

Testing Microservices

See It, Feel It, Touch It, Heal It, Explore It

Testing Microservices

See It, Feel It, Touch It, Heal It, Explore It

Ken Pugh




Who are you?

- Testers, developers, others?
- Why are you here?

Objectives / Tests

- Understand a model for testing microservices
- Collaborate on creating microservice specifications with tests
- Create appropriate cross-functional tests for microservices
- Determine a strategy to test microservice interactions
- Explore ways to use logging
- Possibly apply model to your microservice

Ken Pugh



- ATDD/BDD, TDD, BVDD, Lean, Scrum, SAFe, Design Patterns
- Over 2/5 century of software development experience
- Co-author SAFe® Agile Software Engineering
- Author of seven books


atdd@kenpugh.com
<https://www.linkedin.com/in/kenpugh/>
<http://acceptancetestdrivendevelopment.com>




Lean Agile Acceptance Test-Driven Development: Better Software Through Collaboration



Prefactoring: Extreme Abstraction, Extreme Separation, Extreme Readability



Interface Oriented Design



Overall Rule

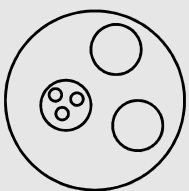
There are exceptions to every statement, except this one

Testing Microservices

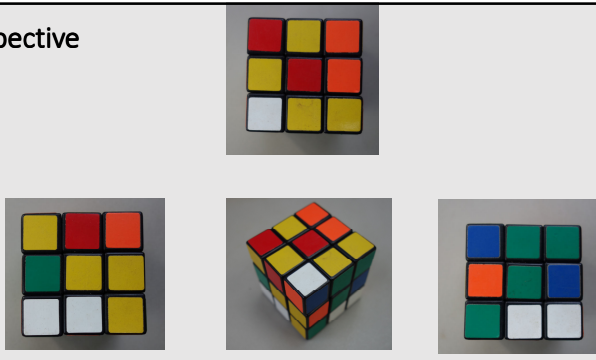
See It, Feel It, Touch It, Heal It, Explore It

2nd Overall Rule

Context is everything
Everything exists in a context
Everything is always true in some context



Perspective



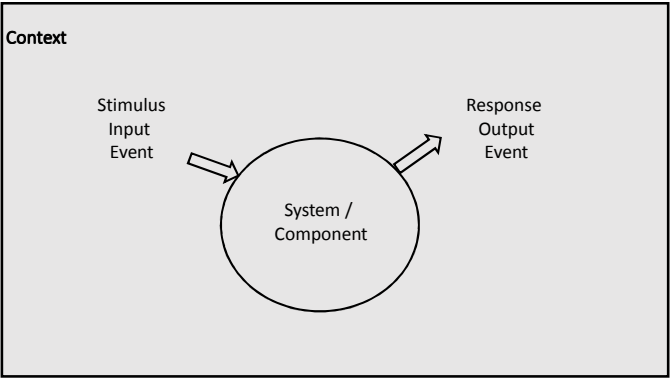
See The Big Picture

Context, External Behavior, Interactions
Monolith versus Microservices

Example Domain

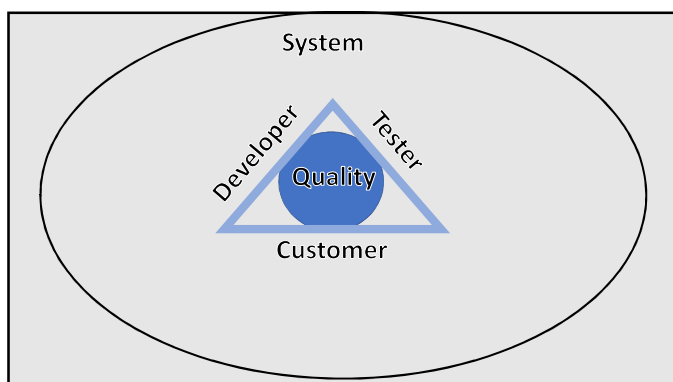
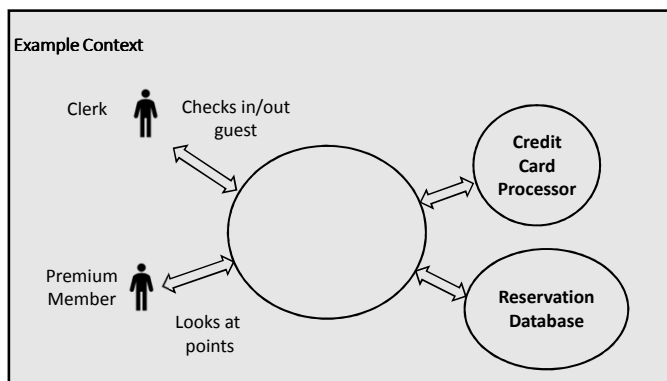
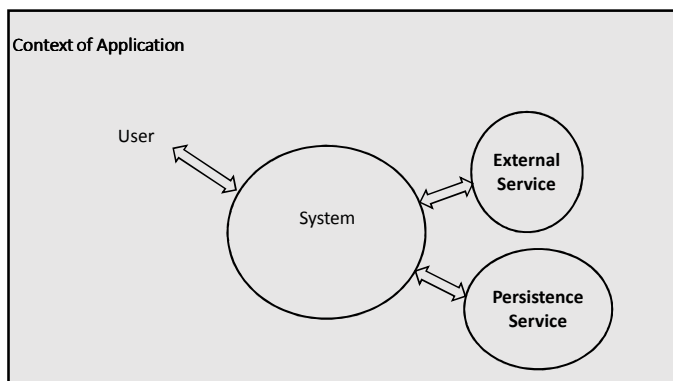
- Premium Member
- Nights – to achieve Premium Level
 - Staying in hotel
- Points – for Rewards
 - Achieve by staying in hotel
 - Using credit card
 - Doing promotion

Context



Testing Microservices

See It, Feel It, Touch It, Heal It, Explore It



Think Test-First

Collaboration helps automation

- Behavior
- | | |
|--|--|
| <ul style="list-style-type: none"> • Flow • Given
<i>State</i> • When
<i>Input Event</i> • Then
<i>Output Event</i>
<i>Change in State</i> | <ul style="list-style-type: none"> Stateless • Given
<i>Input</i> • When
<i>Process</i> • Then
<i>Output</i> |
|--|--|

- External Behavior (1)
- Given a renter occupies a hotel room
 - When the renter checks out
 - Then a bill is produced with all charges including taxes
 - And a transaction is issued to the credit card company
 - And the room is not occupied

Testing Microservices

See It, Feel It, Touch It, Heal It, Explore It

External Behavior (2)

- Given a renter who is a premium member occupies a hotel room
- When the renter checks out
- Then the renter is given night credits for each night
- And points for the total charges
- Given a premium member is logged on
- When they request their status history
- Then qualifying activity is displayed for every category

Monolith / Microservices

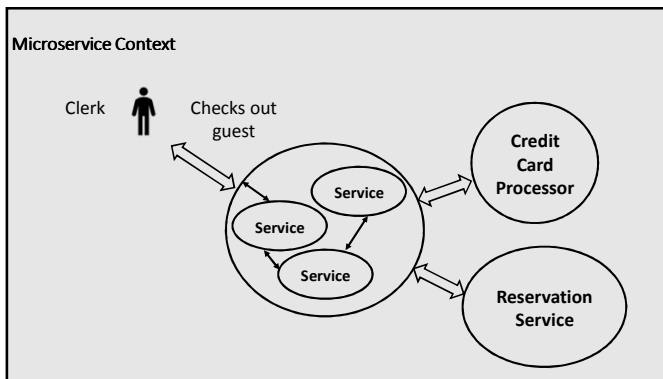
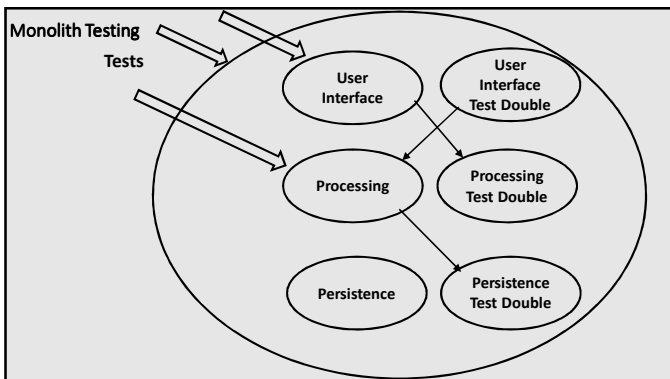
Microservice Attributes

- Self-contained business functionality
- Clear interfaces
- Independently deployable
- Independently scalable (benefit)
- A micro application

Layer/Function	Reservation	Premium Member	Payment
User Interface			
Processing (mid-tier)			
Persistence (database)			

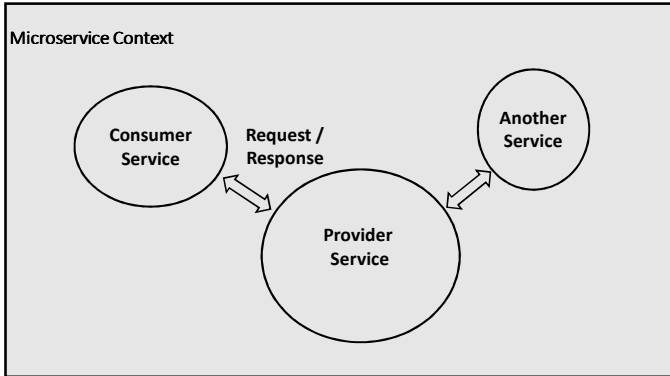
Layer/Function	Reservation	Premium Member	Payment
User Interface			
Processing (mid-tier)			
Persistence (database)			

May not have UI

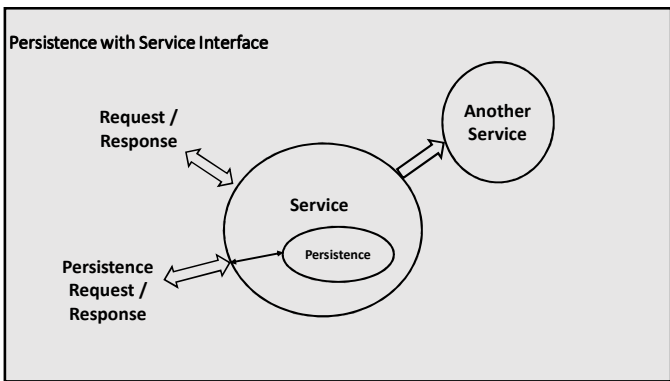
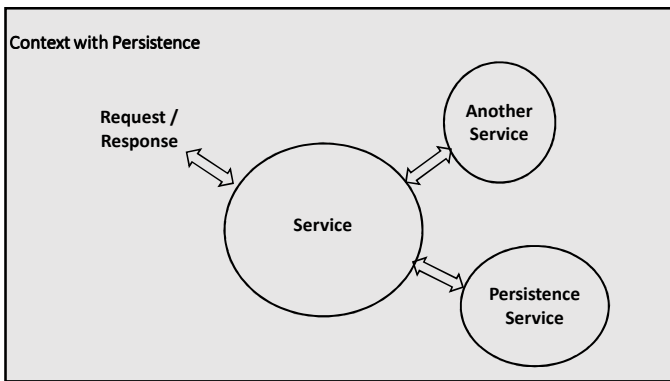
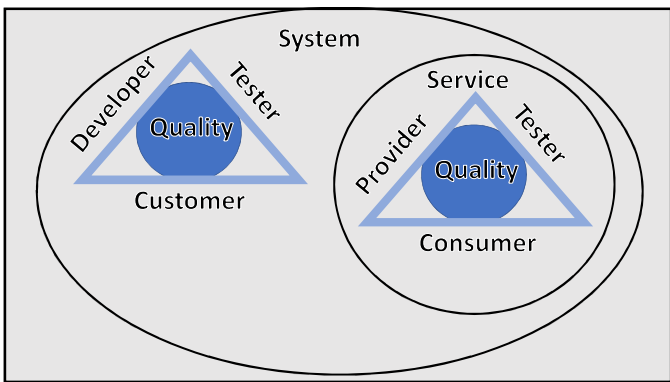
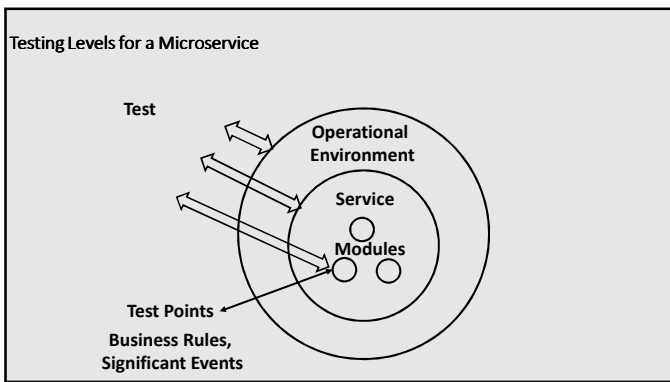


Testing Microservices

See It, Feel It, Touch It, Heal It, Explore It

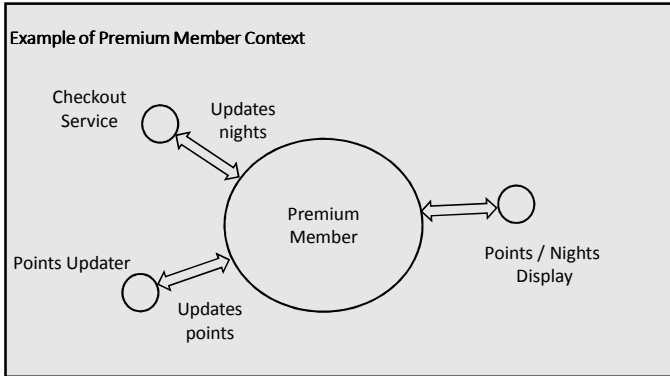


- Terms**
- **Consumer of a service**
 - Sender of request
 - Receiver of response
 - **Provider of service**
 - Receiver of request
 - Sender of response
 - **Producer of event (to be discussed later)**
 - Publishes, Talks
 - **Receiver of event**
 - Subscribes, Listens



Testing Microservices

See It, Feel It, Touch It, Heal It, Explore It



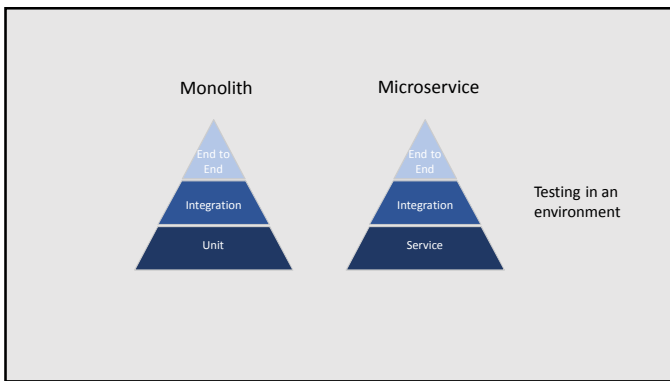
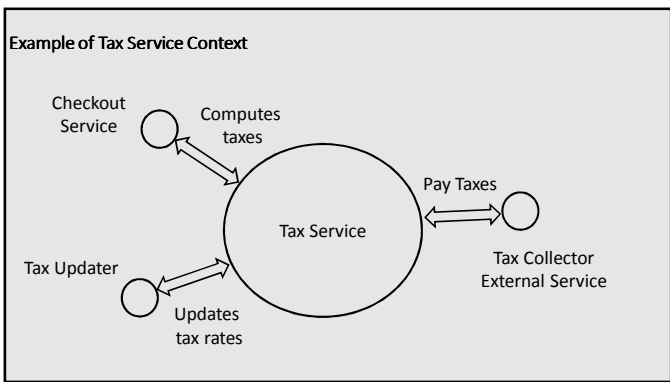
Example Context

- **Hotel**
 - Reserve a room
 - Compute taxes
 - May use premium points
- **Check in**
- **Check out**
 - Compute taxes
 - Record nights and points
 - May record use of points


09Jun19	Room Charge	155.00
09Jun19	State Occupancy Tax	9.30
09Jun19	City Tax	10.85
09Jun19	County Tax	6.20

Exercise

- Draw a context diagram for a microservice you are responsible for testing
- Or
- Draw a context diagram for a hotel tax service
 - Computes taxes for hotel stay



Interfaces



**DESIGN TO
INTERFACES,
NOT
IMPLEMENTATIONS**



**TEST TO
INTERFACES,
NOT
IMPLEMENTATIONS**

- ### Three Laws of Interfaces
1. An implementation of an interface shall do what the interface defines (See It)
 2. An implementation shall do no harm (use excessive memory, hold locks, etc.) (Feel It)
 3. If an implementation is unable to perform its responsibilities, it shall notify someone (Heal It)

See It
Visible tests so everyone understands the behavior

Model

Information Model

Member ID	Date Of Stay	Number Nights
M123	1/1/2019	1

MemberID, DateOfStay, NumberNights,
M123, 1/1/2019, 1

```
<MemberID> M123 </MemberID>  
<DateOfStay> 1/1/2019 </DateOfStay>  
<NumberNights> 1 </NumberNights>
```

```
{  
  "MemberID" : "M123",  
  "DateOfStay" : "1/1/2019",  
  "NumberNights" : 1  
}  
{  
  "member_id" : "M123",  
  "date_of_stay" : "2019-01-01",  
  "number_nights" : 1  
}
```

Information Model (2)

Member ID	Date Of Stay	Number Nights
M123	1/1/2019	1
M123	2/2/2019	2

```
[
  {
    "member_id" : "M123",
    "date_of_stay" : "2019-01-01",
    "number_nights" : 1
  },
  {
    "member_id" : "M123",
    "date_of_stay" : "2019-02-02",
    "number_nights" : 2
  }
]
```

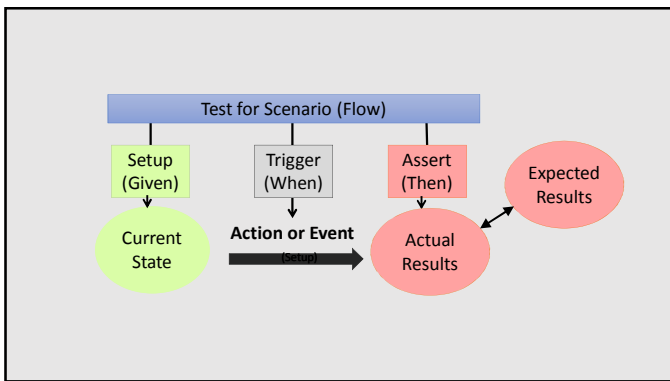
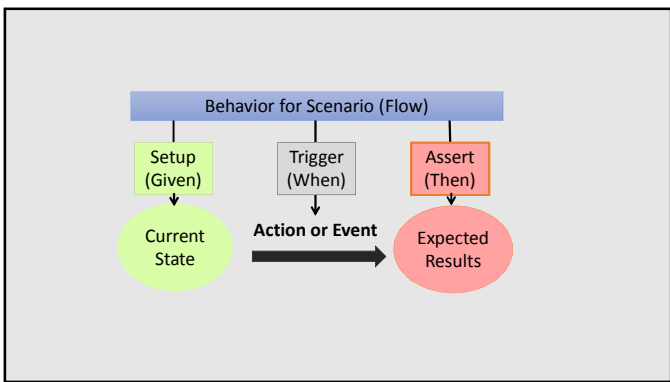
Information Model (3)

Member ID	Date Of Stay	Number Nights
M123	1/1/2019	1

Member ID	M123
Date Of Stay	1/1/2019
Number Nights	1

Member ID	Date Of Stay	Number Nights
M123	1/1/2019	1

Behavior



Example of Behavior

- Member status updated on checkout
- Given Member status is

Member ID	Number of Nights Current Year	Point Total
M123	11	100,000

- When checkout occurs

Date	Number of Nights	Total
7/1/2019	1	\$100.00

- Then Member status is

Member ID	Number of Nights Current Year	Point Total
M123	12	101,000

Business Rule
10 points for every dollar spent

Exercise

- Write up a scenario/test for a microservice you are responsible for testing
- Or
- Write up a scenario/test for a hotel tax service

09Jun19	Room Charge	155.00
09Jun19	State Occupancy Tax	9.30
09Jun19	City Tax	10.85
09Jun19	County Tax	6.20

Example of Taxes

- Given Tax Rates are

Zip Code	State Occupancy Tax Rate	County Occupancy Tax Rate	City Occupancy Tax Rate
27701	.07	.02	.01

- When taxes are computed

Zip Code	Room Charge
27701	100.00

- Then taxes are

State Occupancy Tax	County Occupancy Tax	City Occupancy Tax
7.00	2.00	1.00

Smaller Scenarios

- Can break down scenarios into smaller scenarios

Breaking Down (1)

- Given Member status is

Member ID	Number of Nights Current Year	Point Total
M123	11	100,000

- When checkout occurs

Date	Number of Nights	Total
7/1/2019	1	\$100.00

- Then Point Transaction is created

Member ID	Points	Date	Type
M123	1000	7/1/2019	Hotel Stay

Breaking Down (2)

- Add Point Transaction
- Given current Point Total is

Member ID	Point Total
M123	100,000

- When Point Transaction is received

Member ID	Points	Date	Type
M123	1000	7/1/2019	Hotel Stay

- Then Point Total is

Member ID	Point Total
M123	101,000

Another Example of Behavior

- Current member status displayed
- Given Member status is

Member ID	Number of Nights Current Year	Point Total
M123	12	101,000

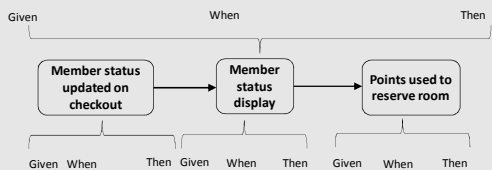
- When Member requests display

Member ID
M123

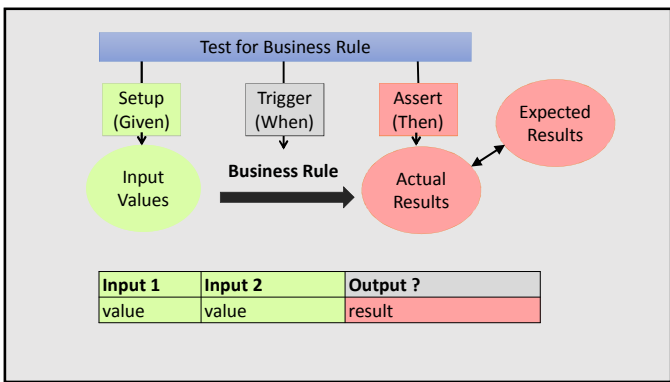
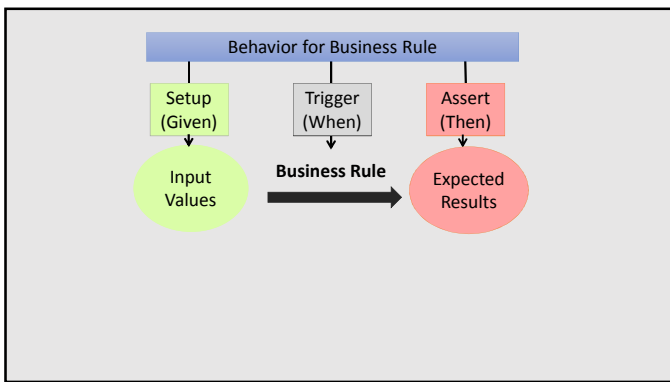
- Then display shows

Number of Nights Current Year	Point Total
12	101,000

Sequence / Workflow



Business Rules



Example of Business Rule

• Member Premium Level Business Rule

Number of Nights Current Year	Premium Level
10	Silver
25	Gold
50	Platinum
365	Residents

• Tests of Business Rule

Number of Nights Current Year	Premium Level ?
9	Unranked
10	Silver
24	Silver
25	Gold
Etc.	Etc.

Test complex business rules directly

Interface (Domain) Terms

Category Values For Points - Enumeration

Hotel Stay
Credit Card
Yearly Bonus
Redemption
Redemption Cancellation

Points – Rules and Tests

Minimum	0
Maximum	200000
Type	Integer

Value	Valid?
-1	No
0	Yes
200000	Yes
200001	No
1.23	No

Invalid Values should produce error or default result

Testing Microservices

See It, Feel It, Touch It, Heal It, Explore It

Exercise

- Is there a business rule or a interface domain term in your microservice?
- If so, write up a behavior for it

Possible Answers

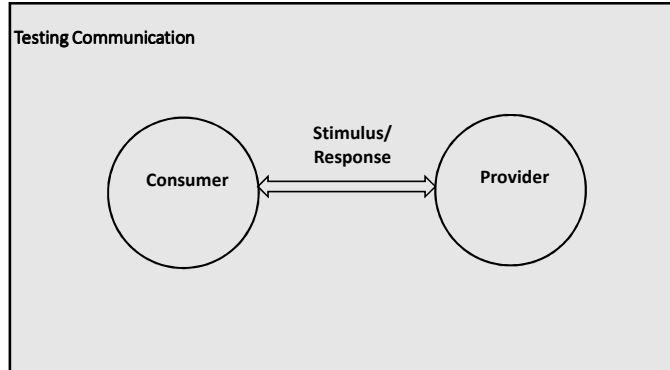
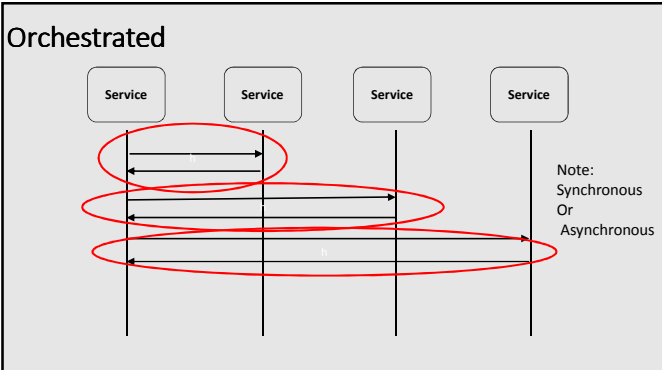
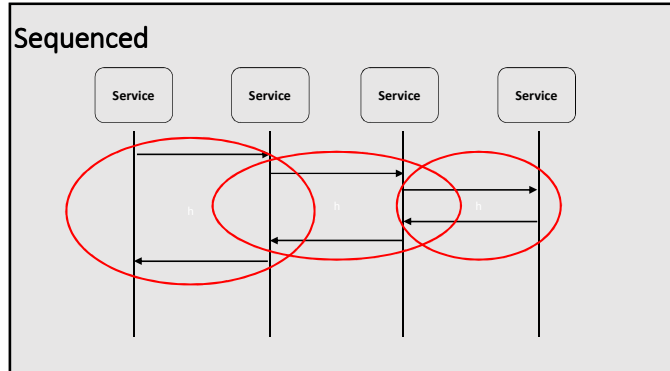
Round off

Tax Rate	Room Charge	Tax
.07	100.00	7.00
.07	100.07	7.00
.07	100.08	7.01

Zip Code	Valid?
27701	Yes
X5734	No

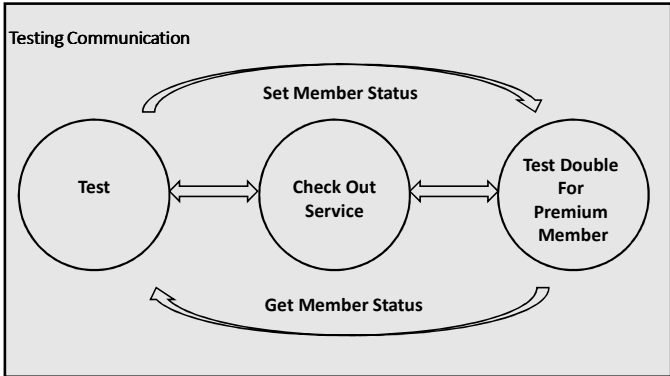
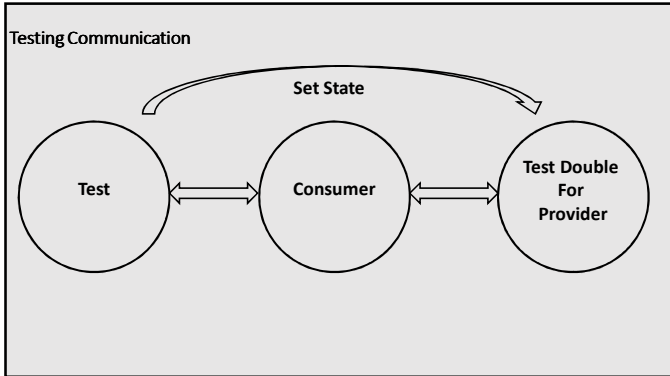
Tax Rates	Valid
-.01	No
.00	Yes
.30	No

Interactions

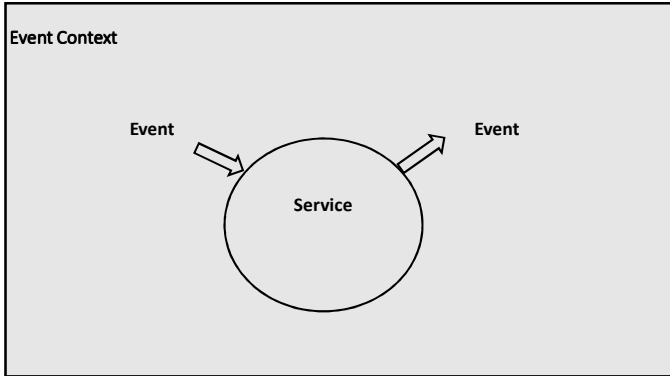
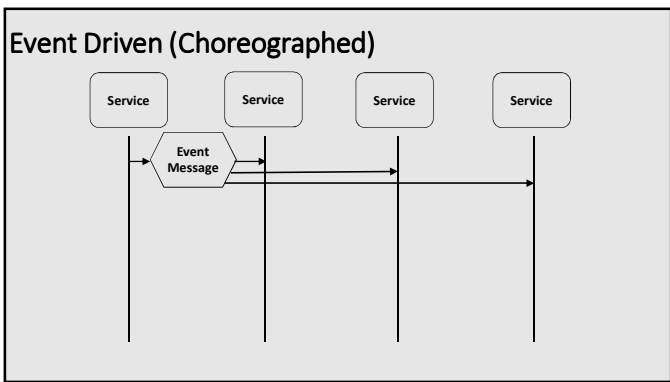


Testing Microservices

See It, Feel It, Touch It, Heal It, Explore It



Events



Event Examples

- Premium Members
 - Add points
 - On checkout
 - Use points
 - On checkout if stay is on points
- Event
 - Checkout Event

Member ID	M123
Date	7/1/2019
Number of Nights	1
Total	\$0
Points Used	20,000

Testing Microservices

See It, Feel It, Touch It, Heal It, Explore It

Event Example (2)

- But what other events should Premium Member subscribe to?
- Reserve use of points

Reservation ID	R123
Points Reserved	20,000

- Cancel use of points

Reservation IDs	R123
Points Reserved	20,000

Example of Event Stream

- Given Member status is

Member ID	Number of Nights Current Year
M123	0

- When events occur

Date	Number of Nights
2/1/2019	5
3/1/2019	1
4/1/2019	5

- Then Member status is

Member ID	Number of Nights Current Year
M123	11

Another Example of Event Stream

- Given Point Total is

Member ID	Point Total
M123	100,000

- When events occur

Date	Type	Amount
1/2/2019	Credit Card Purchases	1000
3/1/2019	Room	500
4/1/2019	Redemption	20,000

- Then Point Total is

Member ID	Point Total
M123	81,500

Question

- How do the microservices in your context collaborate?

Feel It

Behavior for cross-functional aspects of microservices documented in tests

Agile Testing Matrix

		Business Facing			
Guide Development	Functional Tests	Usability	Workflow	Critique Product	
	Service Tests	Exploratory			
	Unit Tests	Performance	Security		Quality Attributes
		Technology Facing			

Adapted from Janet Gregory Lisa Crispin

Feel It

- **Quality attributes**
 - AKA Cross-functional requirements
 - AKA Non-functional requirements
- What quality attributes do you currently test in your systems?
- **Service Level Agreement (SLA)**

Service Level Agreement

Total latency	5 seconds
Throughput	50000 requests/second
Processing time	20 microseconds

Resource Agreement

Memory usage	< 100 MB
MOP per call	1.5

Testing Environment

- Environment (framework / libraries / etc.) can provide aspects of quality attributes
 - E.g. Increase instances
- Are you testing the environment, the configuration, or the service itself?

- **Idempotency**
 - Same outcome if
 - Request received multiple times
 - Response received multiple times
- **One way is to add ID**
 - Event ID
 - Transaction ID
 - Correlation ID

Example of Idempotent Test

- **Given Point Total is**

Member ID	Point Total
M123	100,000

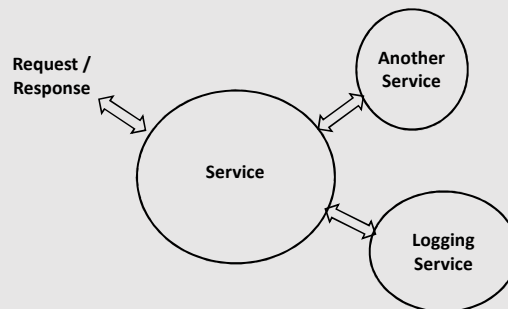
- **When duplicate transactions occur**

Date	Type	Amount	Transaction ID
1/2/2019	Credit Card Purchases	1000	T456
1/2/2019	Credit Card Purchases	1000	T456
1/2/2019	Credit Card Purchases	1000	T456

- **Then Point Total is**

Member ID	Point Total
M123	101,000

Logging



Logging

- Log events with:
 - Timestamp
 - ID (Event, Correlation, etc.)
 - Source
 - Service
 - Operational location
- Log entries include:
 - Requests / responses
 - Significant internal events (test points)

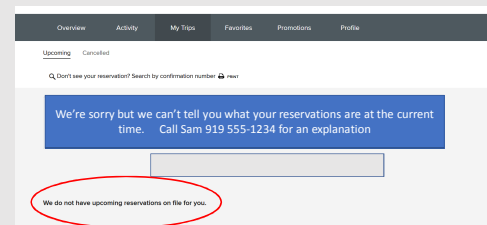
Discussion

- What else might be on an SLA?

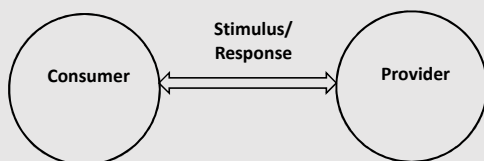
Touch It

Test for interactions of microservices include injection or simulation of faults

What's Going On?



Testing Communication



Injecting Faults

- Test Double could
 - Return unavailable
 - Return after a long time
 - Return with dependency issue
 - Return other potential errors

Exercise

- Would the microservice you are testing require test doubles or should it have a test double?
- Or
- What should a test double for taxes do?

Example of Taxes

- Return tax calculated at fixed rate, regardless of jurisdiction

Heal It

Checking of logs of microservice interactions can be used to determine failure causes

Recording the Symptoms

- **Service Logging**
 - Record request/responses of services
 - Particularly error responses
 - Record significant events (test points) inside a service
- **Analyze logging for**
 - Service Level Agreement
 - What is "normal"
 - Error rates
 - Performance
 - Look for abnormal
 - What is significantly different

"Testing in Production"

- **Synthetic (AKA Fake) transactions or synthetic users**
 - Perform operations to test workflows in production
 - E.G. Reserve room and then unreserve it

Discussion

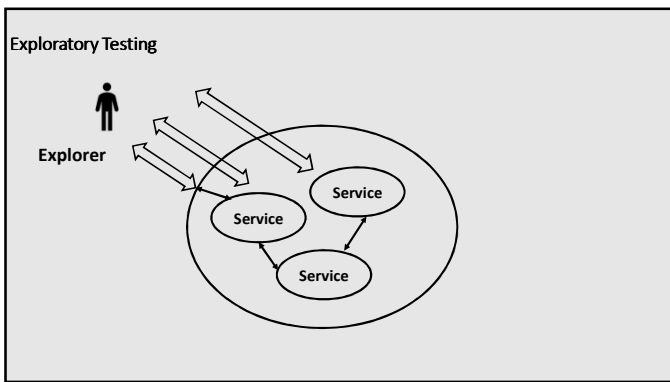
- Are there any significant events in your service that should be recorded?
- What level of errors is acceptable for your system?
- Can you use synthetic transactions?
- Testing is about tradeoffs - effort versus benefit
 - How much testing should you do on complete system?

Explore It

Testing going beyond described behaviors.

Exploratory Testing

- Usually done on application as a whole
- Could be done on individual service
- Could be done on multiple services



Exercise

- Anything you want to explore with taxes?

Objectives / Tests Review

- Understand a model for testing microservices
- Collaborate on creating microservice specifications with tests
- Create appropriate cross-functional tests for microservices
- Determine a strategy to test microservice interactions
- Explore ways to use logging
- See It
- Feel It
- Touch It
- Heal It
- Explore IT

Go Forth and See, Feel, Touch, Heal, Explore Your Microservices

Ken Pugh

atdd@kenpugh.com
<https://www.linkedin.com/in/kenpugh/>
<http://acceptancetestdrivendevelopment.com>