

Microservice Testing





Somkiat Puisungnoen

Somkiat Puisungnoen

Update Info 1

View Activity Log 10+

...
Timeline About Friends 3,138 Photos More

When did you work at Opendream?
... 22 Pending Items

Post Photo/Video Live Video Life Event

What's on your mind?

Public Post

Intro
Software Craftsmanship

Software Practitioner at สยามชัมนาภิกิจ พ.ศ. 2556

Agile Practitioner and Technical at SPRINT3r

Somkiat Puisungnoen 15 mins · Bangkok · ⚙️

Java and Bigdata





Page

Messages

Notifications 3

Insights

Publishing Tools

Settings

Help ▾



somkiat.cc

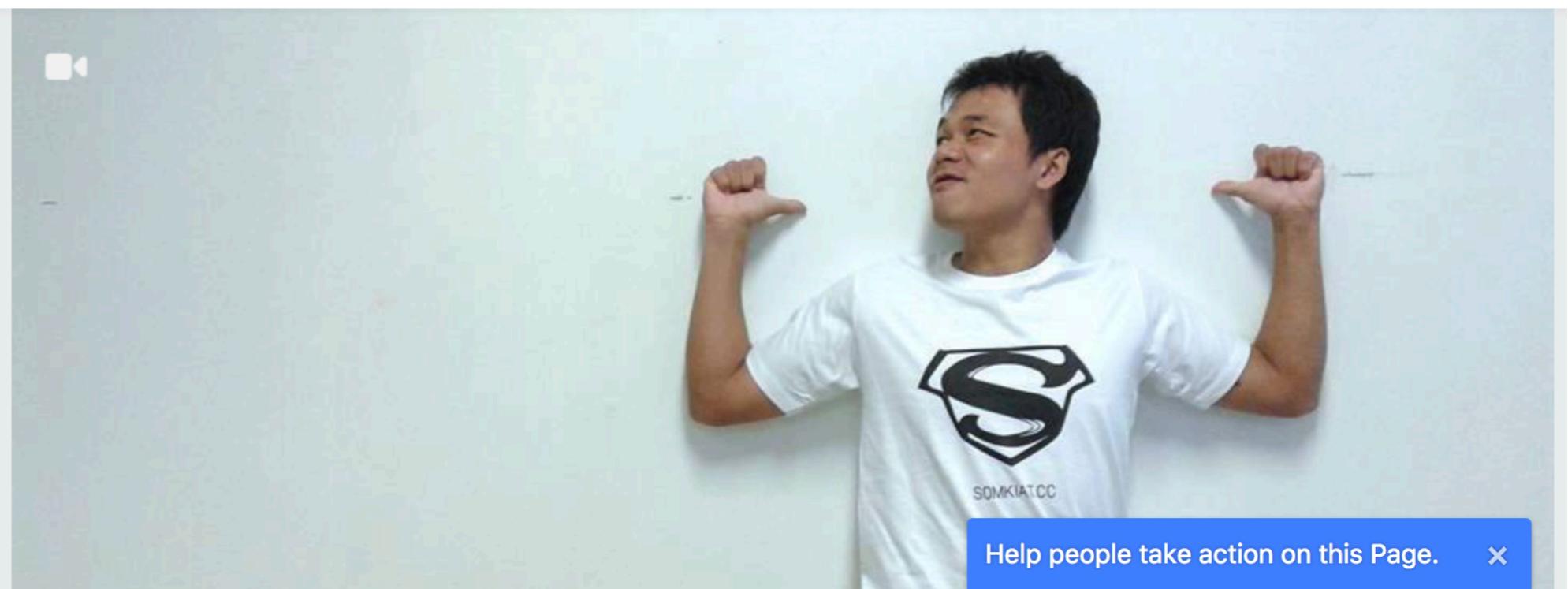
@somkiat.cc

Home

Posts

Videos

Photos



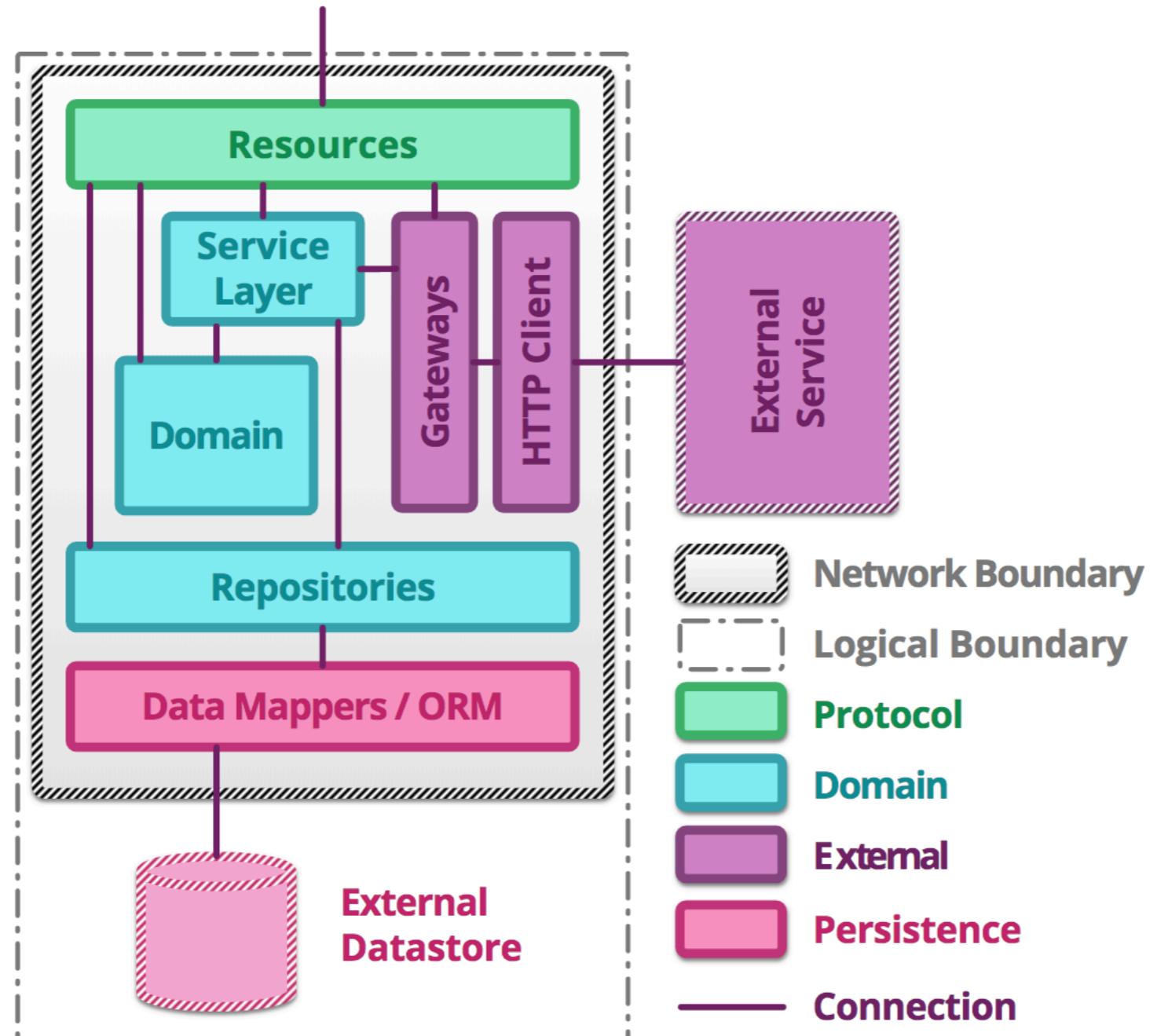
<https://github.com/up1/course-microservice-testing>



Microservice Testing



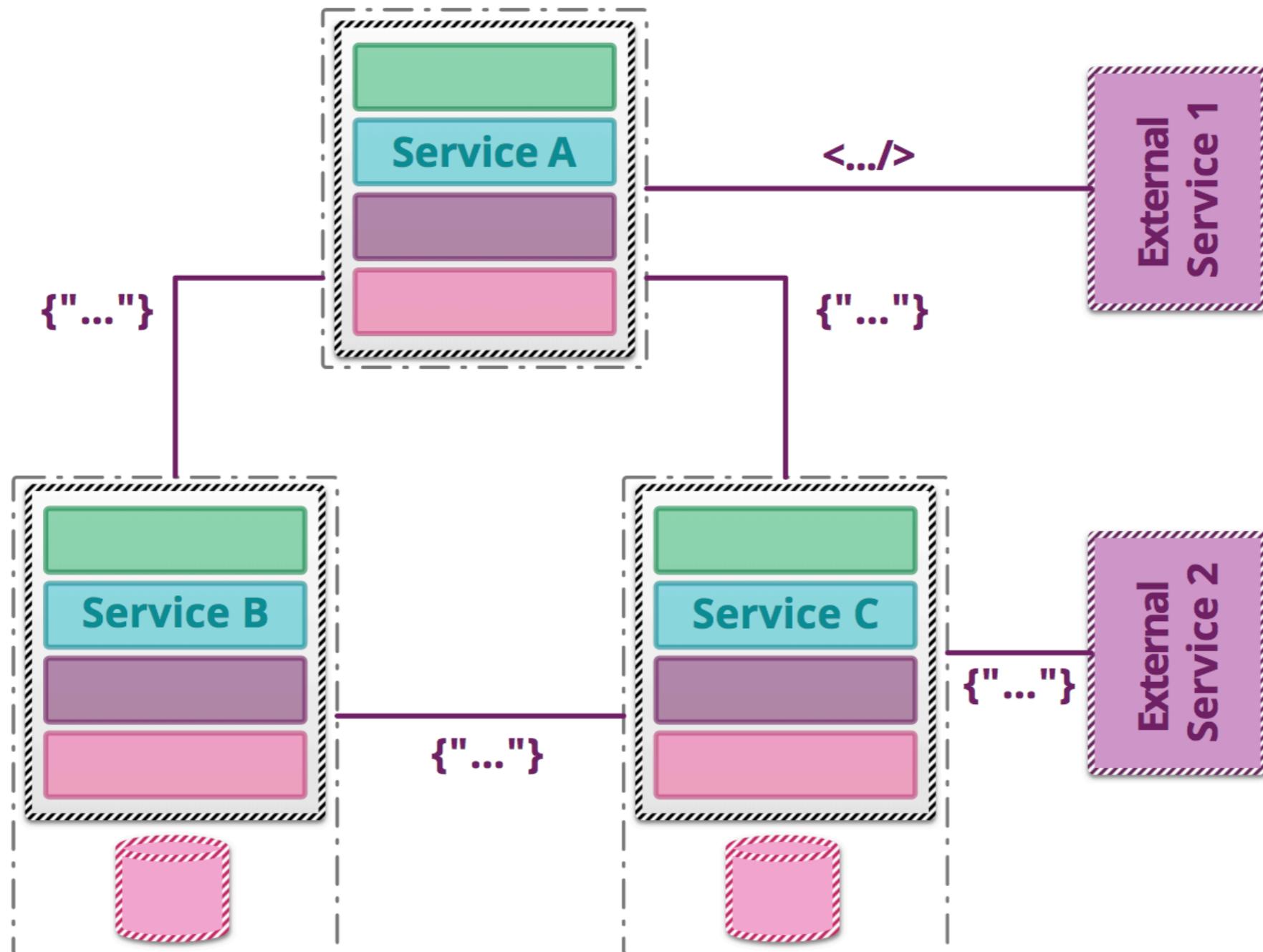
Service structure



<https://martinfowler.com/articles/microservice-testing>



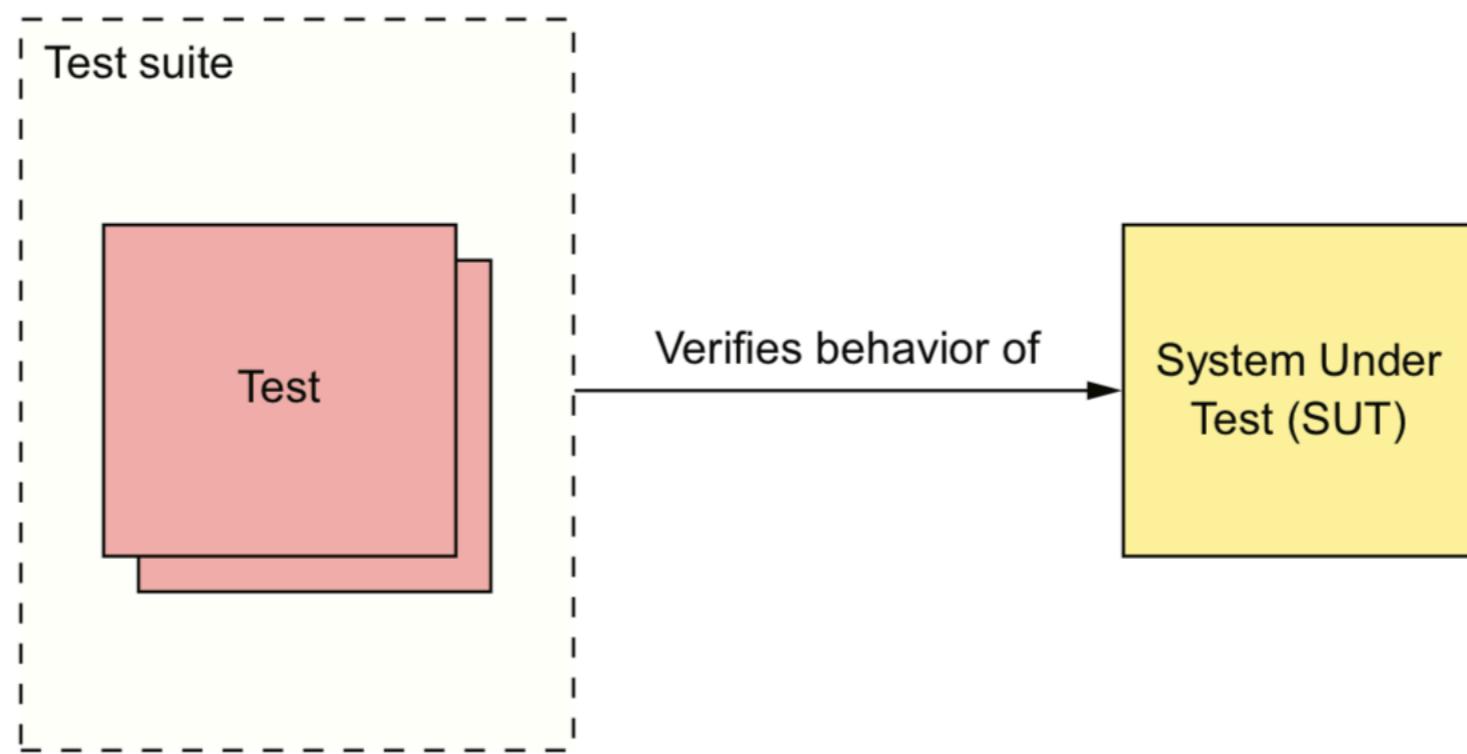
Multiple services



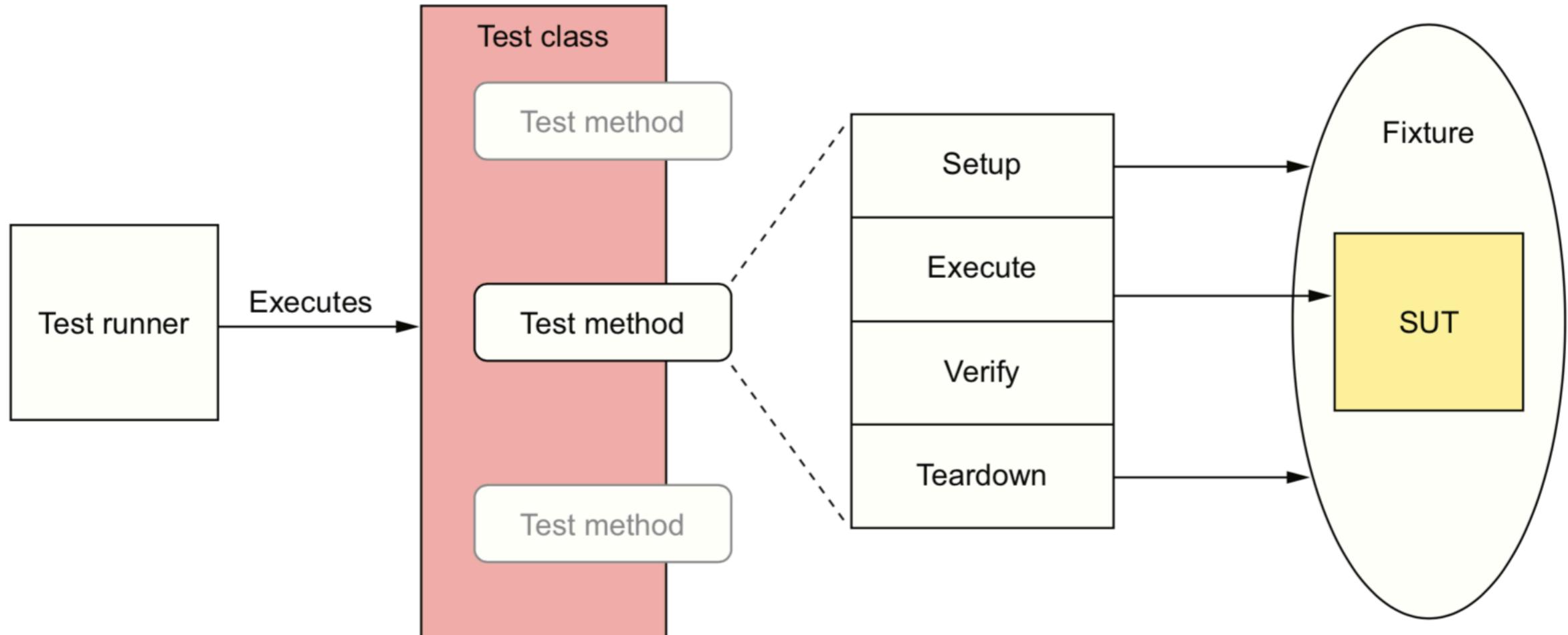
<https://martinfowler.com/articles/microservice-testing>



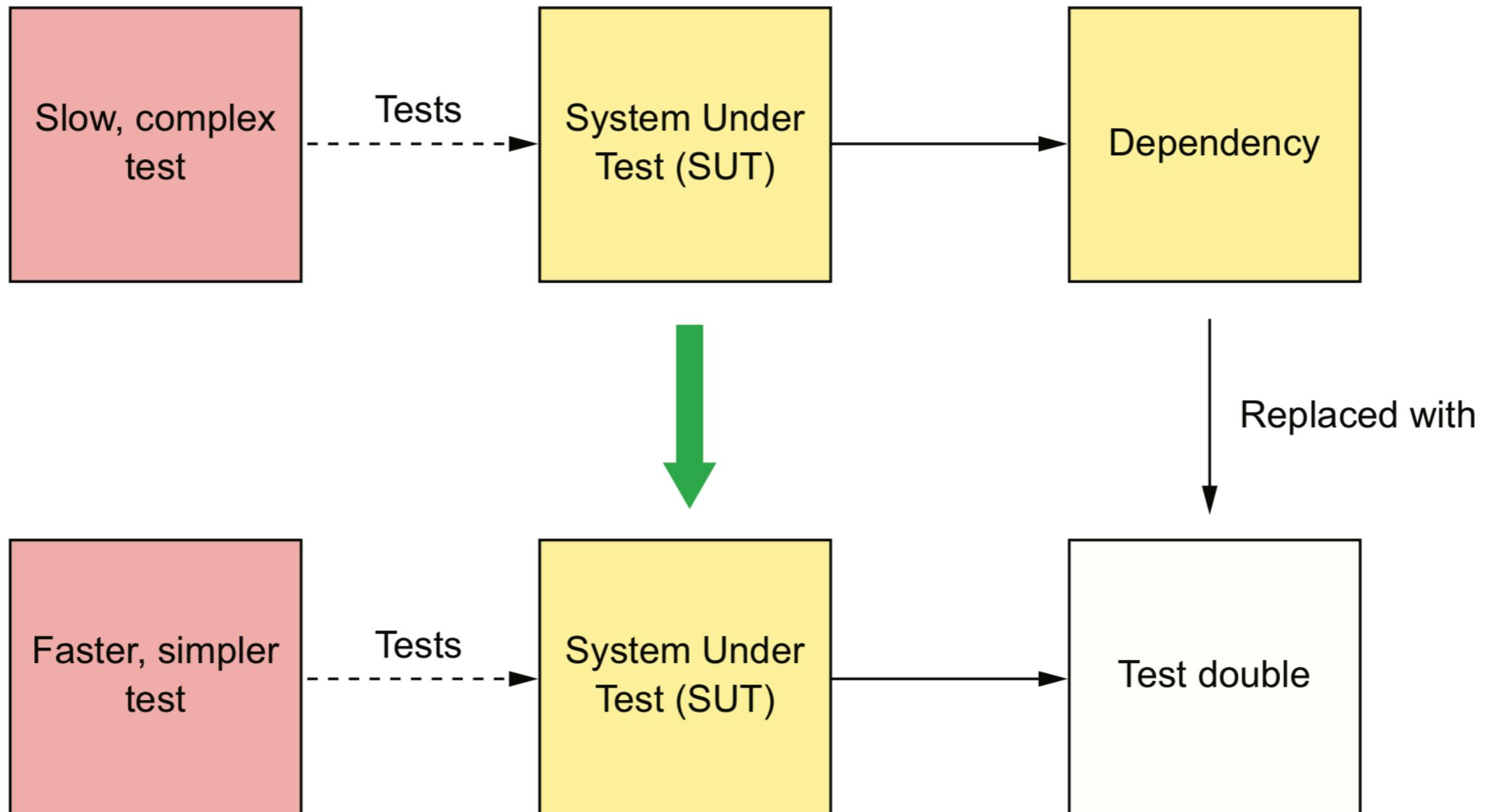
Testing



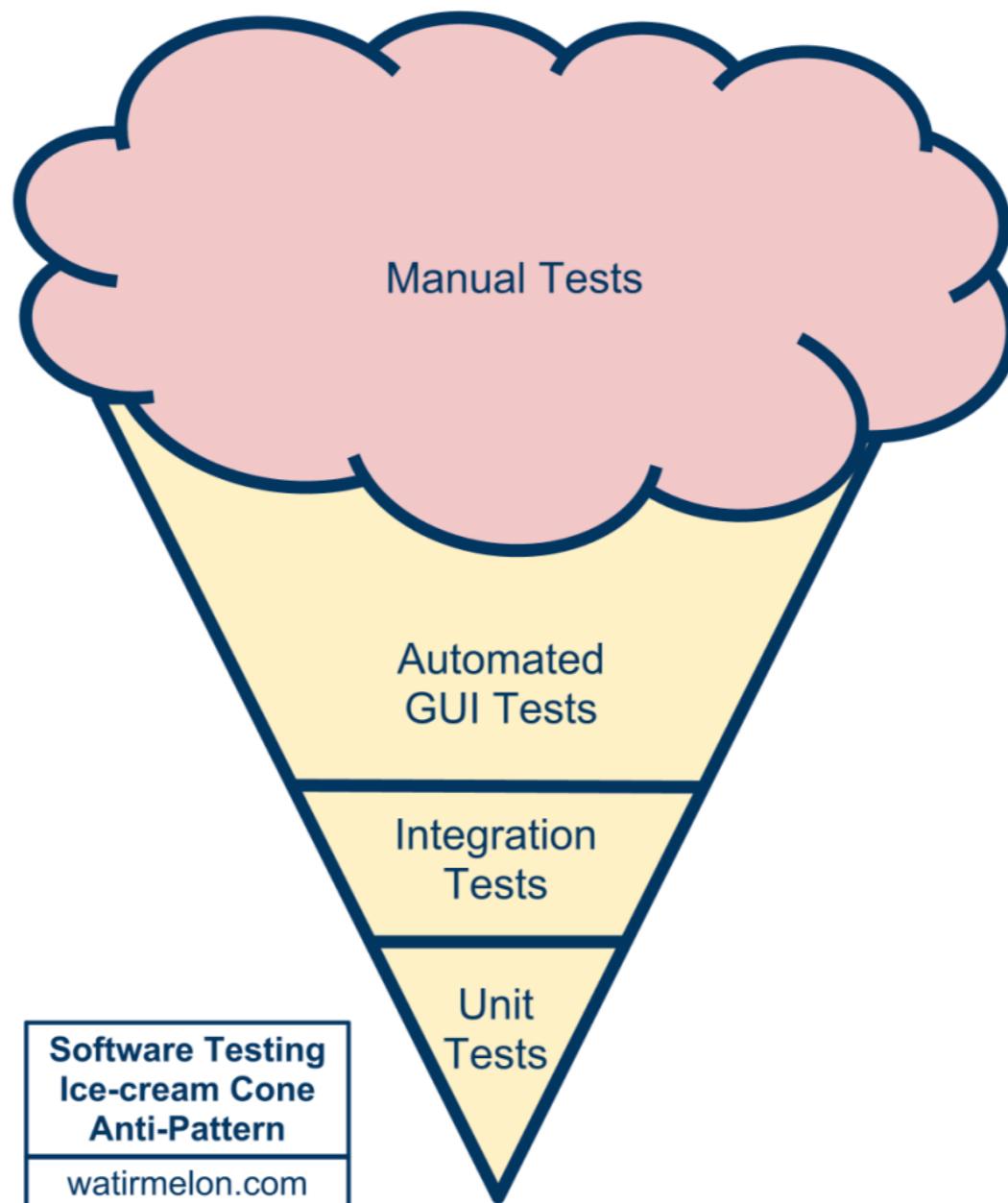
Structure of Automated tests



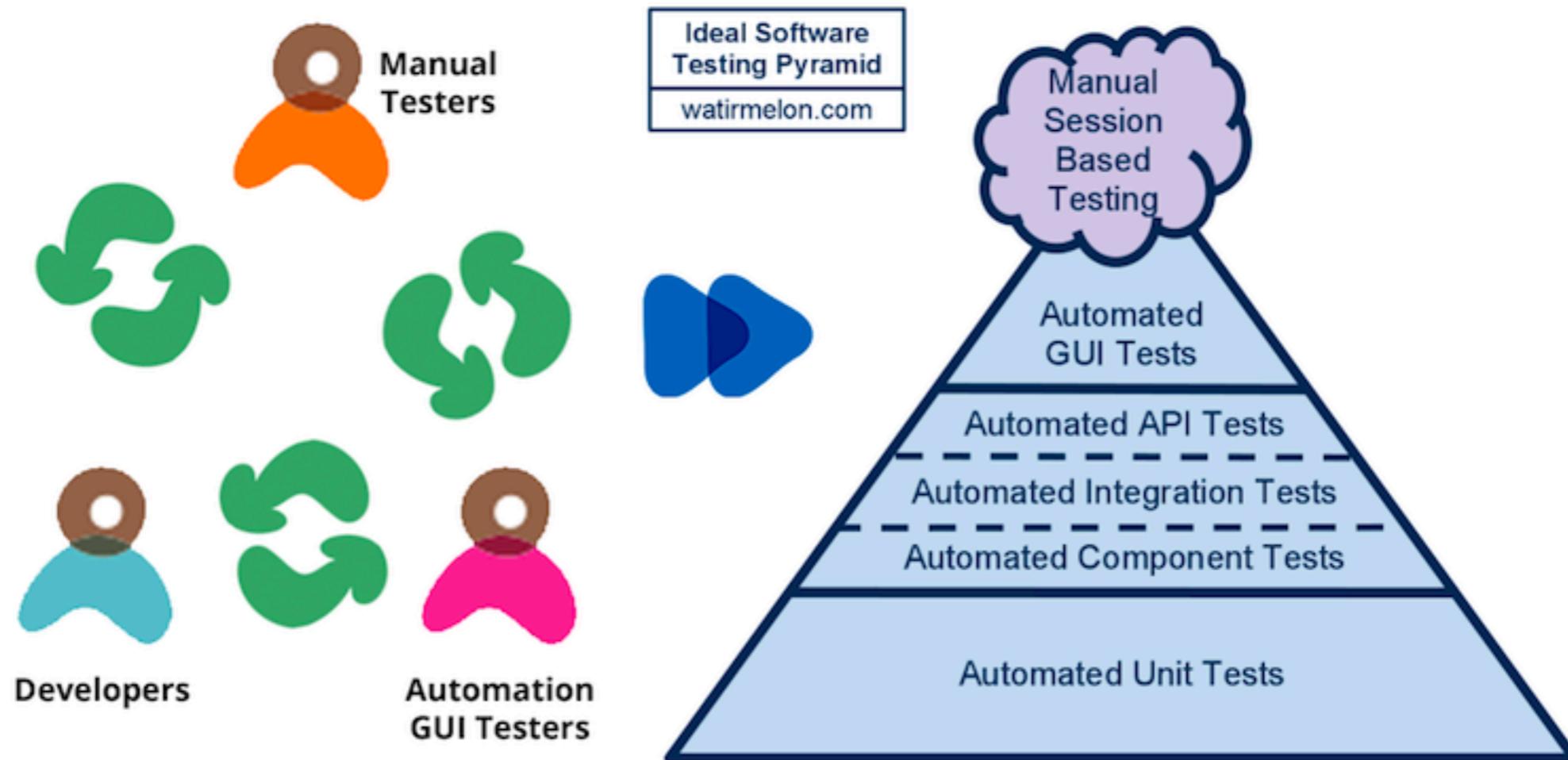
Working with Test double

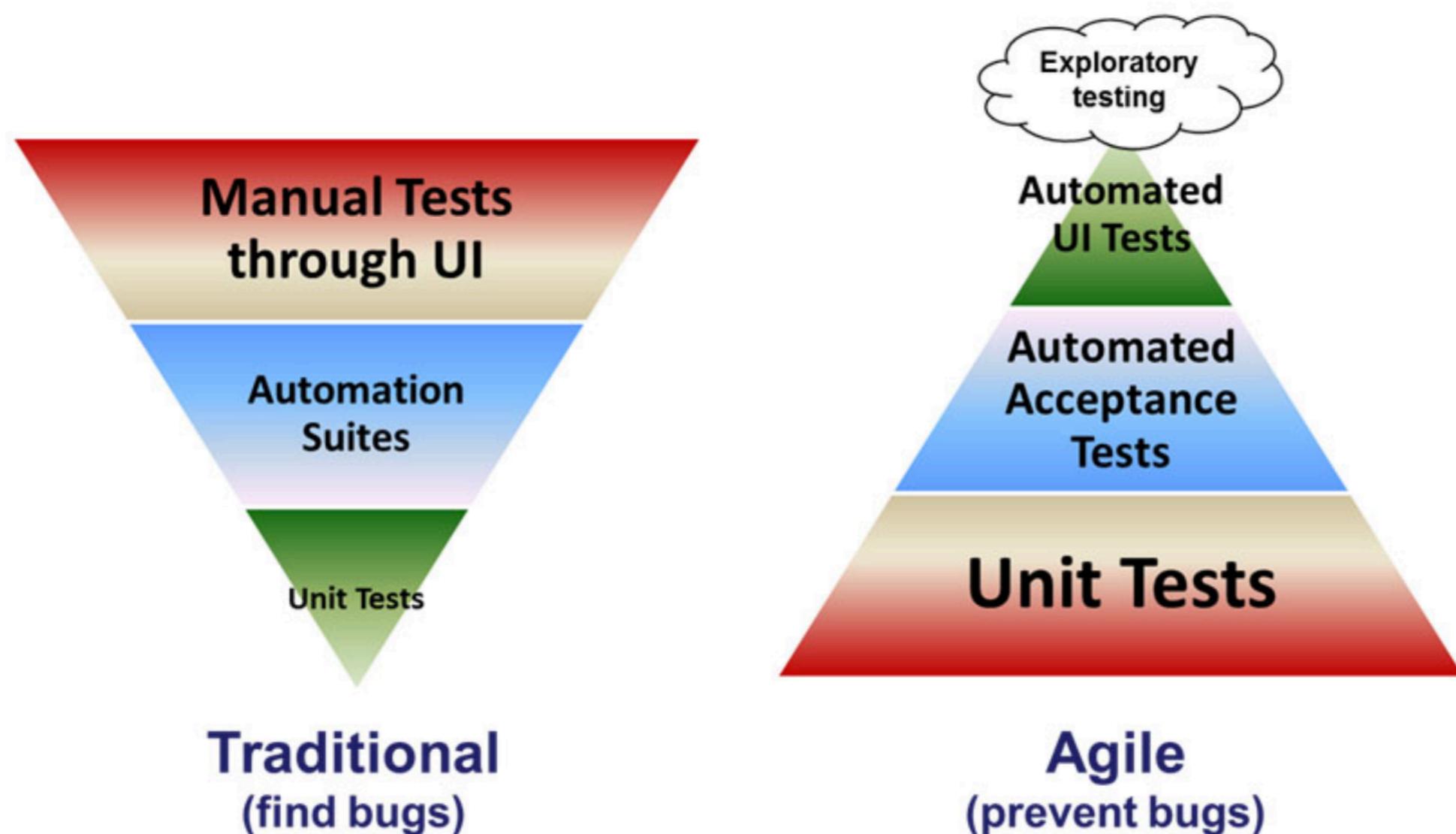


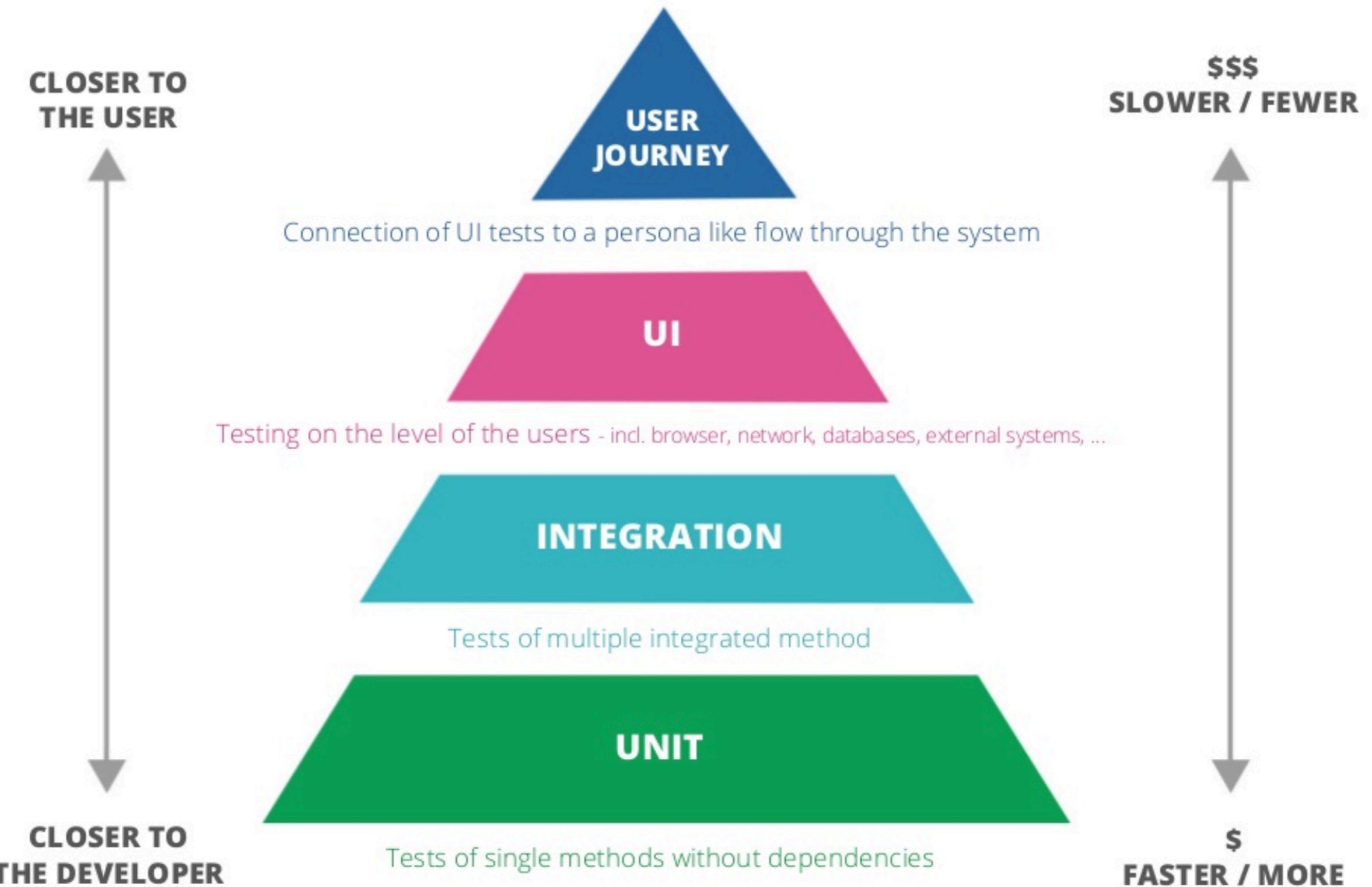
Ice-cream testing



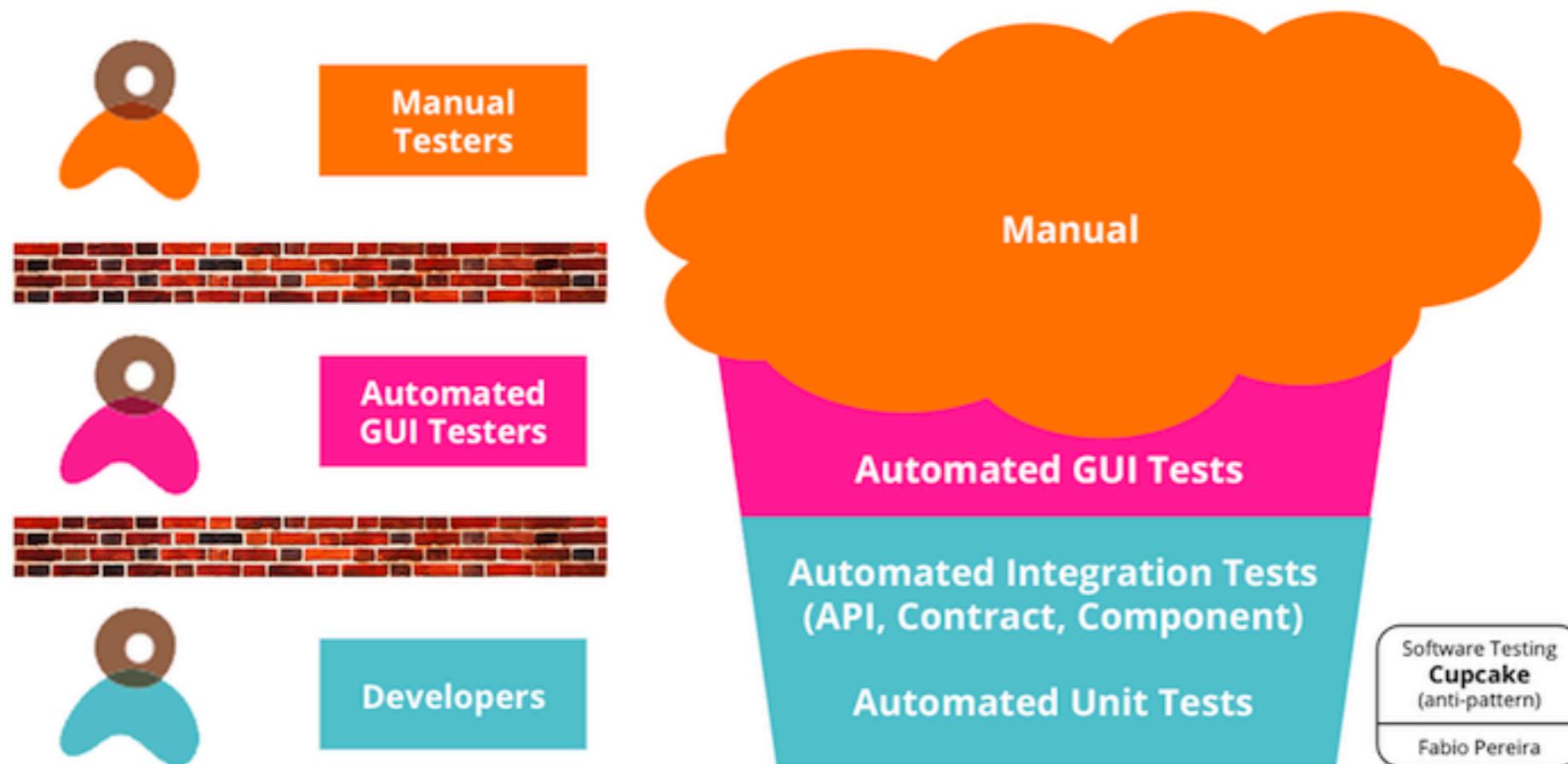
Pyramid testing



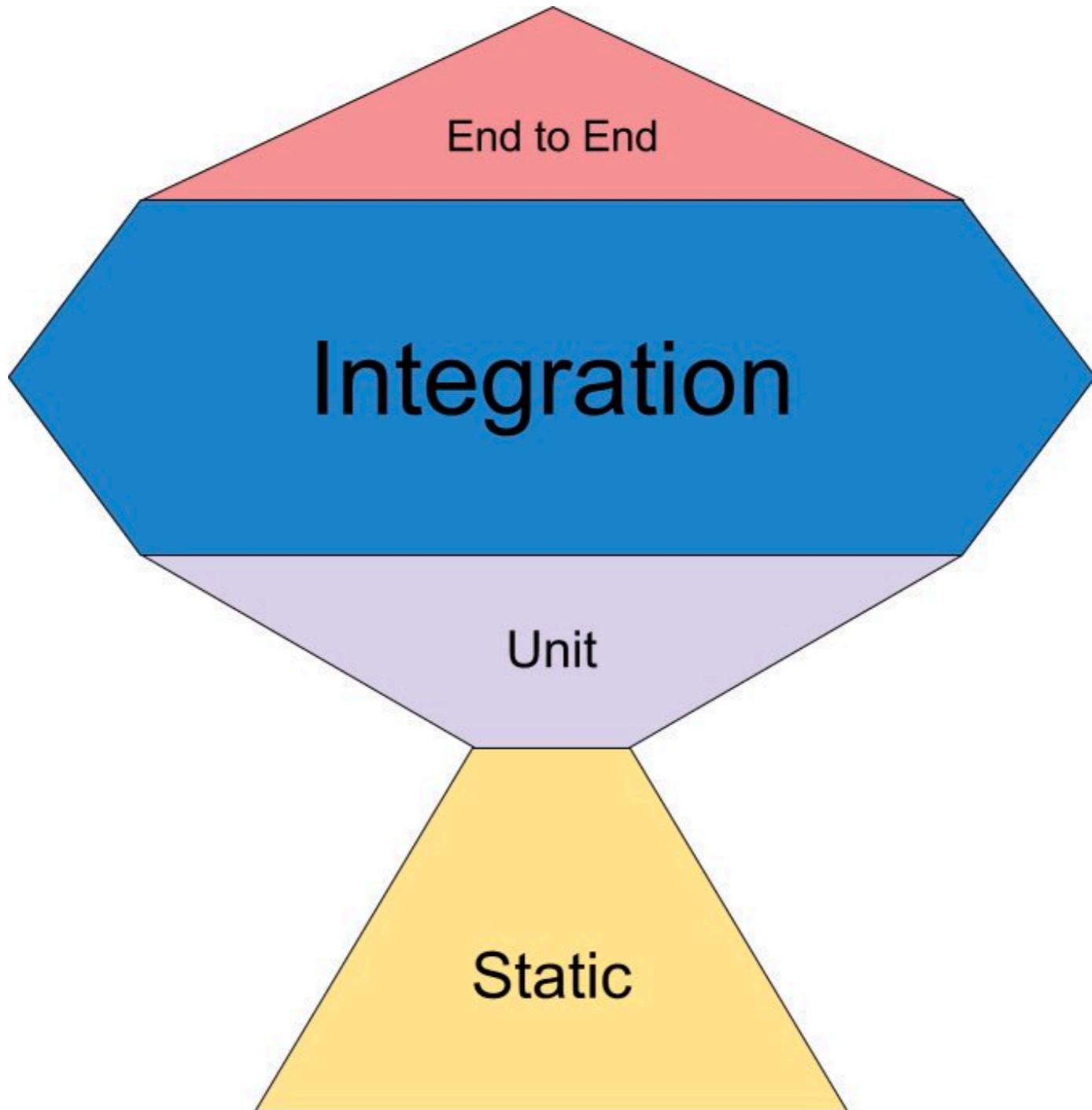




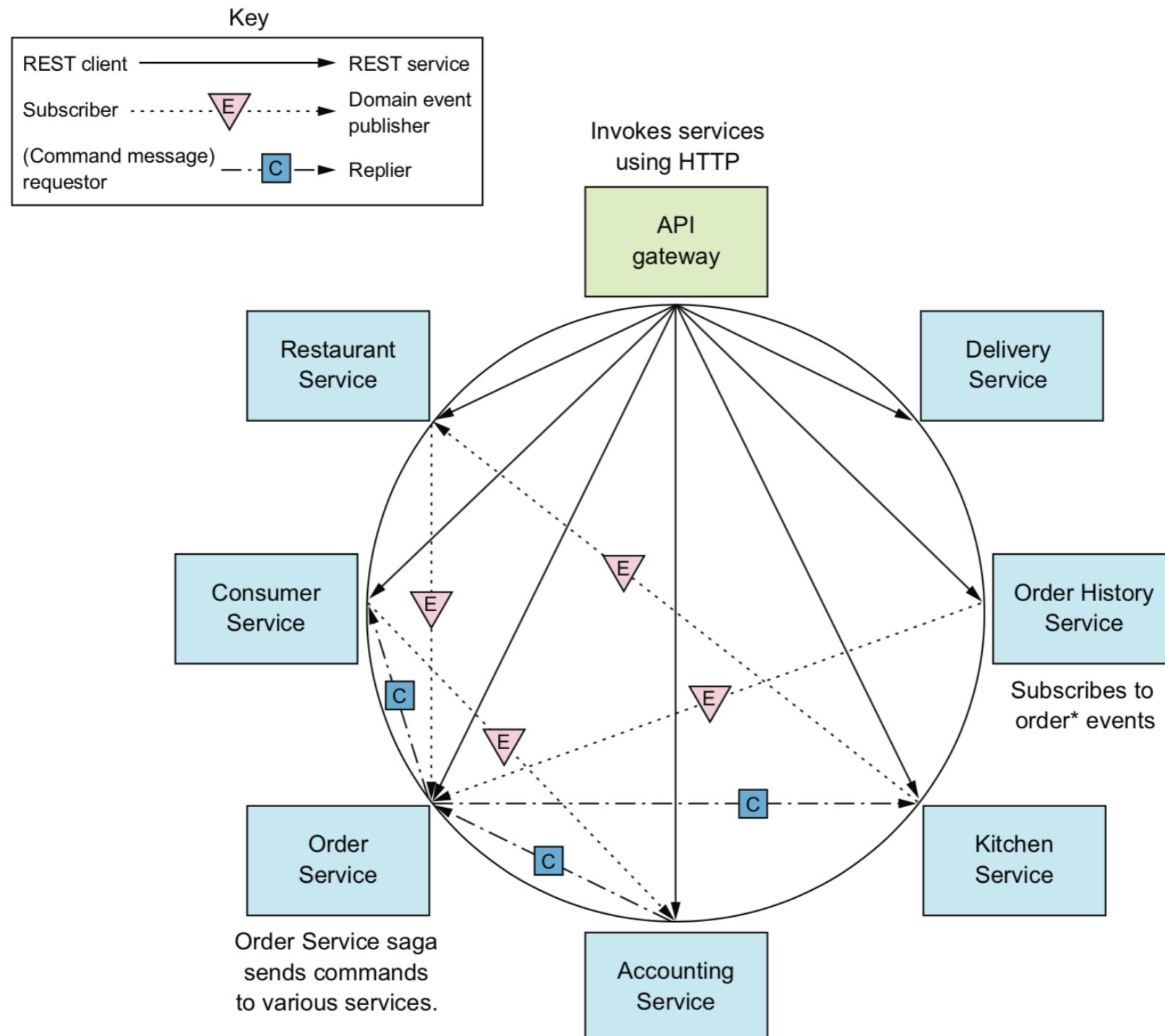
Cupcake testing



Trophy testing

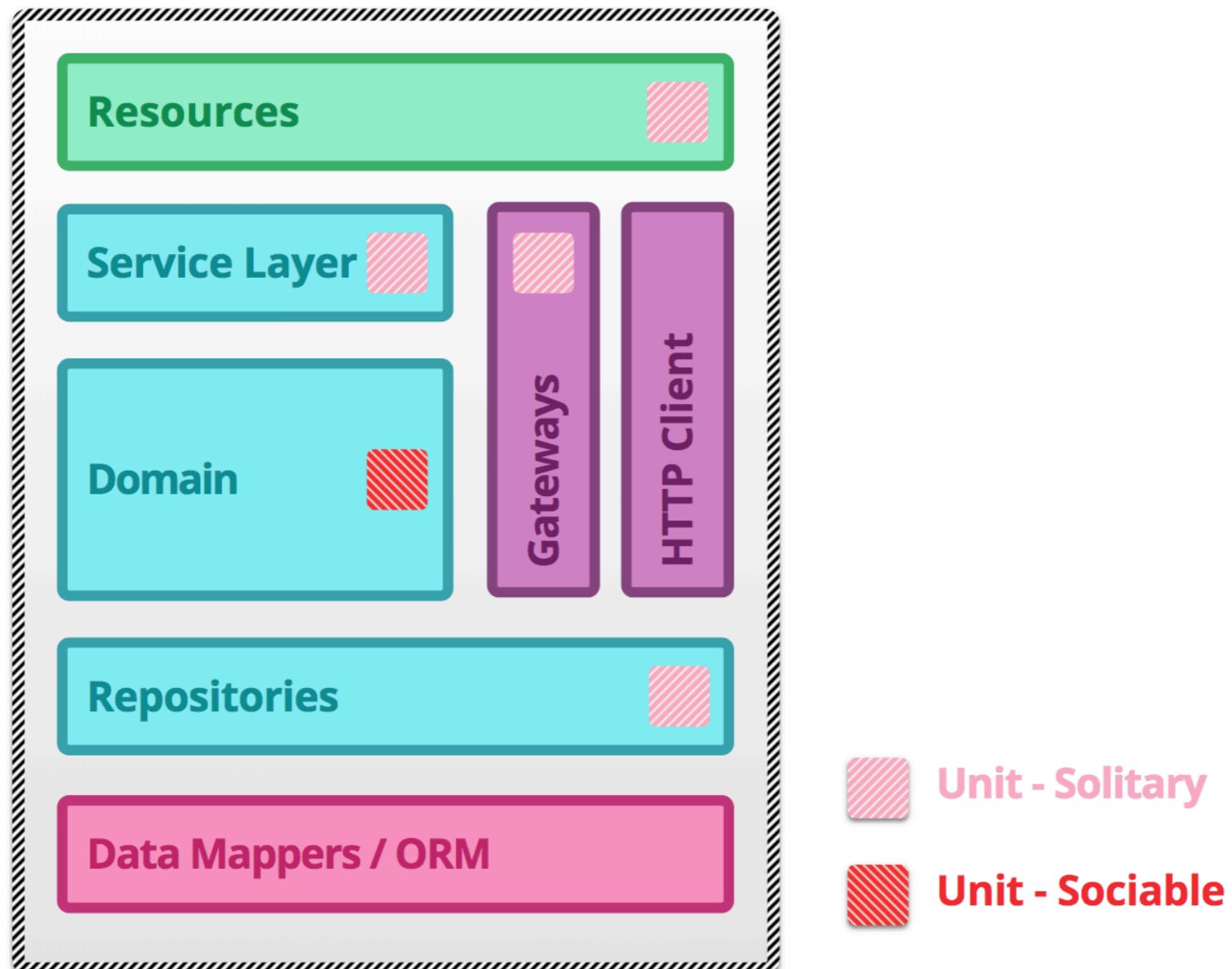


Testing Microservices ?

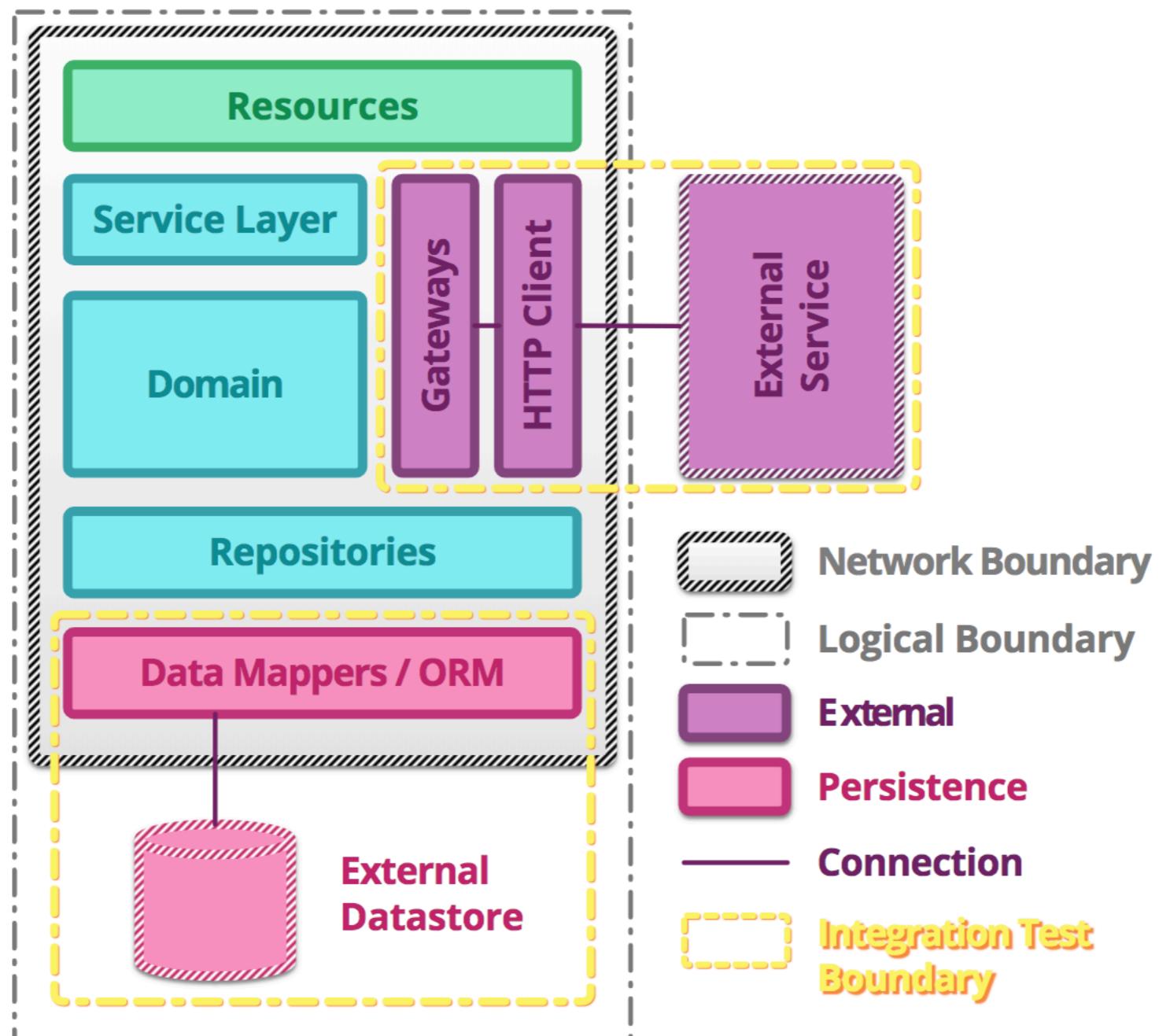




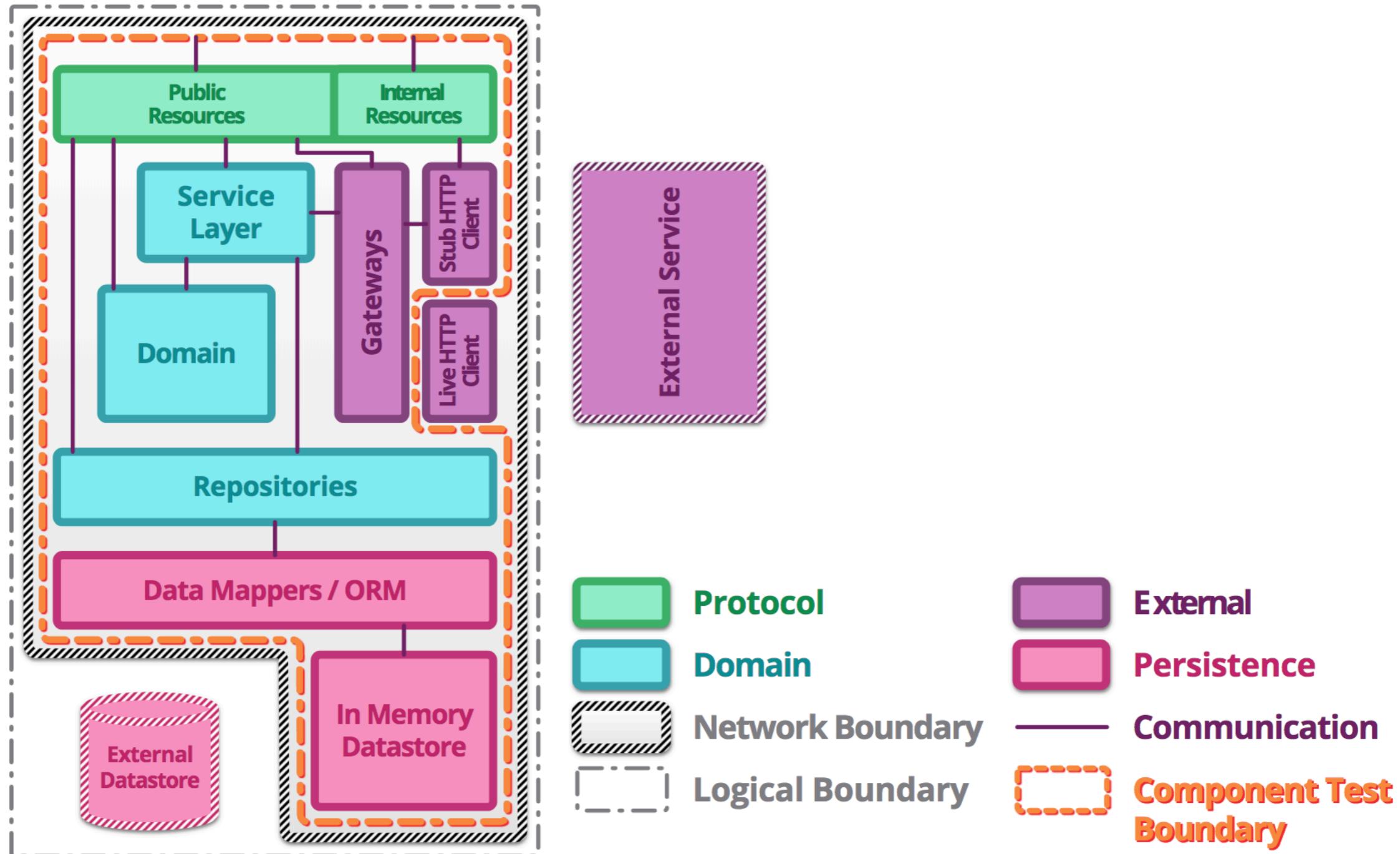
Unit testing



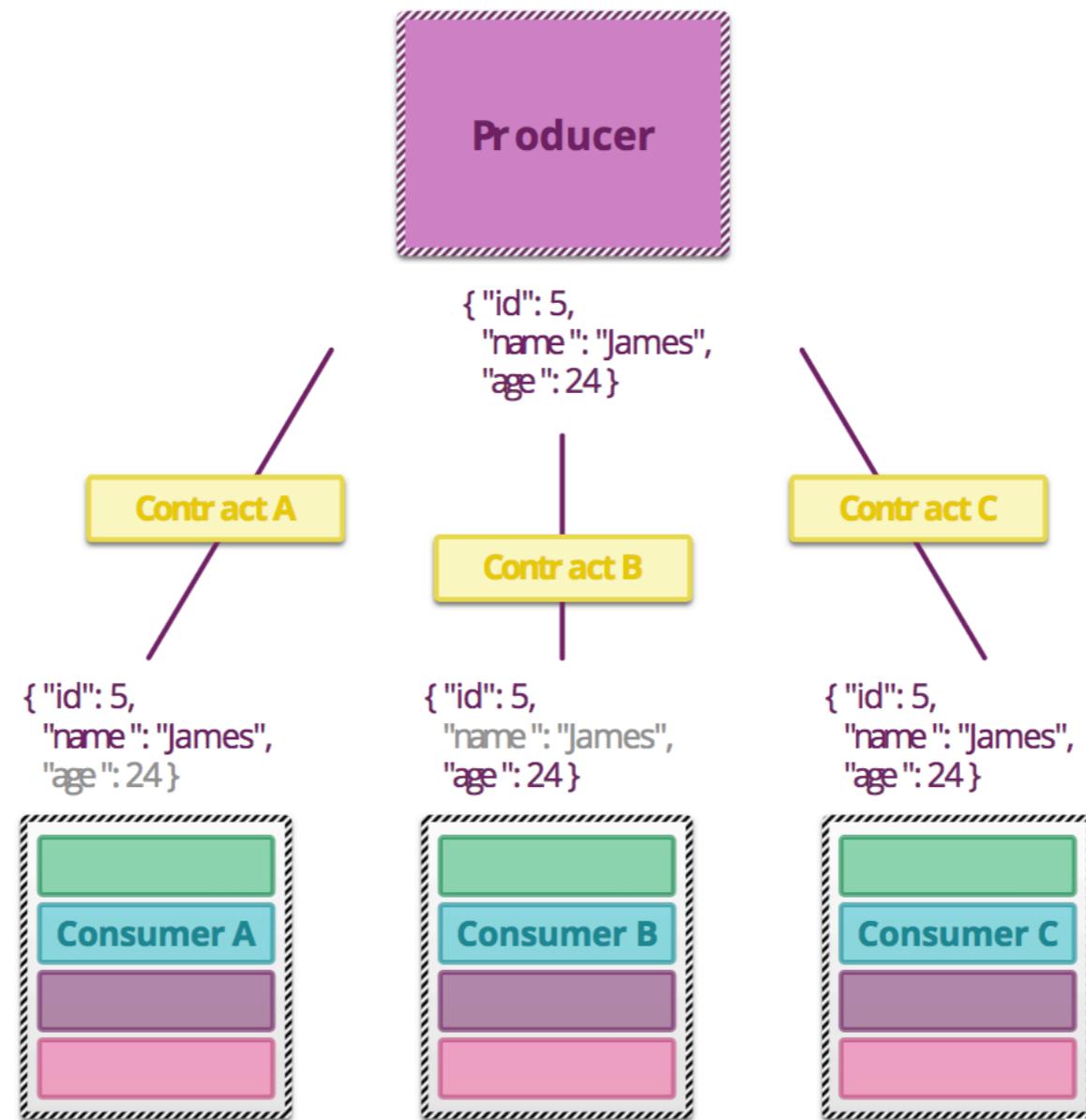
Integration testing



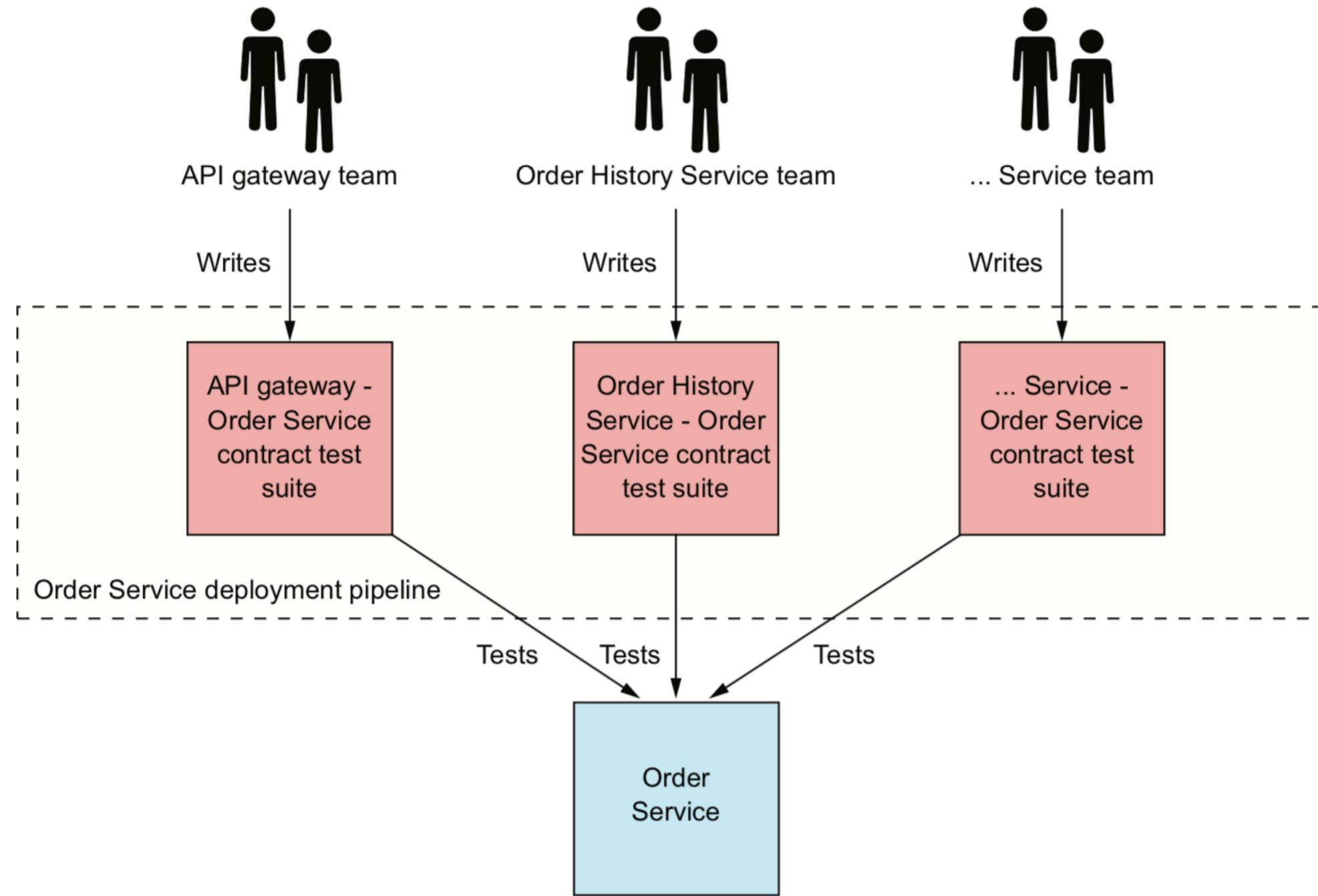
Component testing



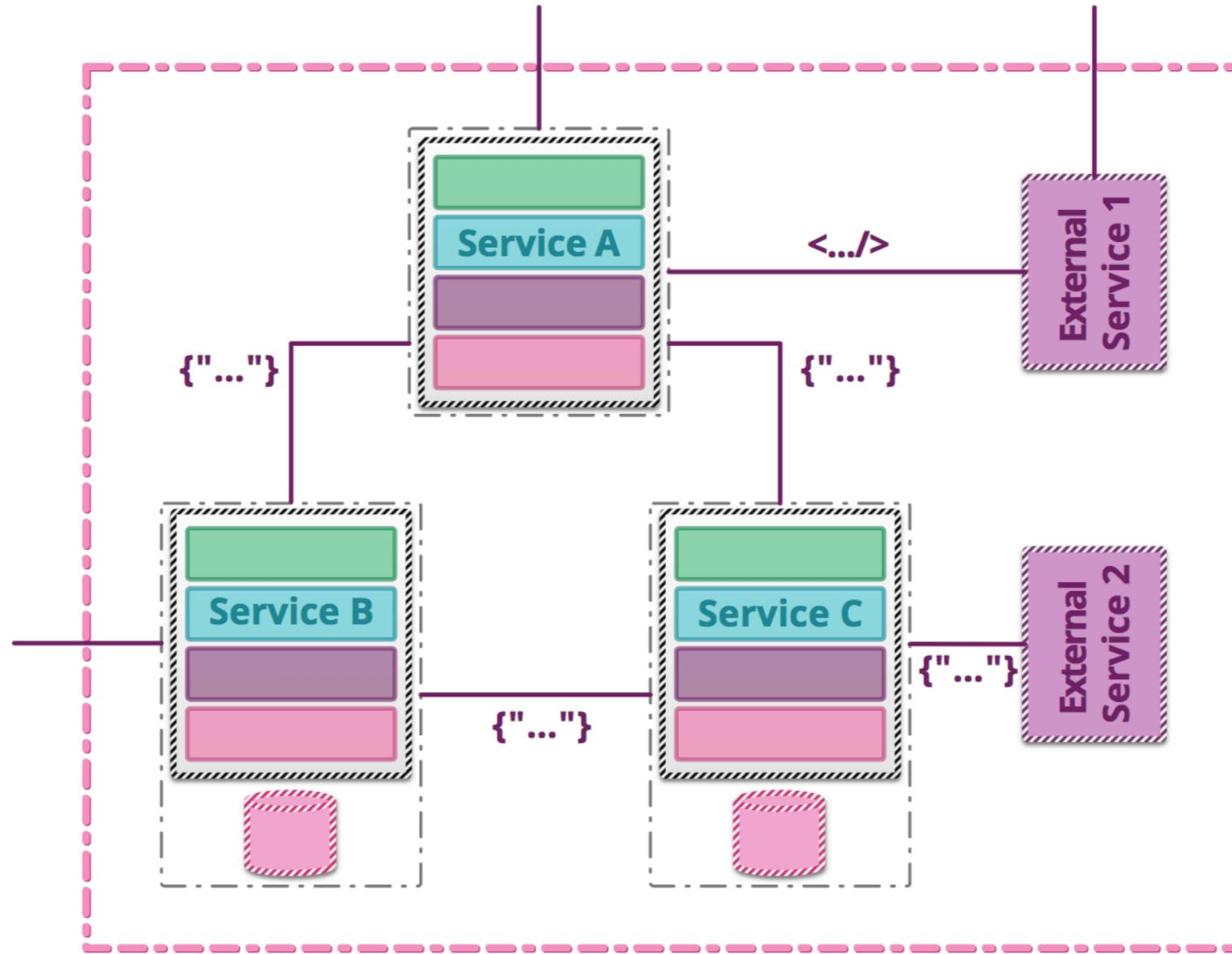
Contract testing



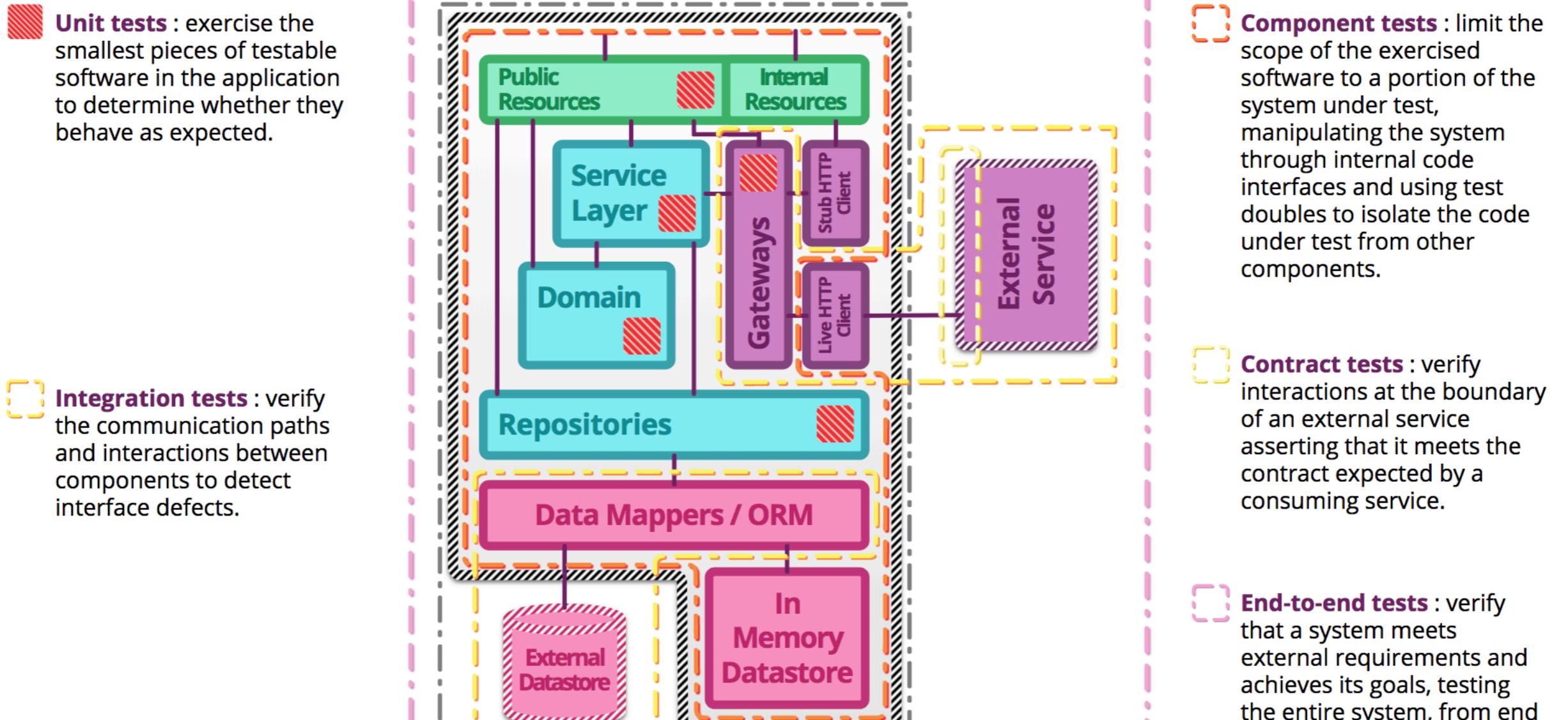
Consumer Contract testing



End-to-End testing



Summary



What is your testing strategy ?



Workshop



Microservices need automation



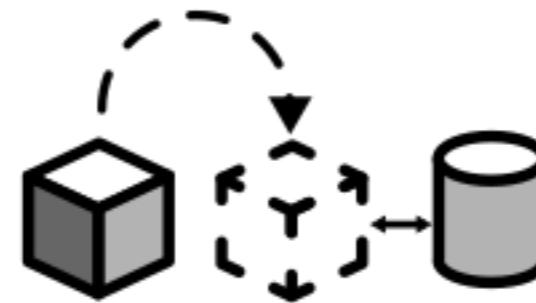
Build



Test



**Provide
Infrastructure**



Deploy



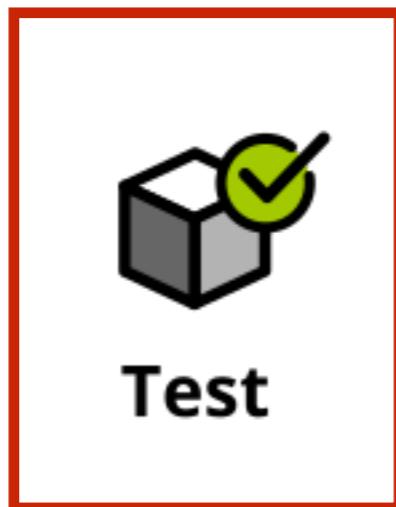
**Test
More**



Microservices need automation



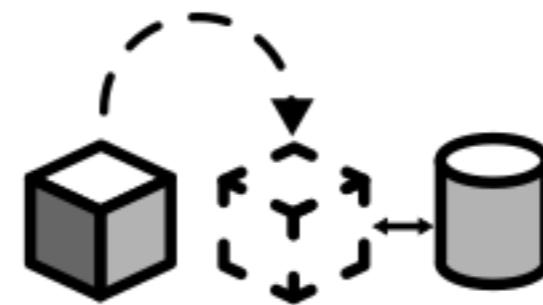
Build



Test



Provide
Infrastructure



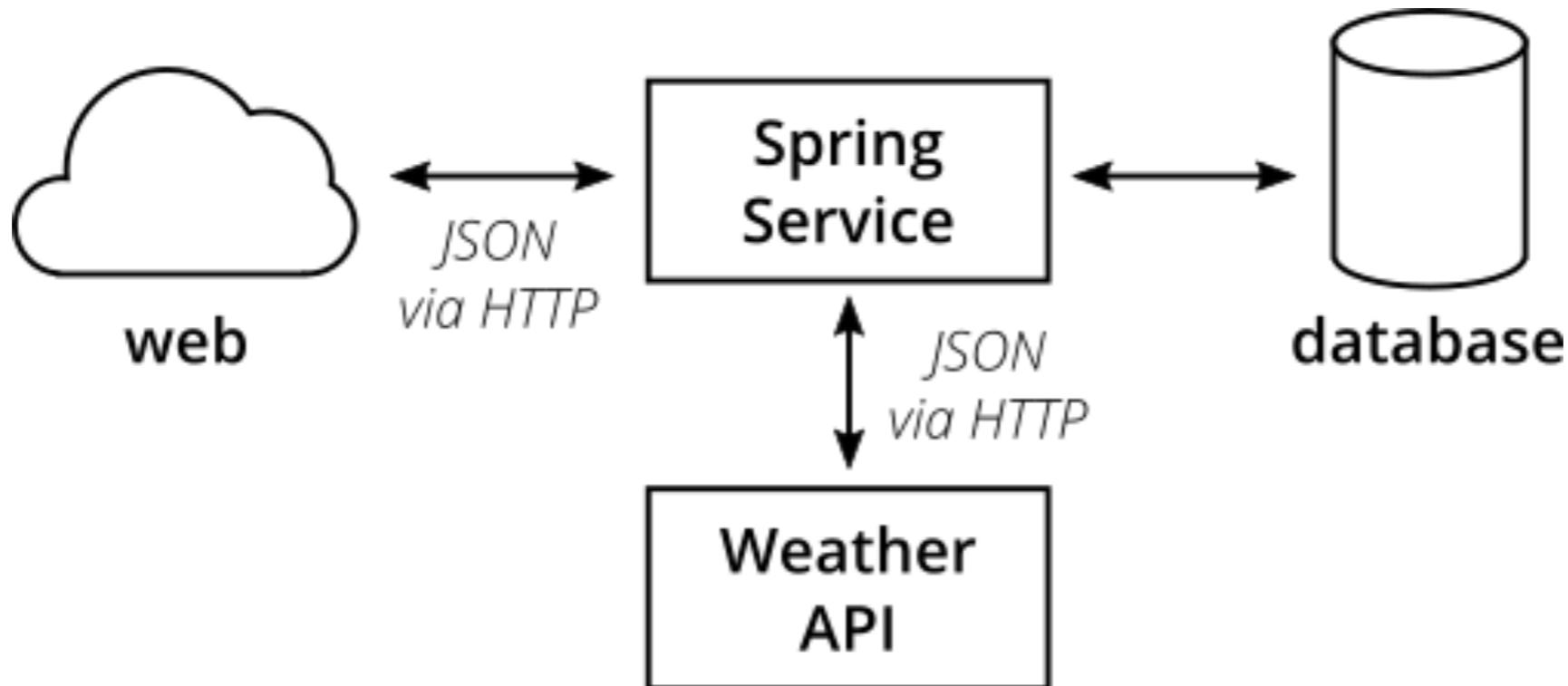
Deploy



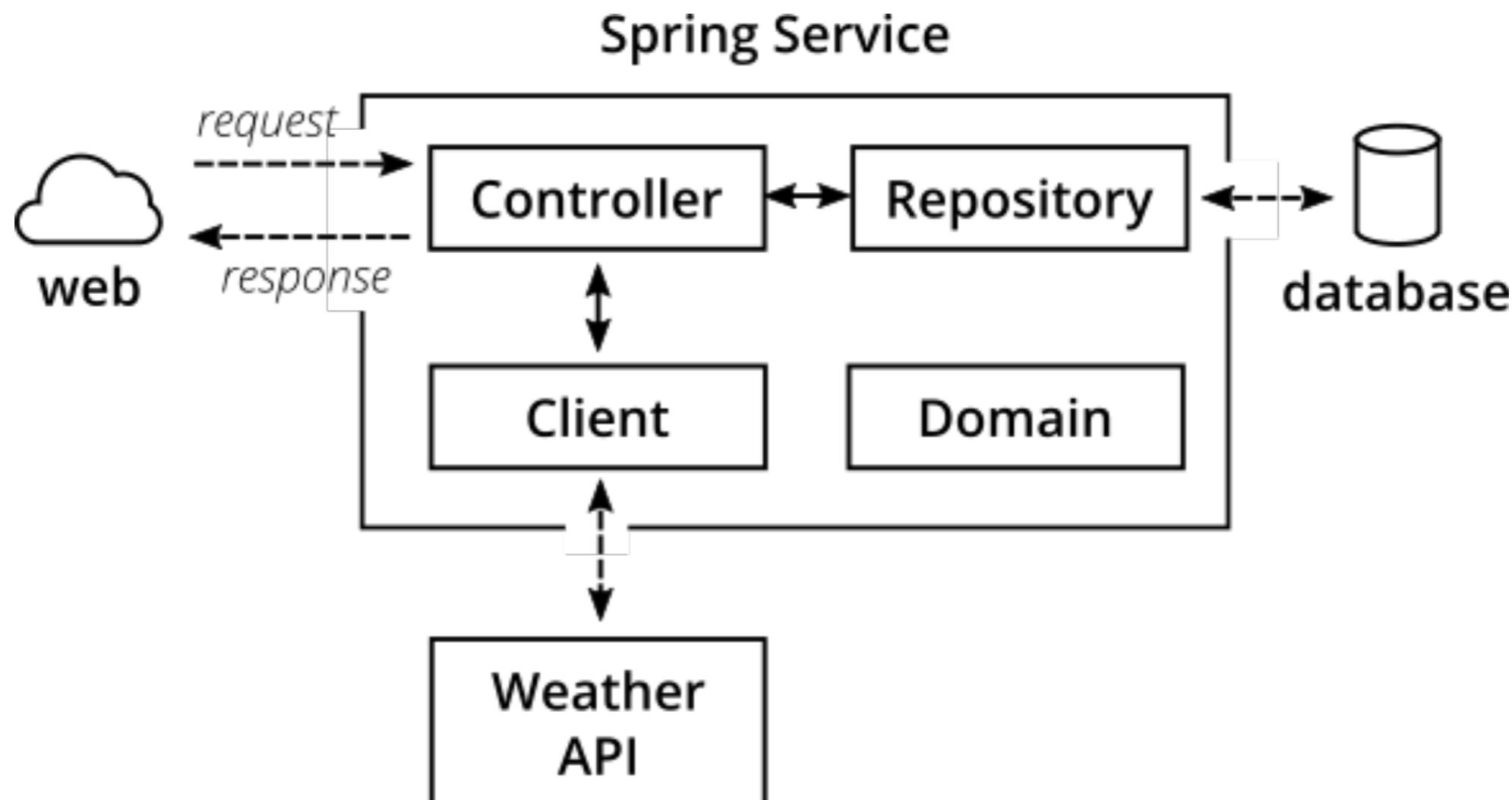
Test
More



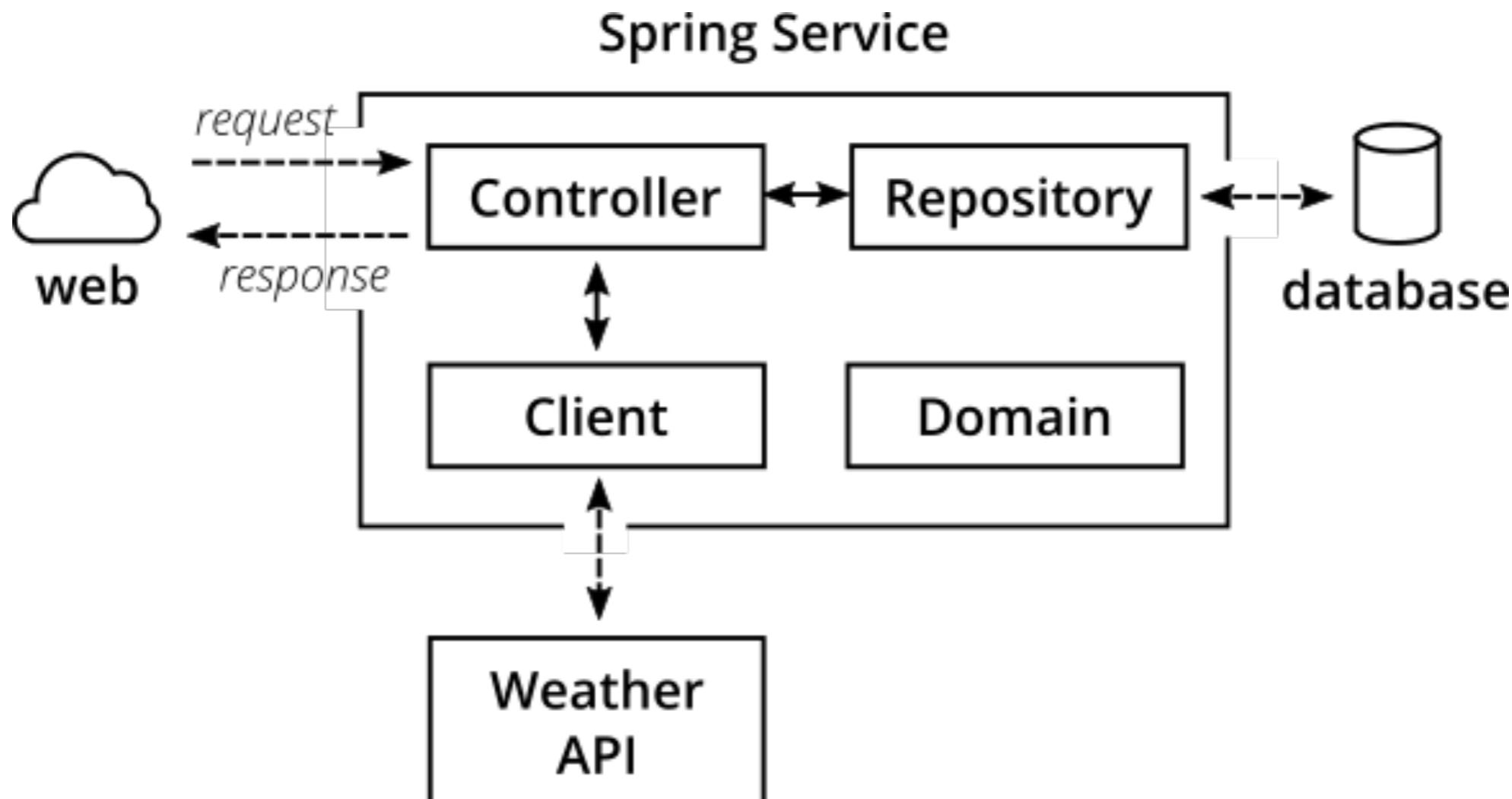
Sample Application



Internal architecture



Internal architecture



https://github.com/up1/demo-microservice-testing



Run all tests

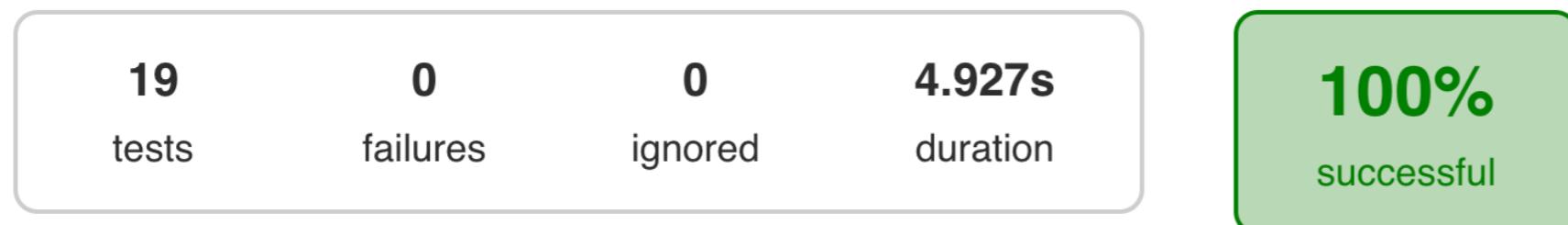
```
$gradlew test jacocoTestReport
```



Test report

/build/reports/tests/test/index.html

Test Summary



Packages

Classes

Package	Tests	Failures	Ignored	Duration	Success rate
example	12	0	0	4.050s	100%
example.person	1	0	0	0.285s	100%
example.weather	6	0	0	0.592s	100%



Code coverage report

/build/reports/coverage/index.html

 demo-microservice-testing

demo-microservice-testing

Element	Missed Instructions	Cov.	Missed Branches	Cov.	Missed	Cxty
 example.person		14%		0%	17	22
 example.weather		64%		33%	17	27
 example		92%		n/a	1	8
Total	200 of 406	50%	44 of 52	15%	35	57



Run sample project

```
$set WEATHER_API_KEY=<data>  
$gradlew bootRun
```



Endpoint of services

URL	Detail
<u>http://localhost:8080/hello</u>	Hello service
<u><a href="http://localhost:8080/hello/<lastname>">http://localhost:8080/hello/<lastname></u>	Get data from database
<u>http://localhost:8080/weather</u>	Get data from REST APIs



What to testing ?



What to testing ?

Unit testing

Integration testing

Component testing

Contract testing

End-to-End testing



How to testing ?



Depend on technology and tool



Working with Java

Testing type	Tools
Unit testing	JUnit, Mockito
Integration testing	Database, API mocking eg. WireMock, Stubby4J, MBtest
Component testing	JUnit, Mockito Database, API mocking
Contract testing	Pact, MBtest
End-to-End testing	UI testing and API testing eg. Selenium, Cypress, Postman



Workshop

