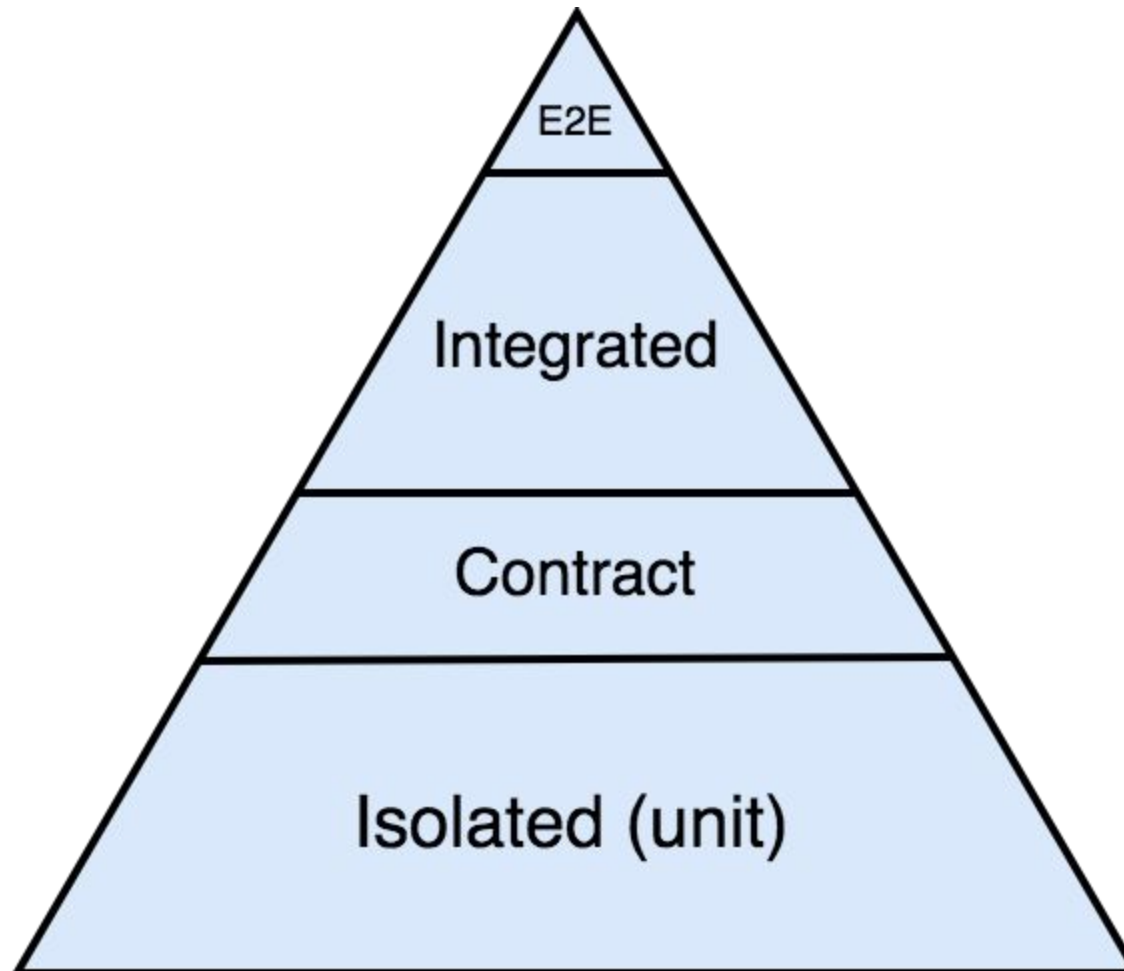
Are **not** a silver bullet!





# The Testing Pyramid of Hell







# Speed up your releases

## Do less

- Integration testing

## Do more

- Contract testing
- Aggregated logging
- Metrics
- Semantic monitoring
- Alerting
- Correlation IDs

# My contract testing journey

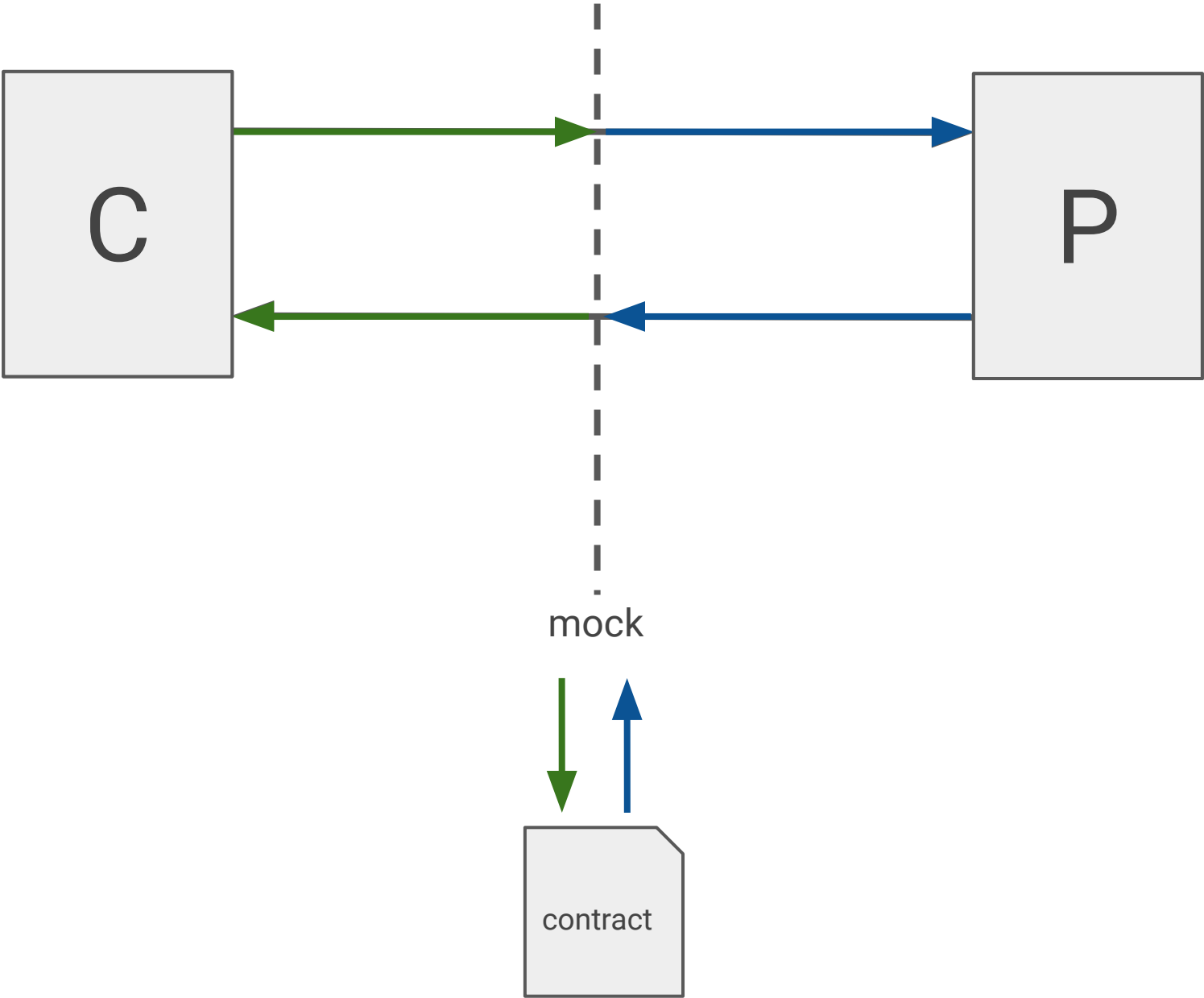


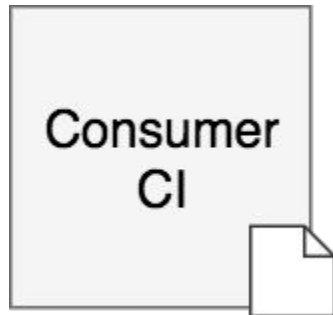
# PACT

pact.io

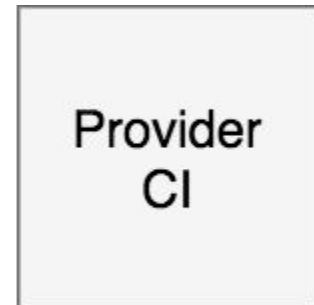
- Open source
- Multiple languages
  - JVM
  - .NET
  - Javascript
  - Python
  - Go
  - + more
- HTTP contracts
- Message contracts







?

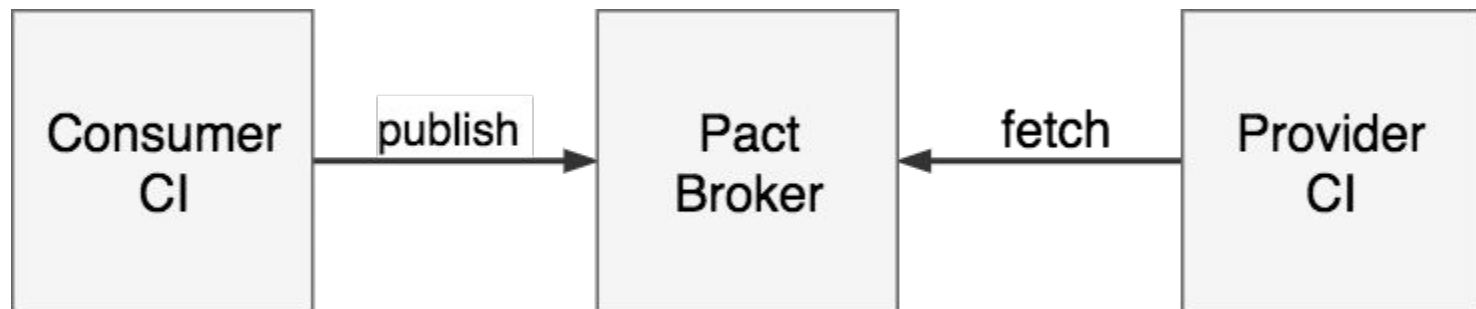




Too many microservices?  
You need another microservice!



# Pact Broker





# Problem 1: Contract exchange Solution: Pact Broker



# Problem 2: brittle tests



# Problem 2: brittle tests

## Solution: flexible matching



# Problem 3: Dealing with contract changes



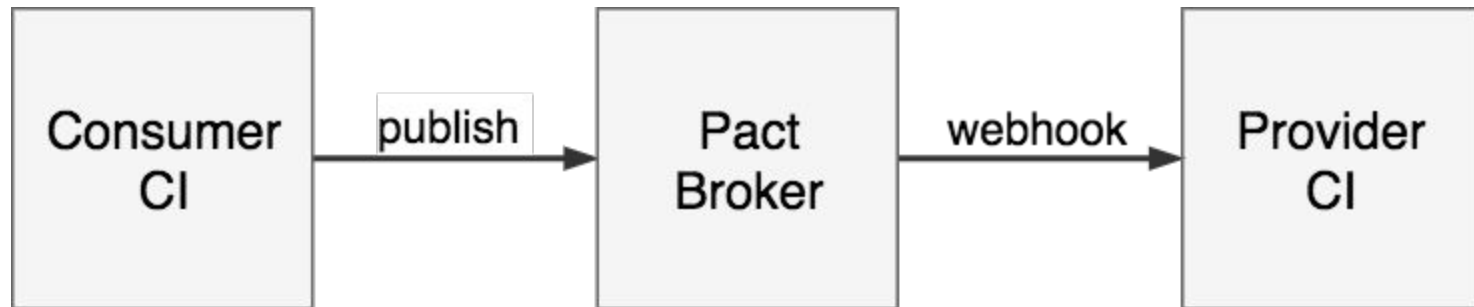


The other service needs to know when a contract has changed

Contracts are not a substitute for good communication between teams



# Pact Broker webhooks



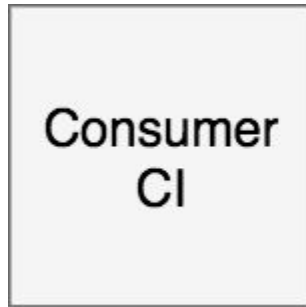


Contracts are **STILL** not a substitute for  
good communication between teams

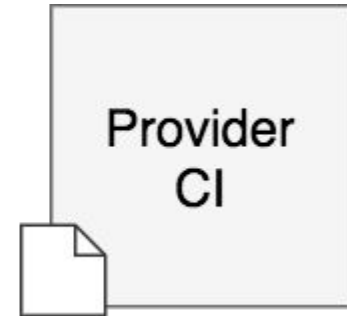


# A contract testing journey

- Automate the contract exchange
- Contracts should be as flexible as possible - but no more
- The provider needs to know when a contract has changed
- Remember: contracts are not a substitute for good communication between teams



?

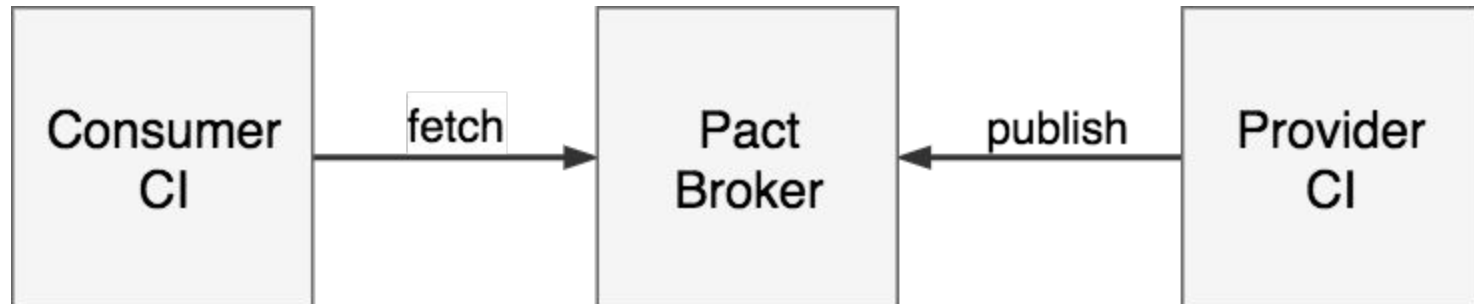


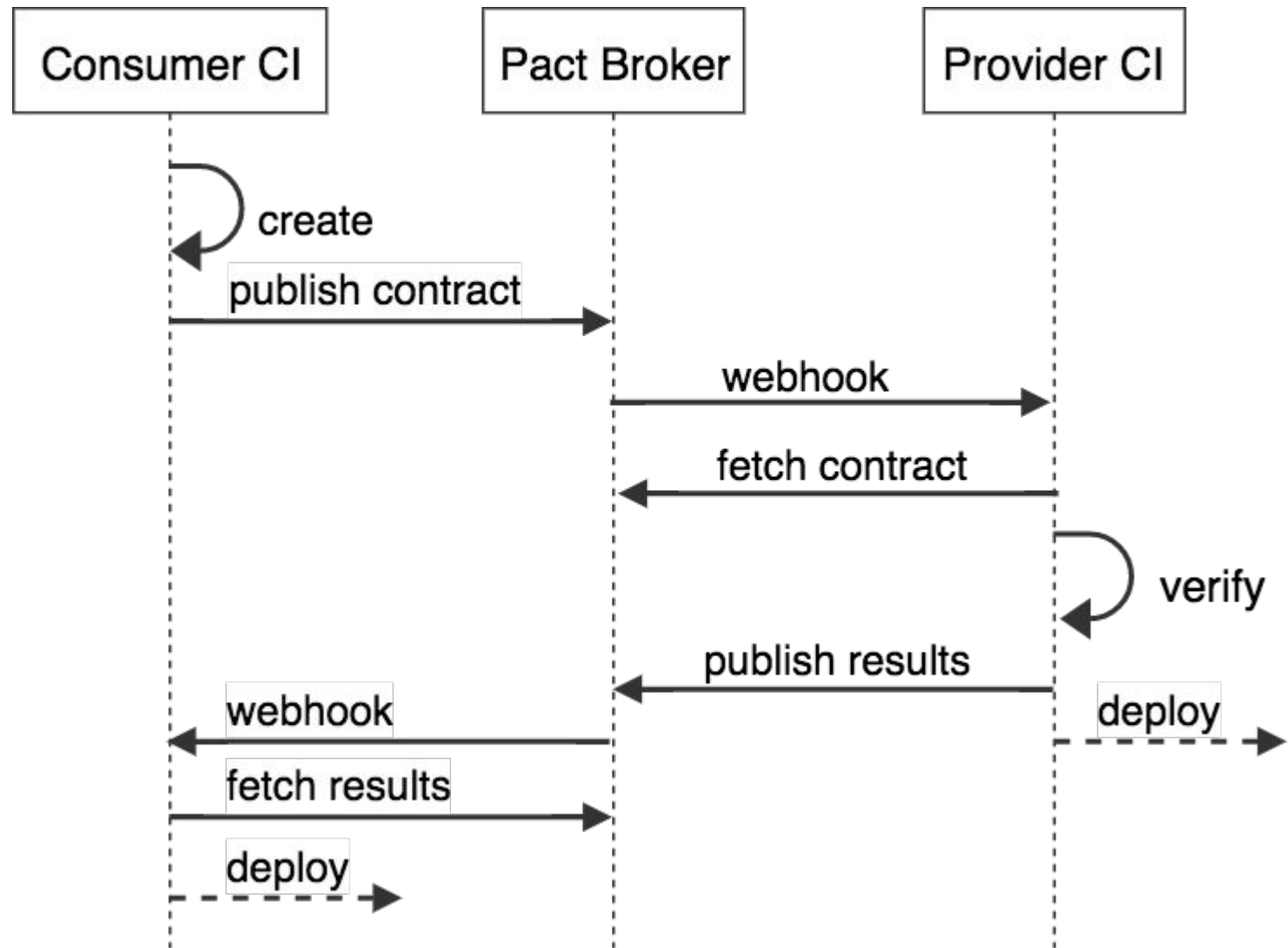


# Problem 4: Communicating verification results back to consumer

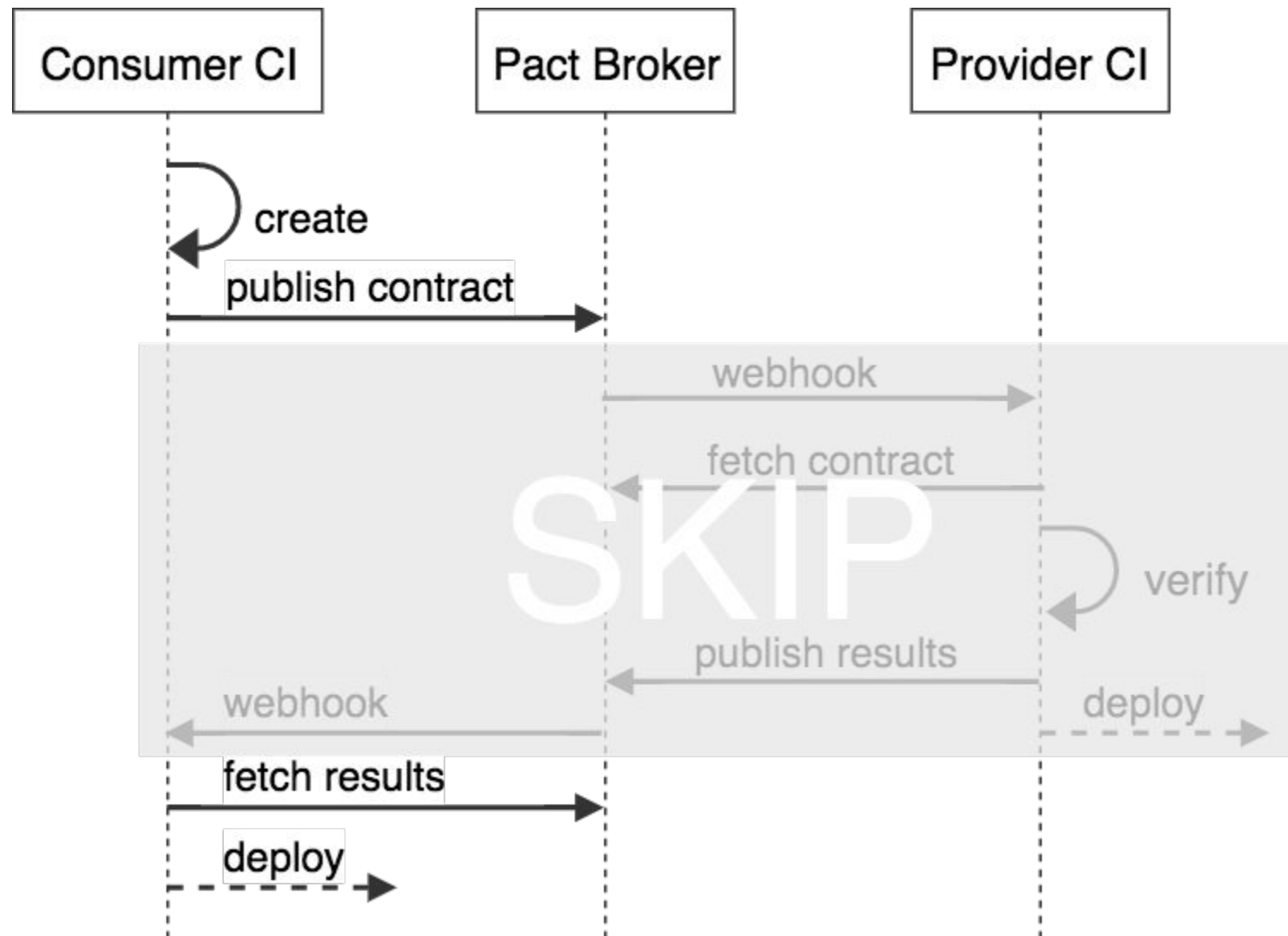


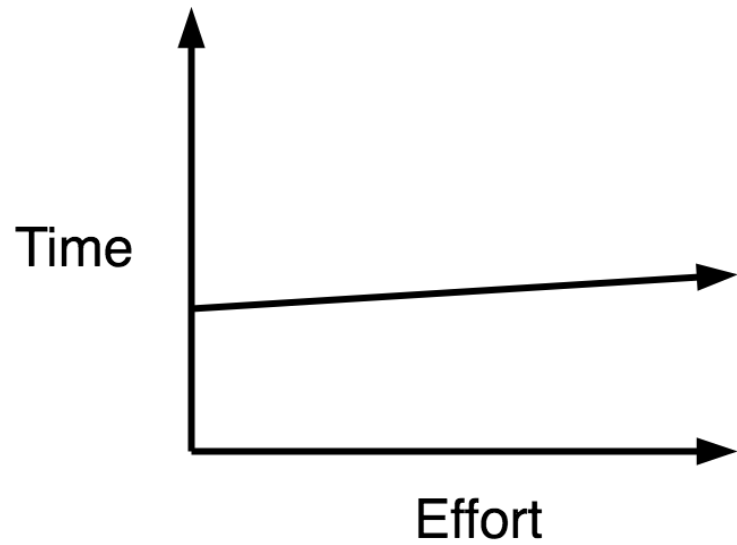
# Pact Broker verifications



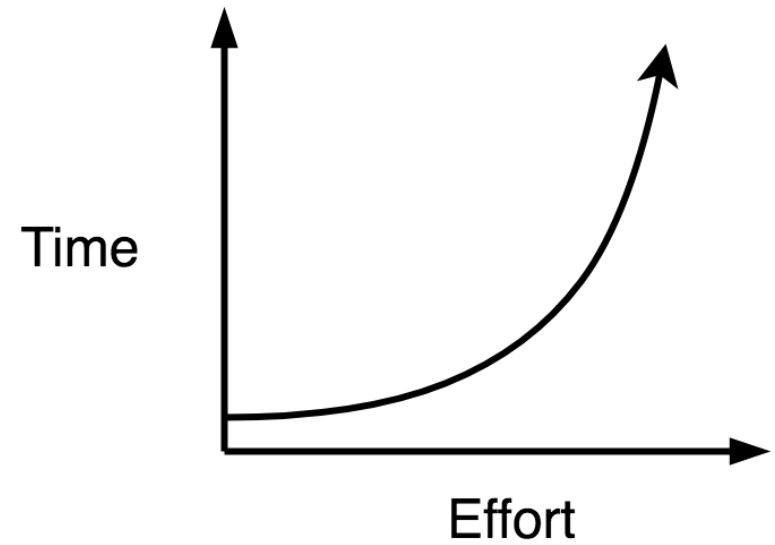








**Contract  
tests**



**E2E  
tests**



If you can't deploy your services  
independently,  
you don't have microservices.  
You have a distributed monolith.



**Warning!**

Do not build a  
distributed monolith



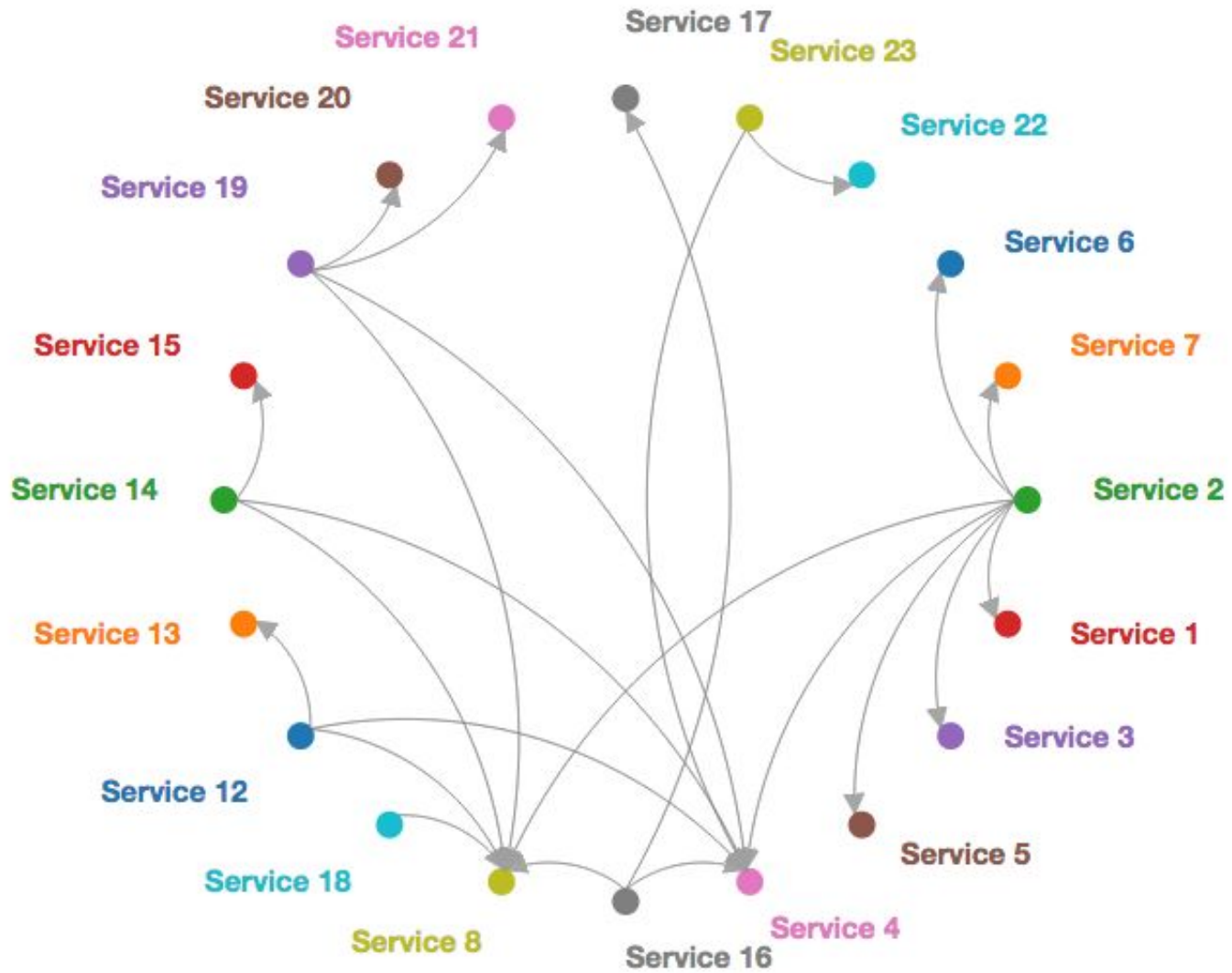
# “The Matrix”

<b>Consumer version</b>	<b>Provider version</b>	<b>Verification result</b>
11	54	success
12	54	failure
12	55	success
13	56	success
13	57	failure



# Can I deploy?

Consumer version	Provider version	Verification result
11 ✓	54 <b>PROD</b>	success
12	54	failure
12	55	success
13	56	success
13	57	unknown





Given **there is an alligator named Mary**, upon receiving a **request for an alligator** from Zoo App, with

```
{
  "method": "get",
  "path": "/alligators/Mary",
  "headers": {
    "Accept": "application/json"
  }
}
```

Animal Service will respond with:

```
{
  "status": 200,
  "headers": {
    "Content-Type": "application/json;charset=utf-8"
  },
  "body": {
    "name": "Mary"
  }
}
```

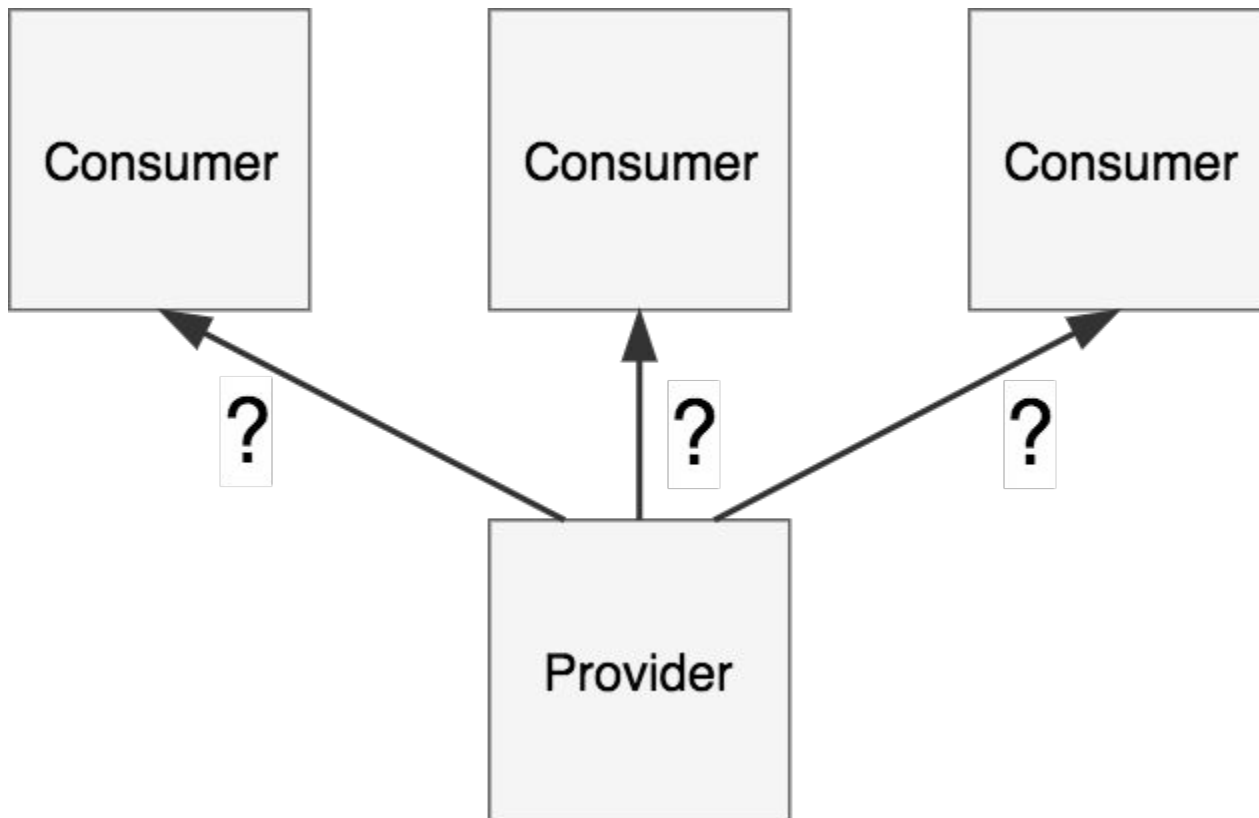




# What about Swagger/OAS?

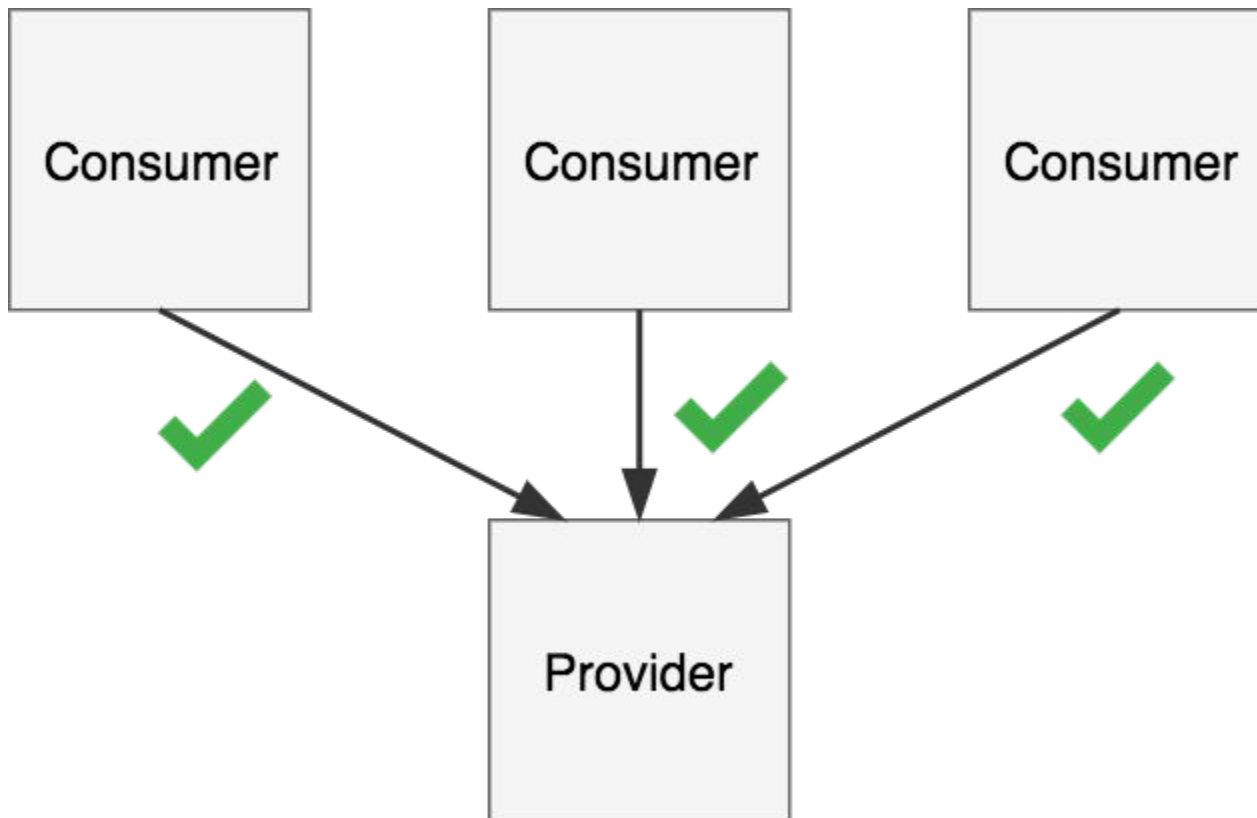


# Provider contracts



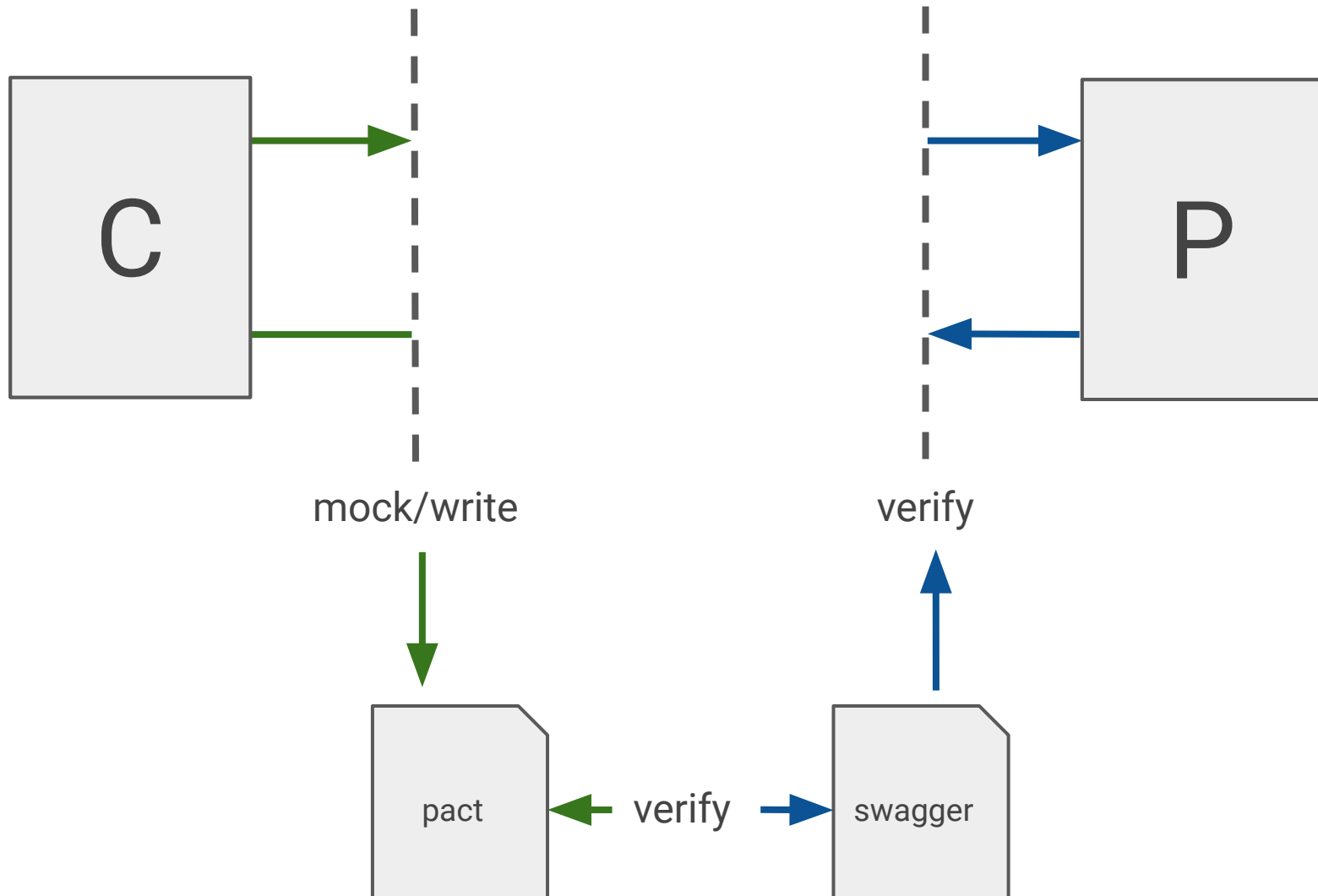


# Consumer contracts





# Pact+Swagger





What are the next  
problems to  
solve for  
Pact?

- OAS support
- Improved Broker workflow



# Contracts

## Solved problems

- Shipping code faster

## New problems

- ???

Microservices.  
Test smarter, not harder

pact.io  
pactflow.io  
slack.pact.io  
@pact\_up

Beth Skurrie (DiUS)  
@bethesque

TRISH KHOO | TRANSPARO | @HOGFISH

# TAMING THE BEAST

Automated testing for complex data pipelines





**Previously:**

Technical Lead / Manager  
Google (Geo)  
California, USA

**Now:**

Consultant / Trainer  
Transparo  
Brisbane, Australia



~500 BILLION

Events per day



~8 MILLION

Events per second during  
peak hours



SEVERAL HUNDRED

Event streams

# Netflix