

Contract Testing with Pact



Topics

Problem ?

Contract testing

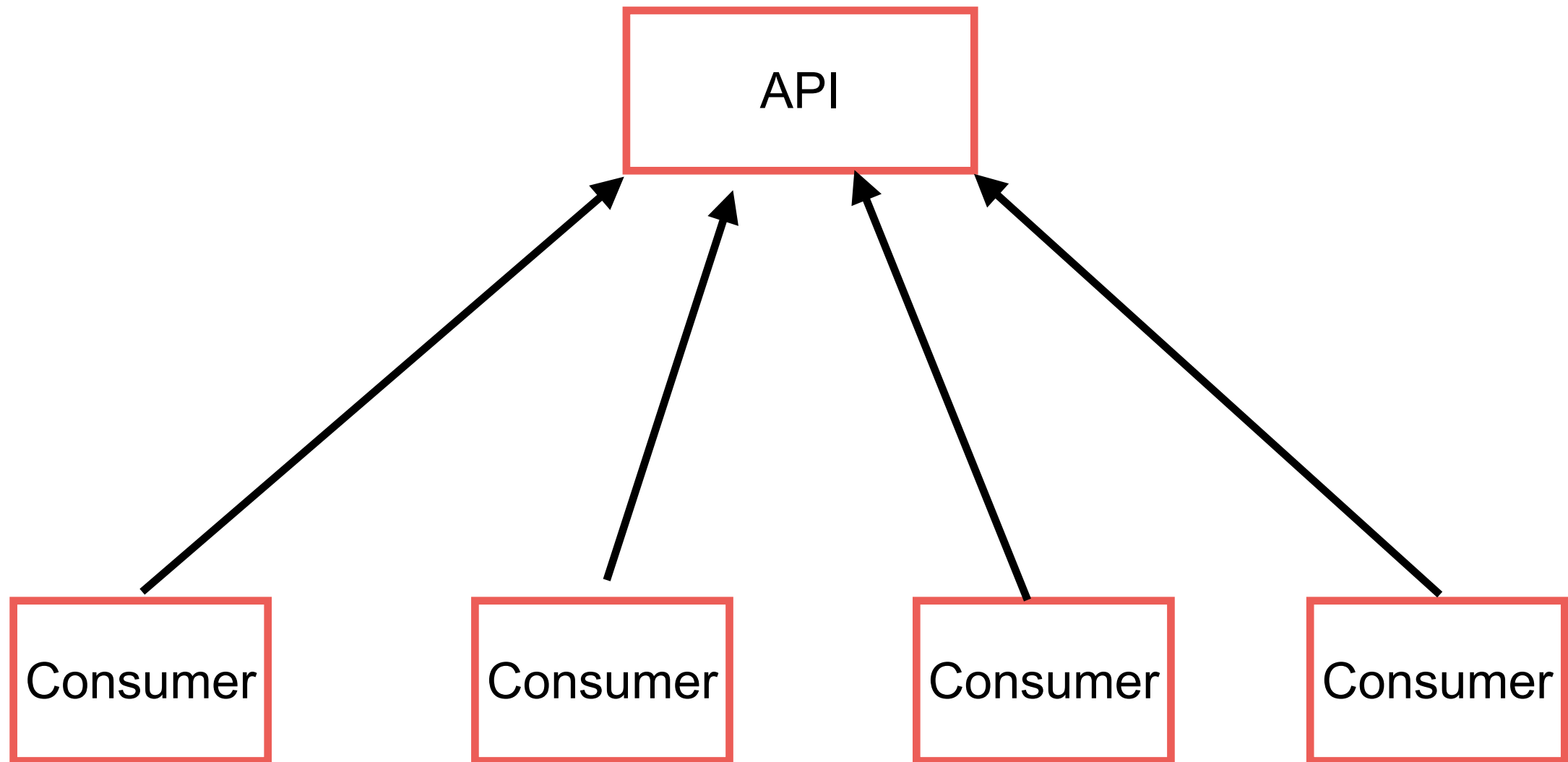
What is Pact ?

Workshop with NodeJS and Go



Problem ?

Change !!

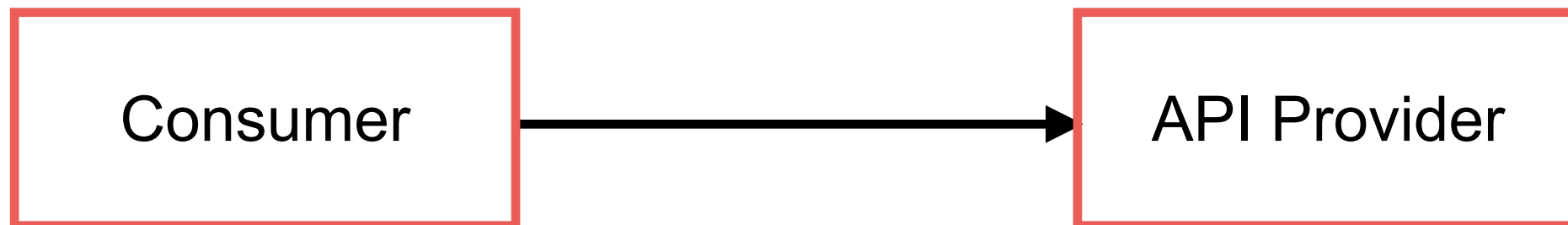


Ensure API compatibility



Integration test

High confidence to release



Problems of integration test

Control Dependency !!

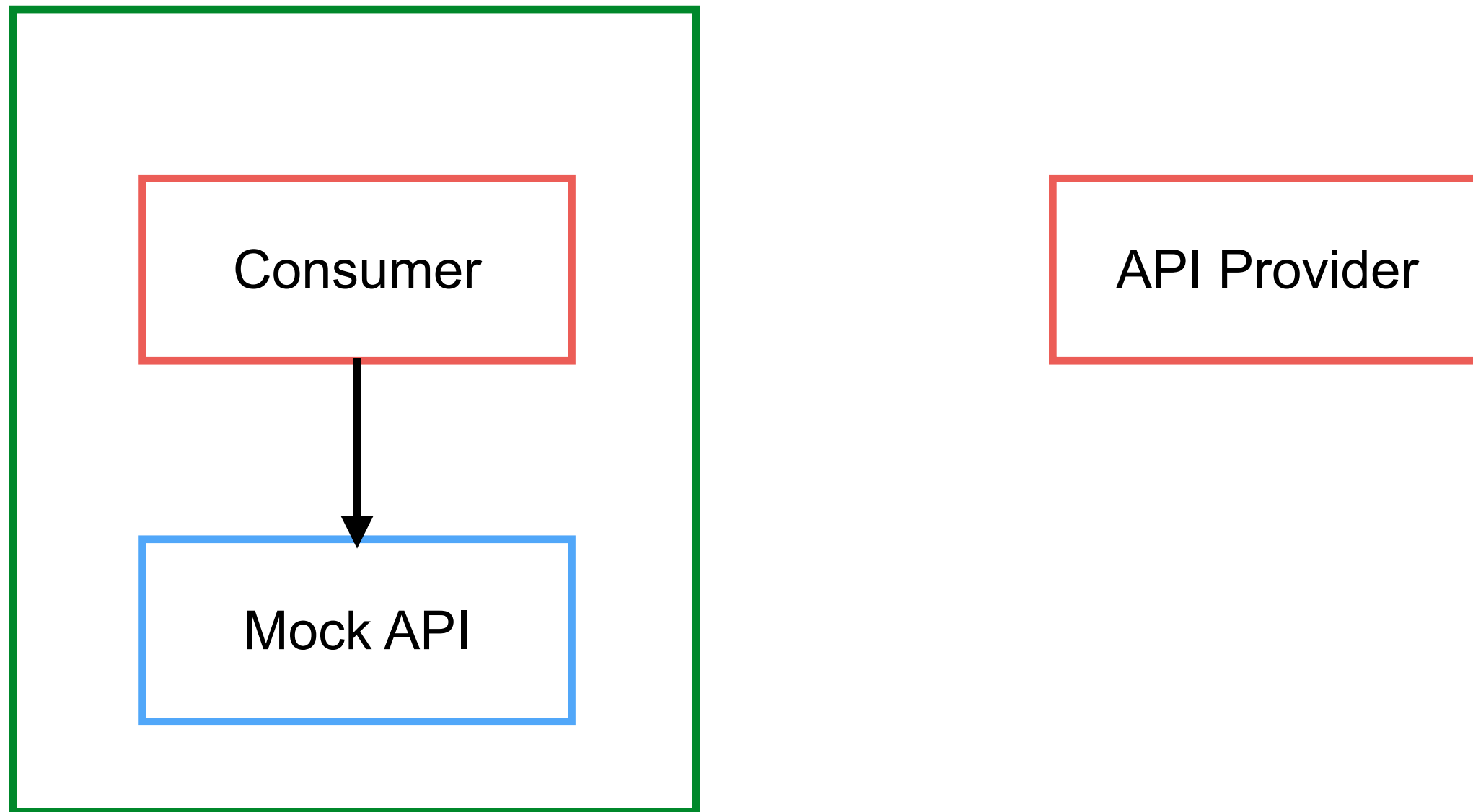
Slow feedback

Break easily

A lots of maintenance



Component test



Improve collaboration and confidence with Contract testing



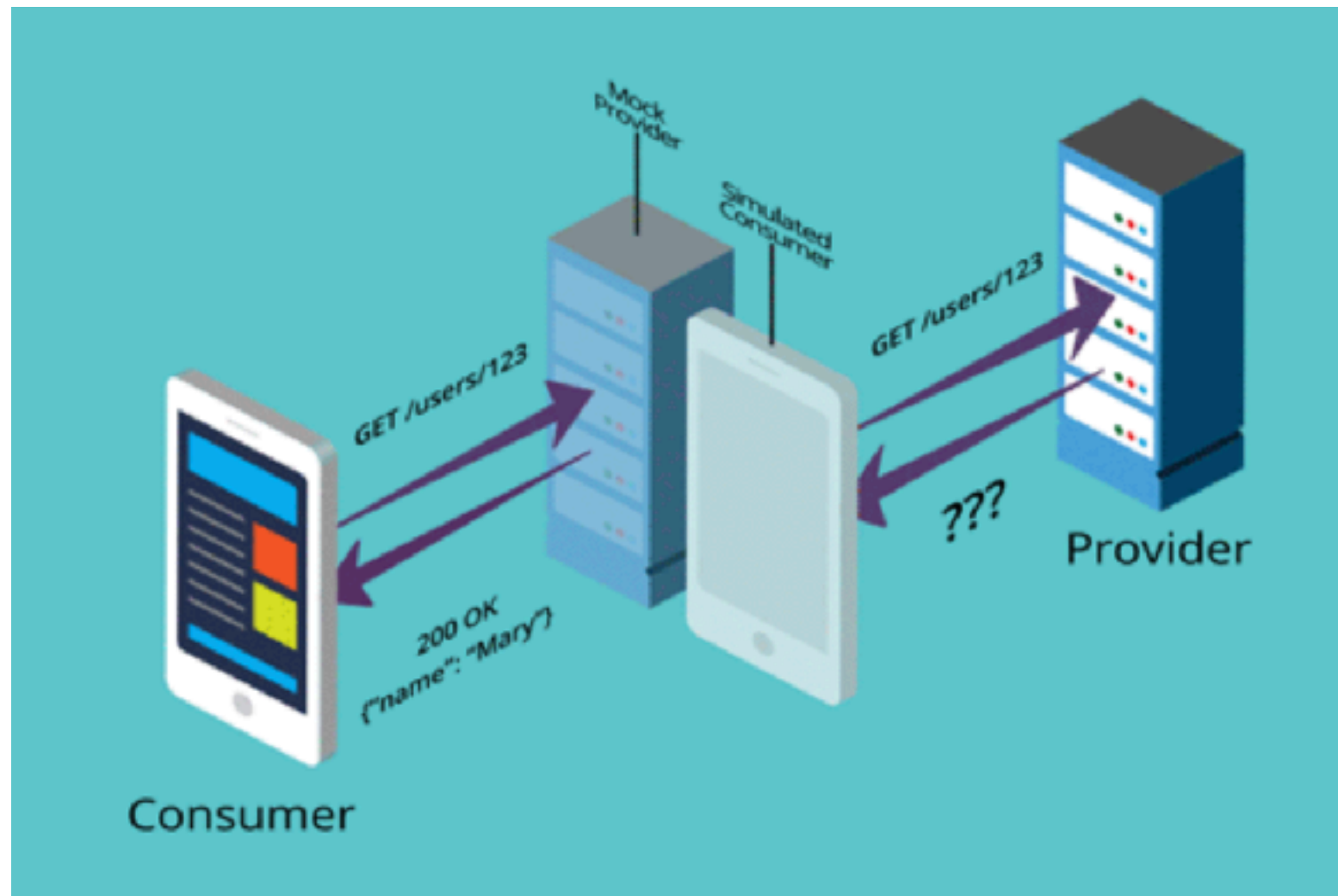
Contract Testing

PACT

<https://docs.pact.io/>



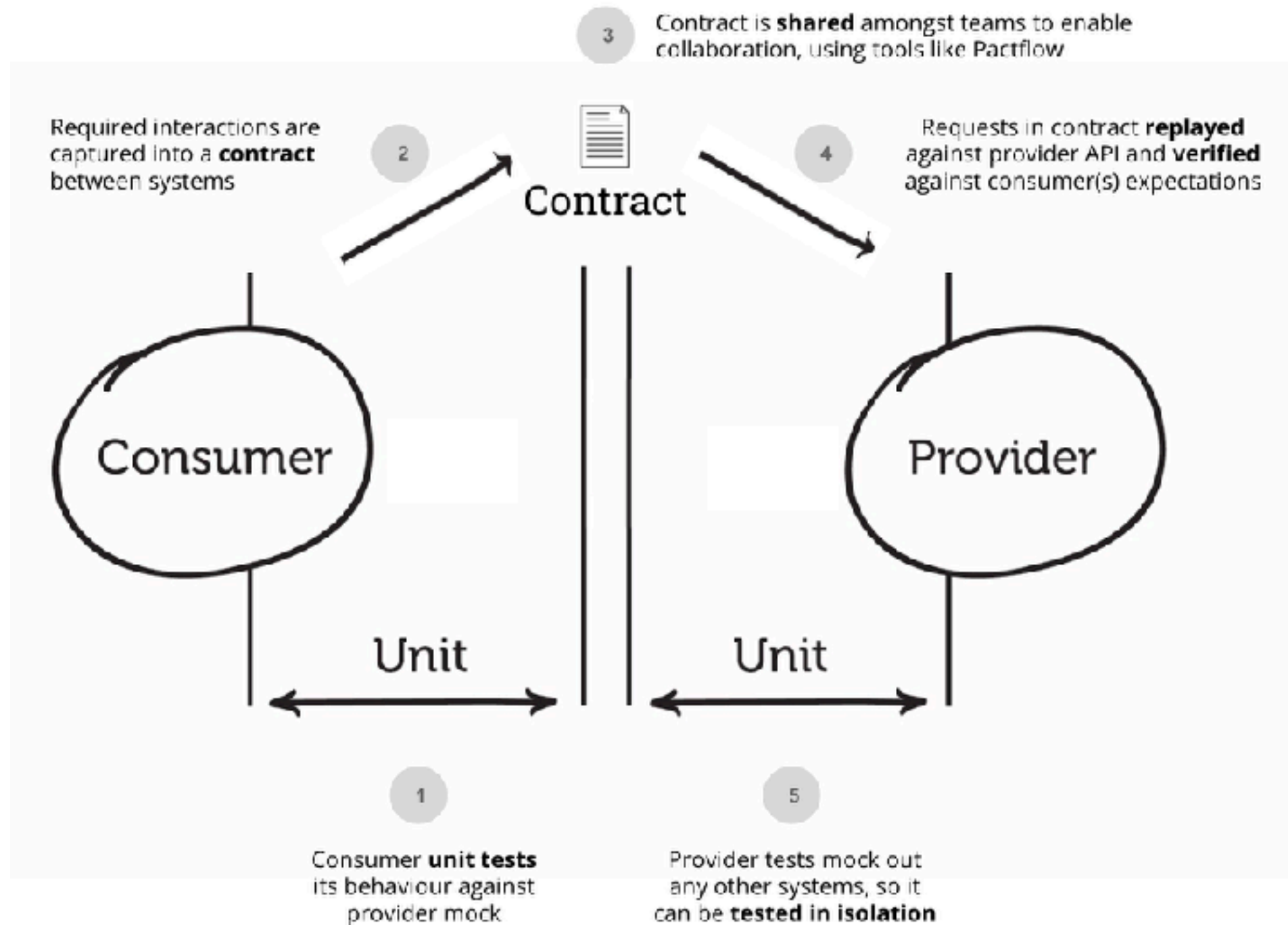
Contract Testing



<https://pactflow.io/how-pact-works/#slide-2>



Contract Testing

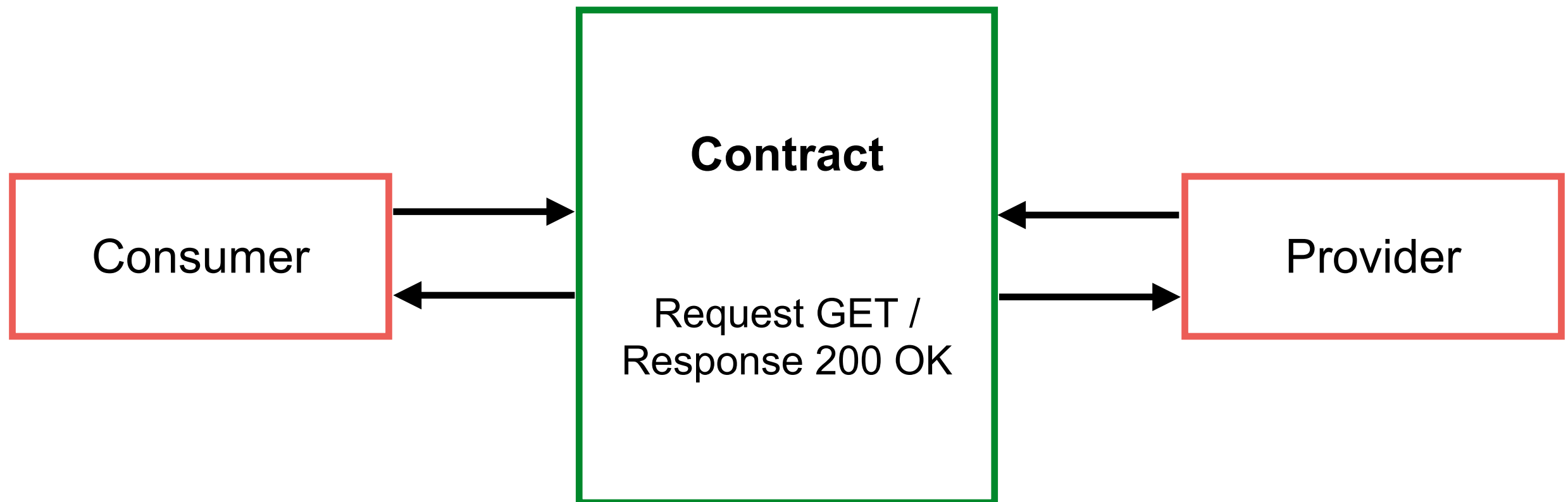


<https://pactflow.io/how-pact-works/#slide-1>



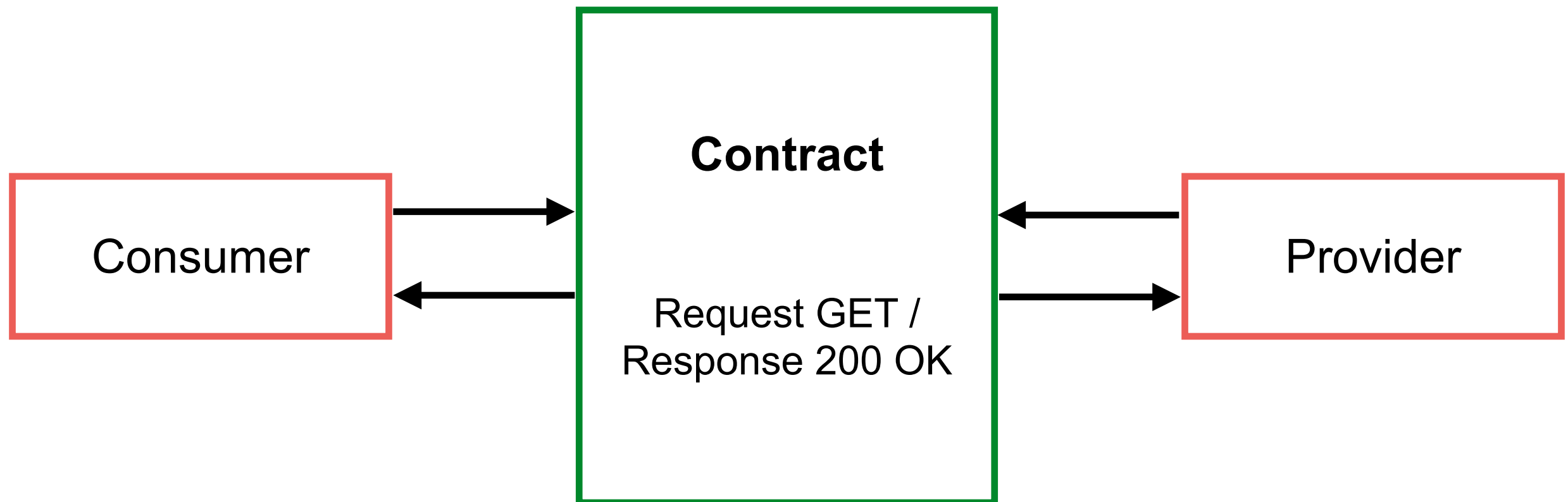
Contract Testing

Check and ensure the interoperability of software application in isolation

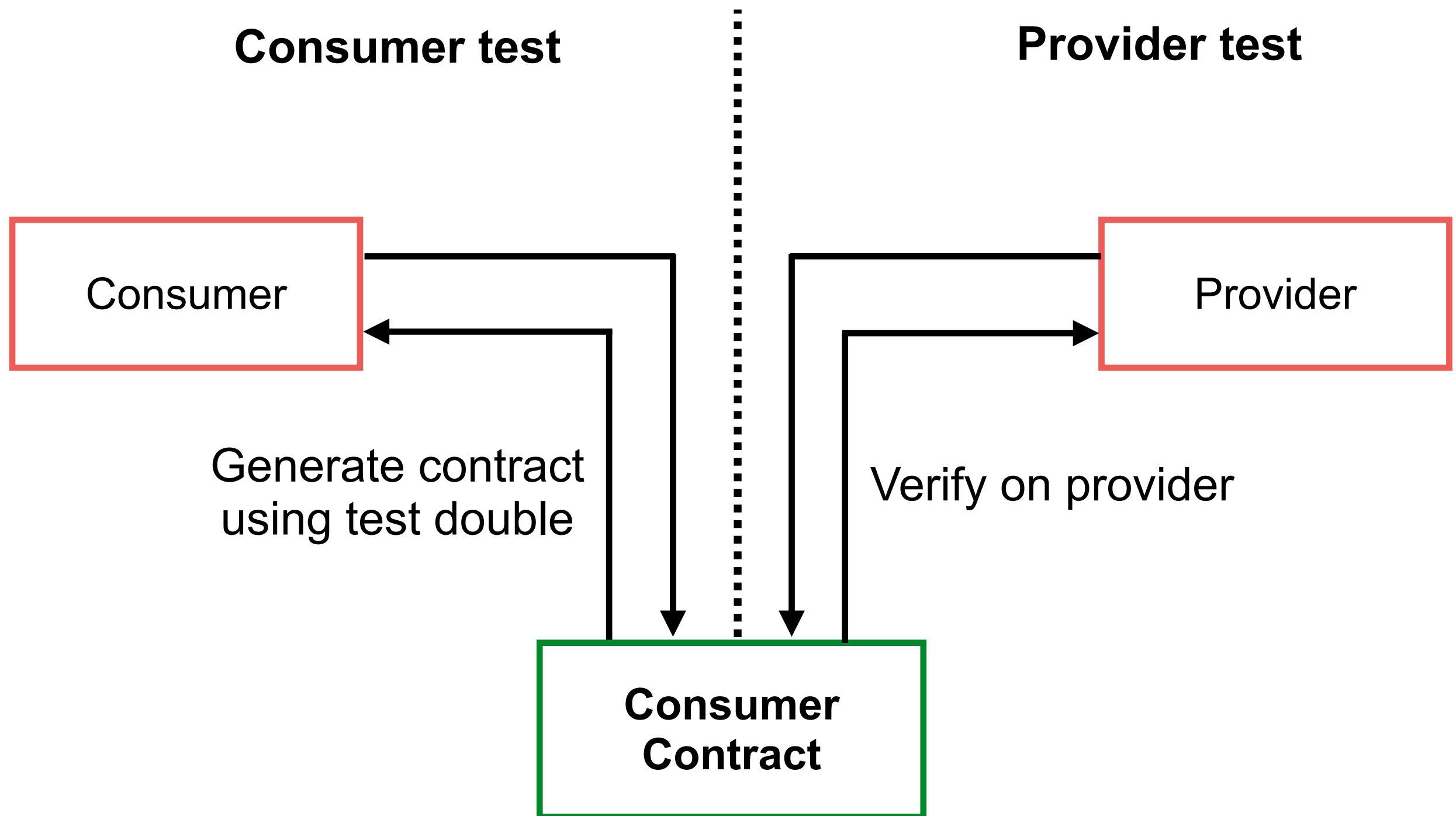


Contract Testing

Enable teams to deploy microservices independently



Consumer-driven Contract Testing



Working with Pact

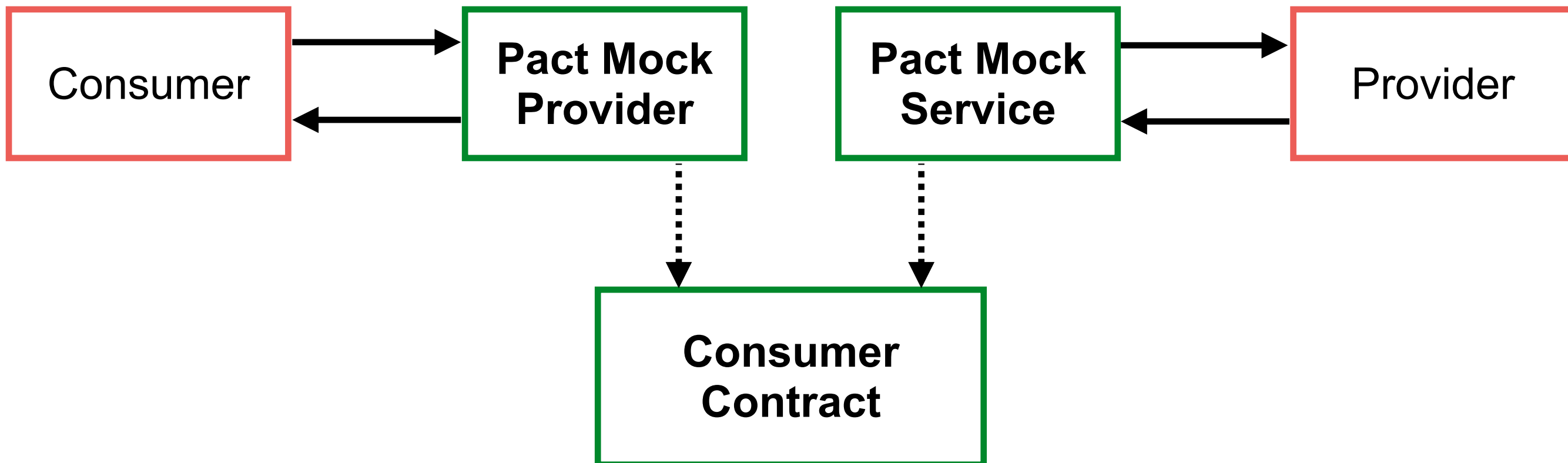


Working with Pact

Code-first consumer-driven contract testing tool

Consumer test

Provider test



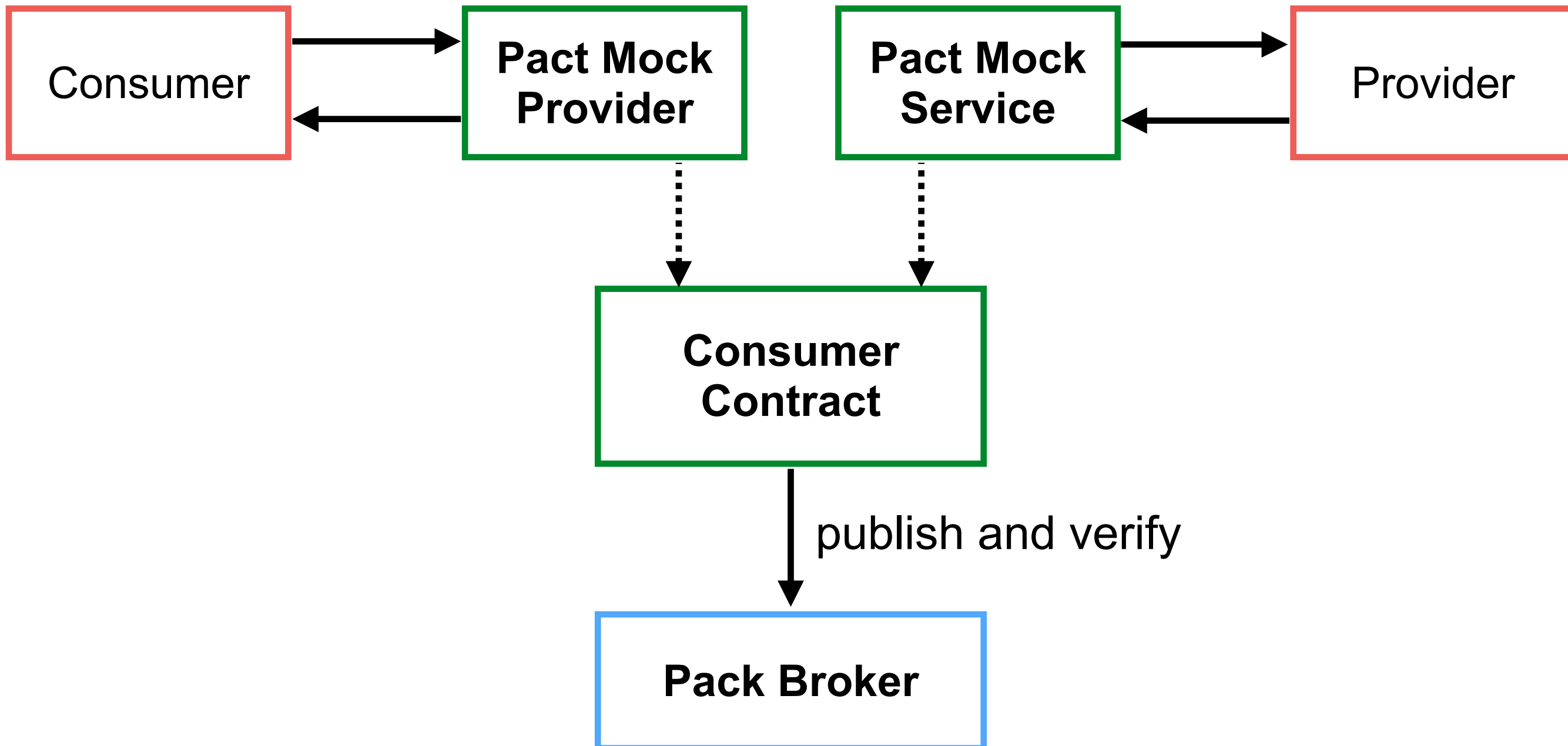
Pact broker

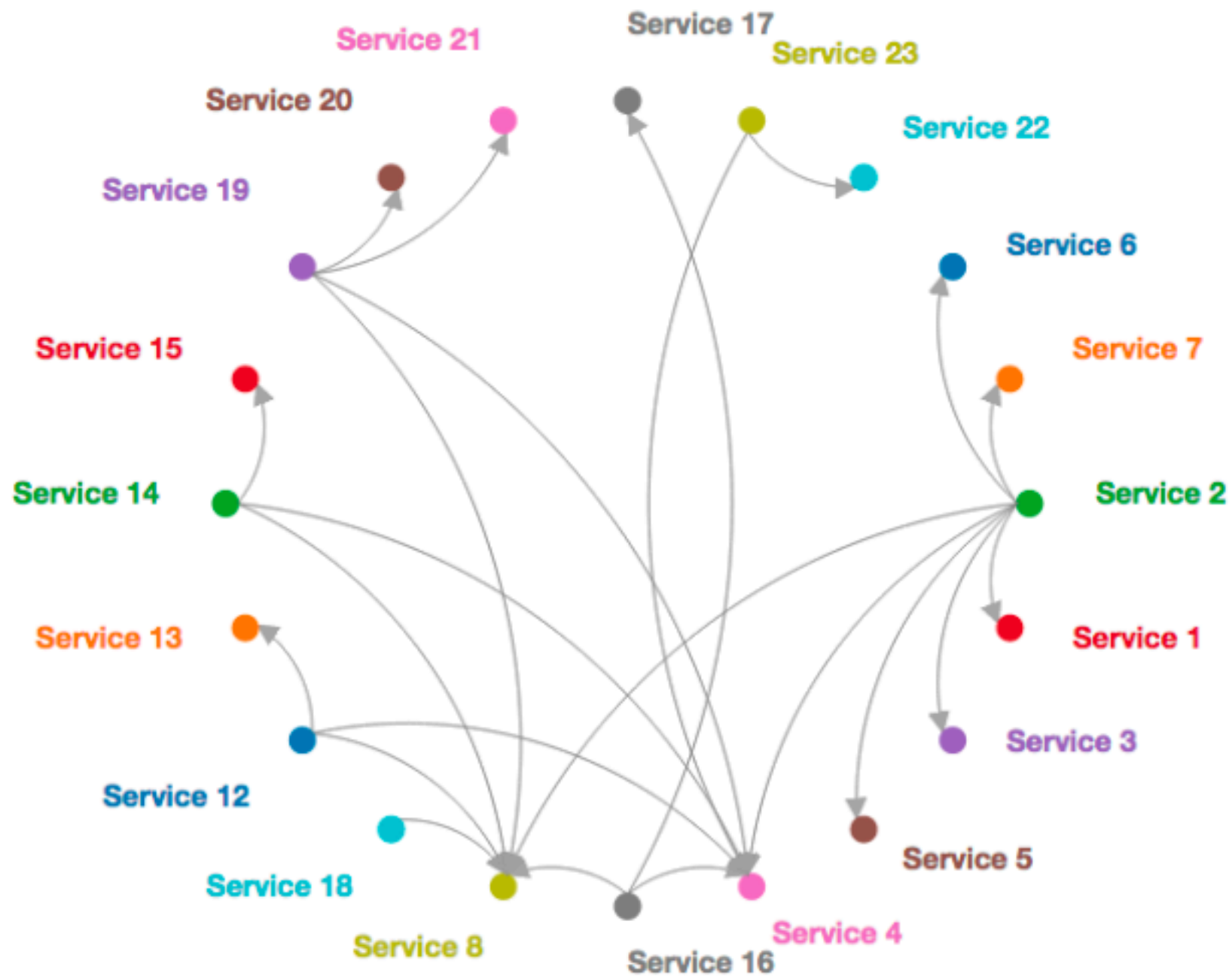


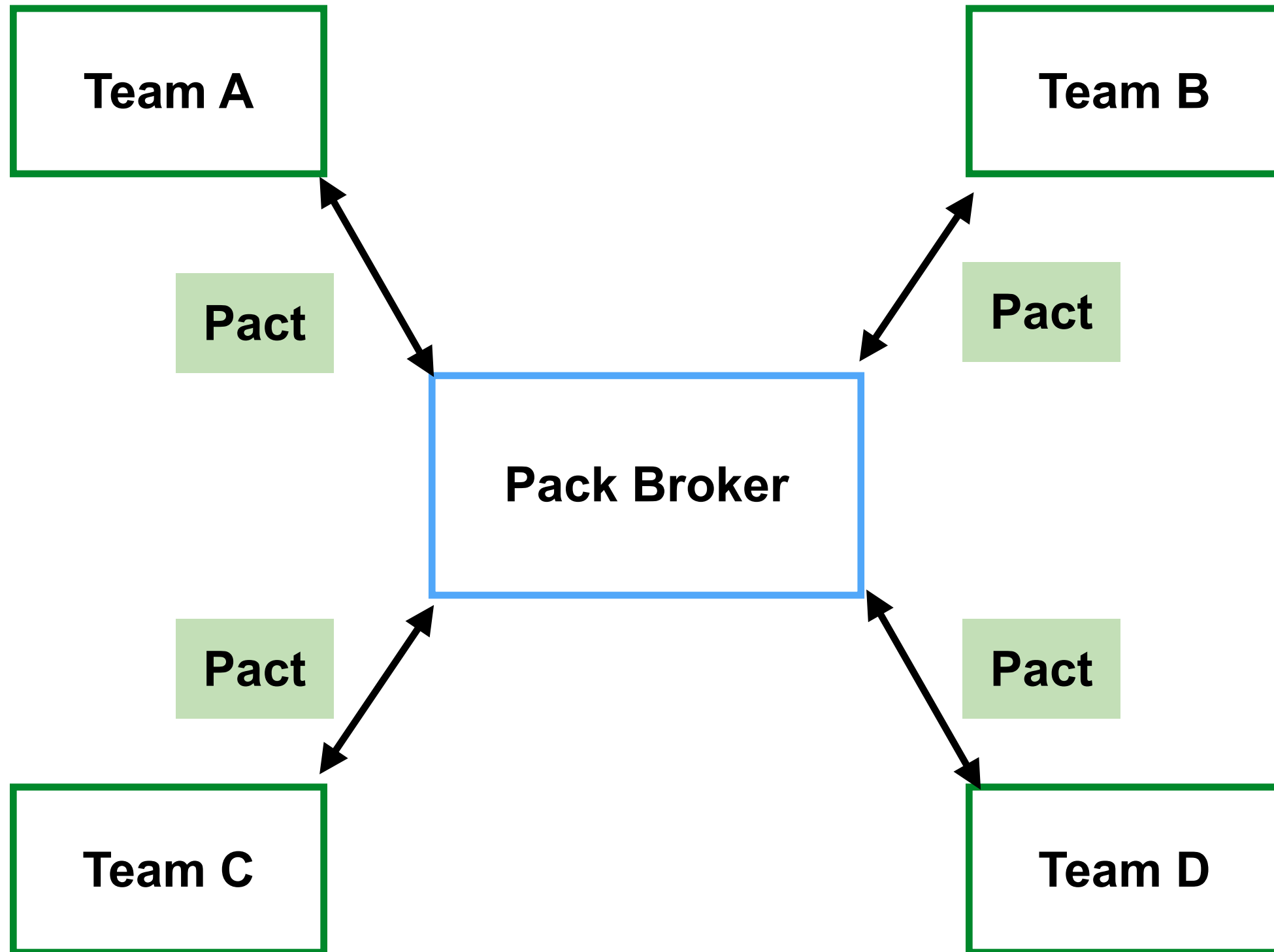
Working with Pact broker

Consumer test

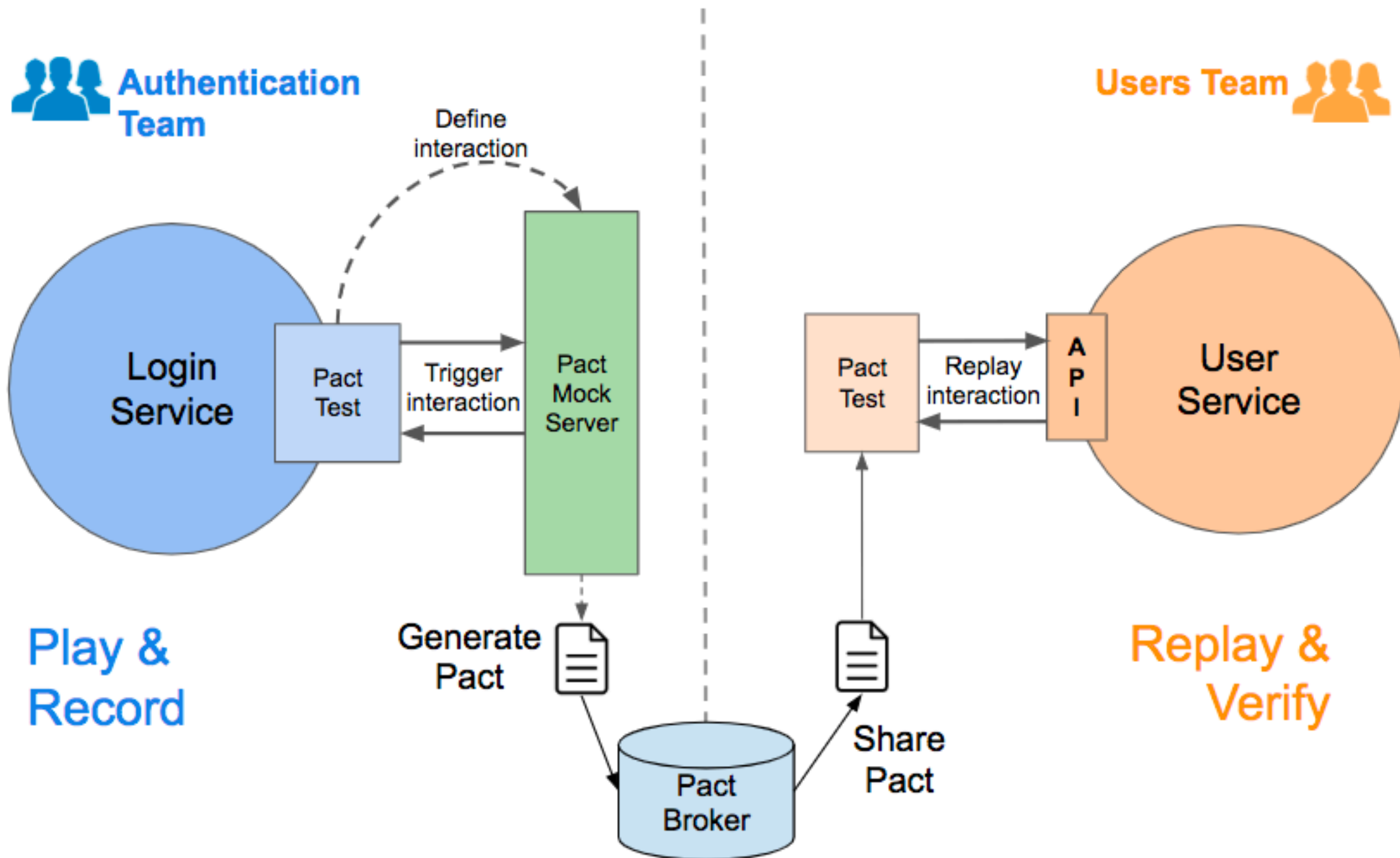
Provider test







Example with Pact



Steps to use Pact

- Write code for consumer test
- Generate contract from consumer test
- Create Pact Mock Service to test the provider API
- Create Pact Mock Provider as mock API
- Publish contract to Pact broker
- Verify contract from Pact broker

<https://github.com/up1/course-contract-testing/wiki>



Write code for consumer test



Design First with APIs

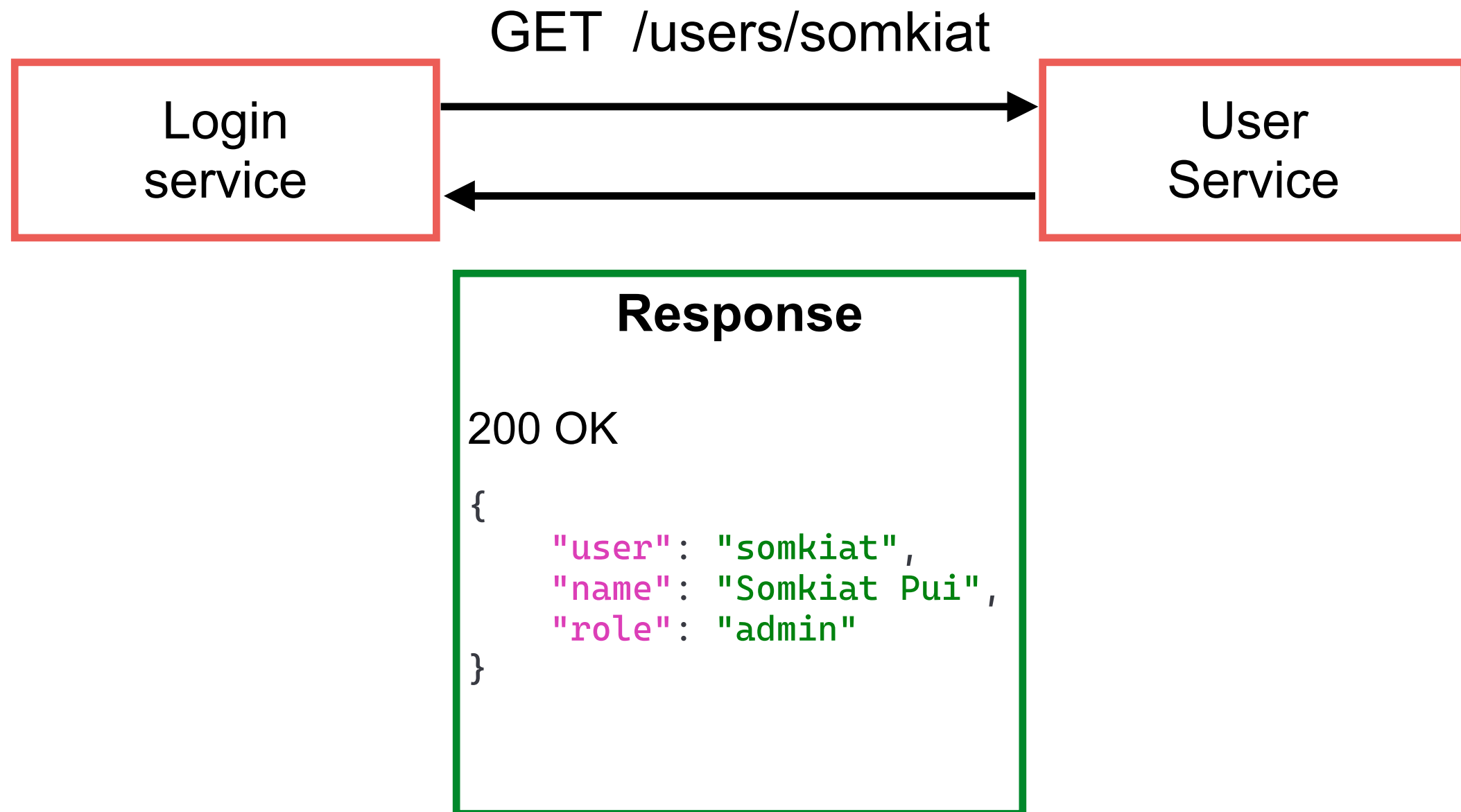
Design APIs specification

Create contract test

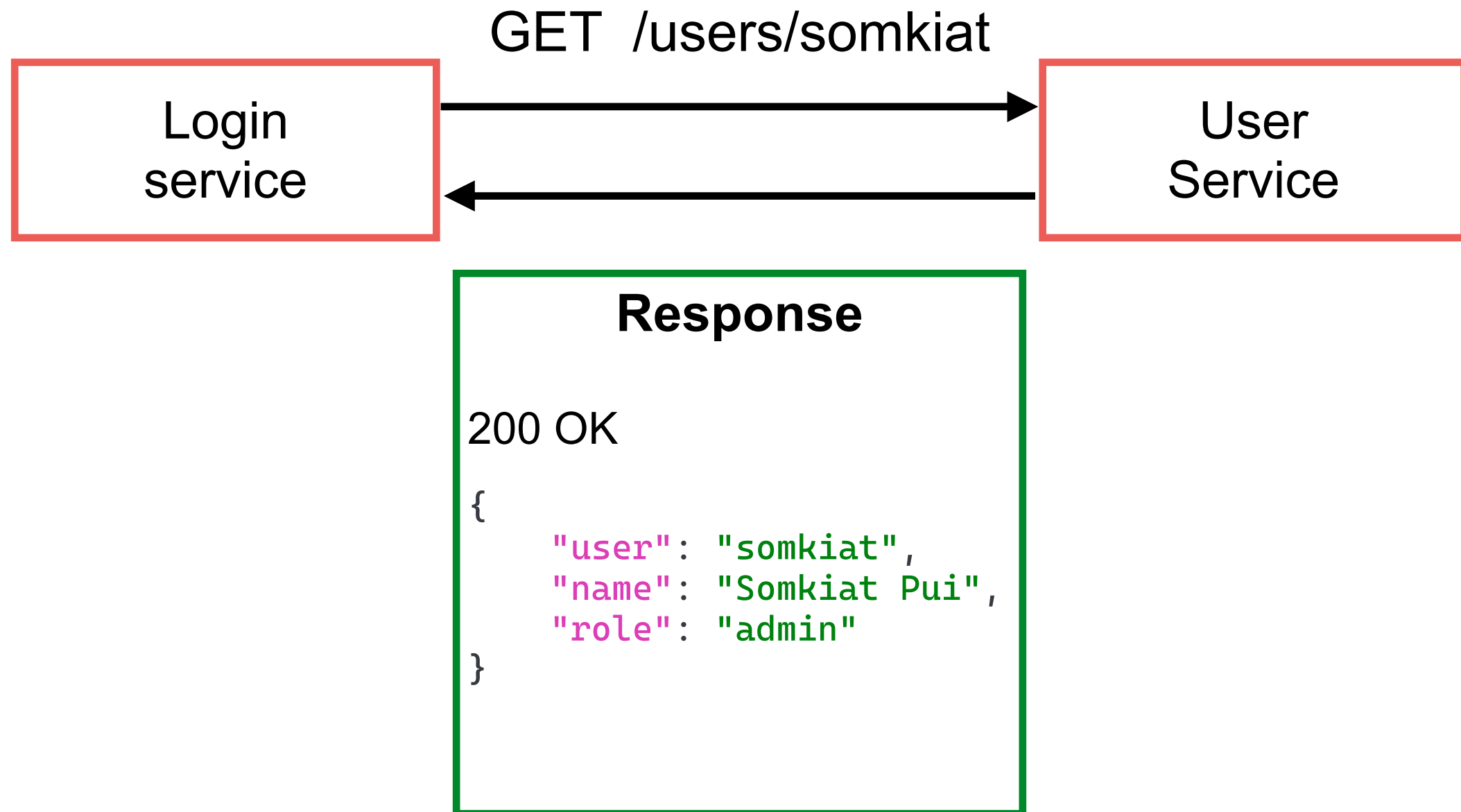
Generate contract from test



API



How do we test ?

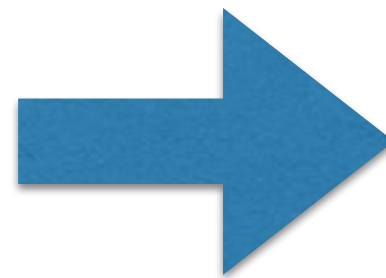


Change in User service !!

Response

200 OK

```
{  
  "user": "somkiat",  
  "name": "Somkiat Pui",  
  "role": "admin"  
}
```



Response

200 OK

```
{  
  "user": "somkiat",  
  "name": "Somkiat Pui",  
  "roles": ["admin", "user"]  
}
```



Change in User service !!

User
Service

Testing

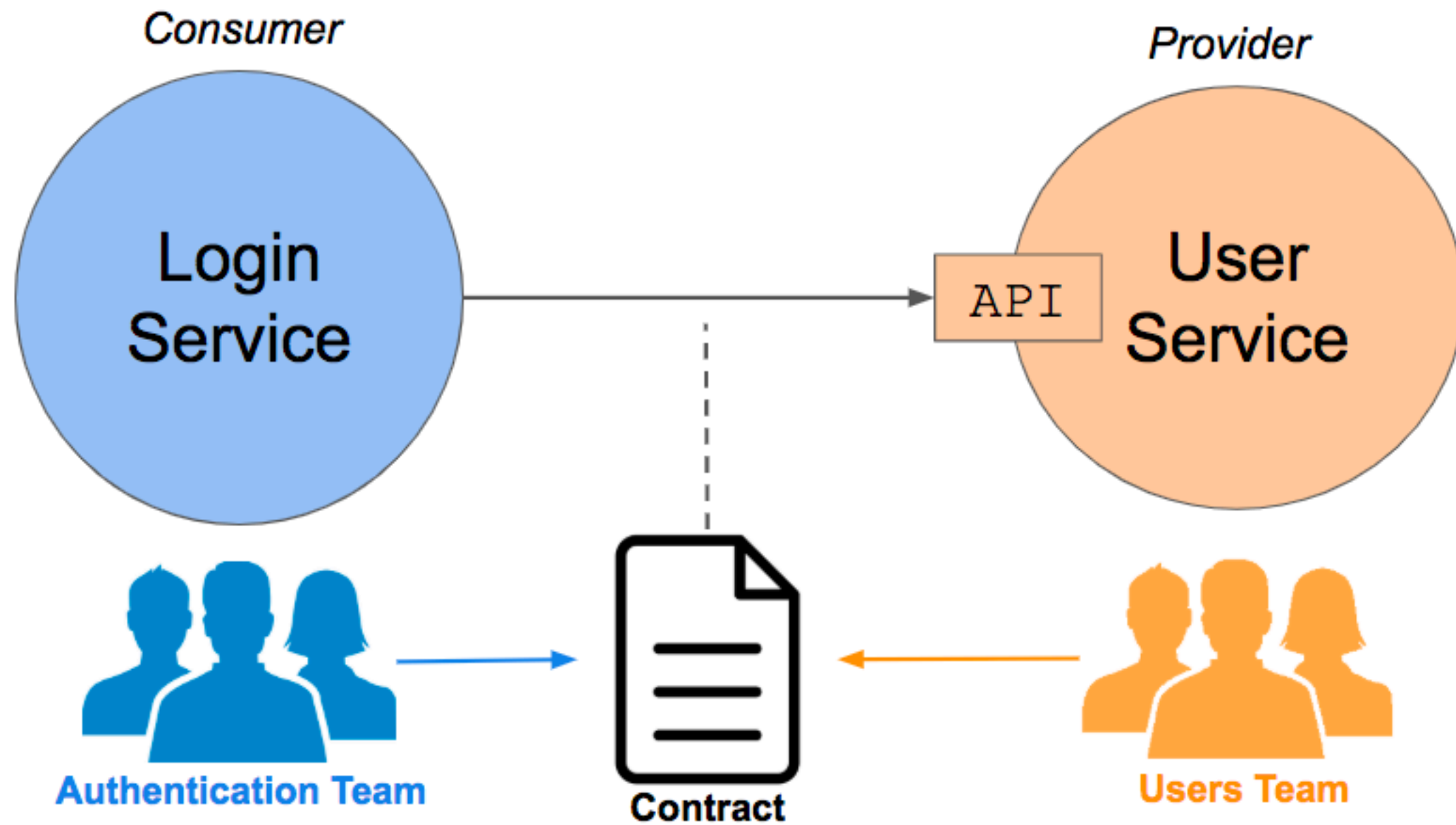
Deploy

Login
service

Testing ?

Deploy ?



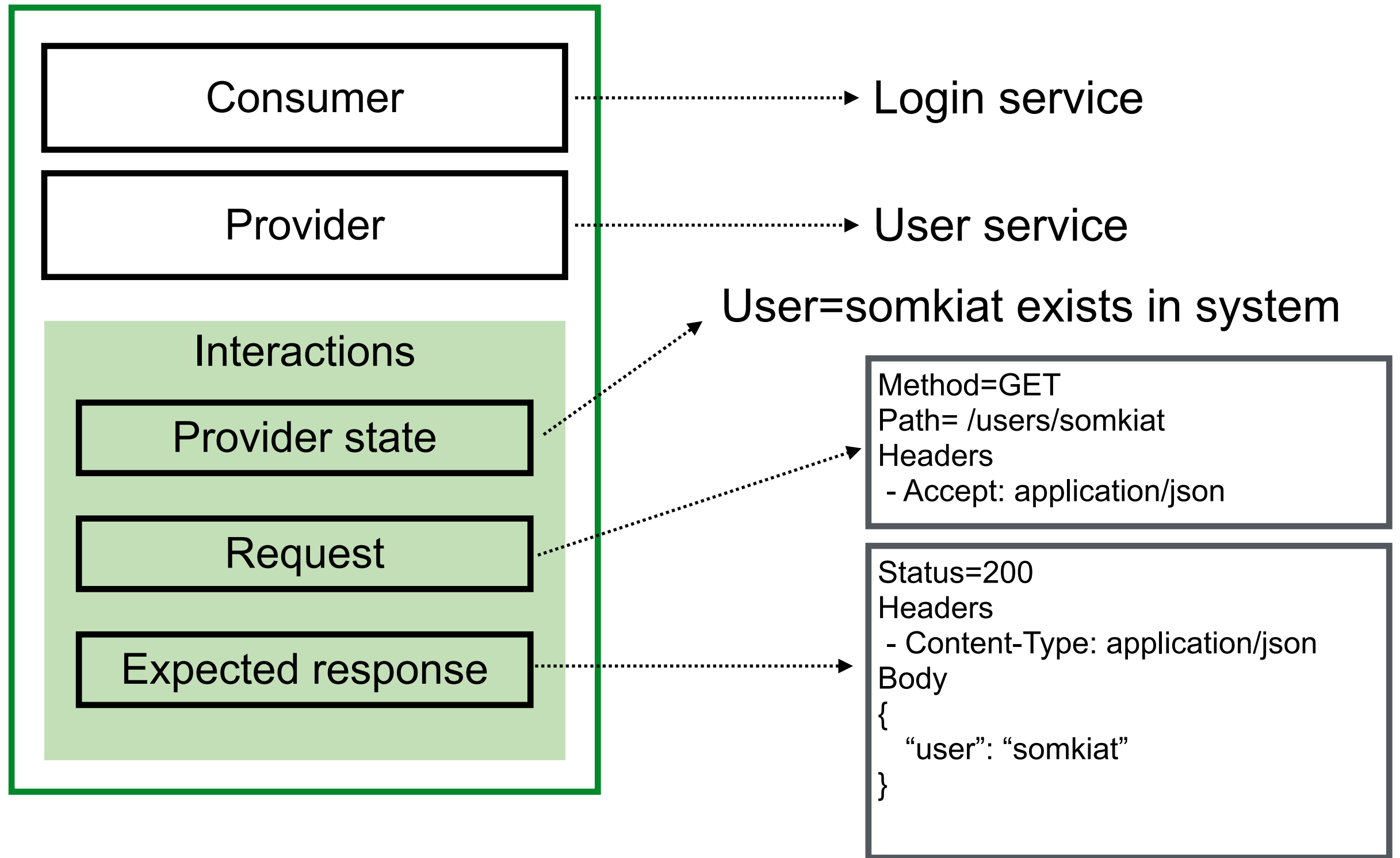


Structure of contract test

Consumer and Provider
Interaction, provider state
Request and expected response



Structure of contract test

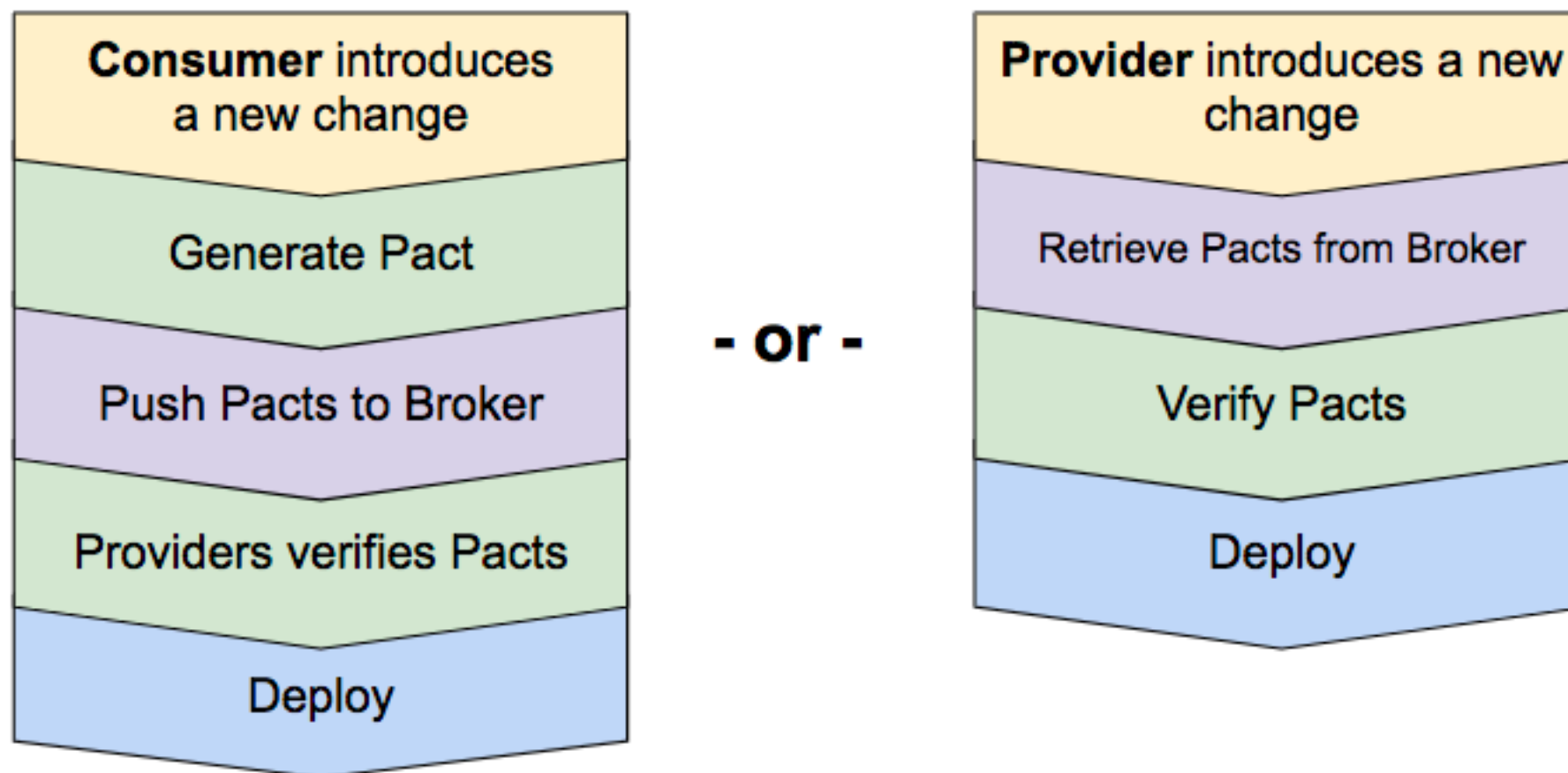


Working in CI/CD pipeline



CI/CD pipeline

When code/api changes !!



CI/CD pipeline

When code/api changes !!

Implement
changes

Generate Pact
Contract test

Push pact to
broker

Each provider
verified Pacts

Deploy service



Contract test

Provider state setup for each interaction

Don't limit contracts to happy paths

Automation process in deployment pipeline



Q/A

