GRACE HOPPER CELEBRATION

ANITA B.ORG

API Specification Driven Testing

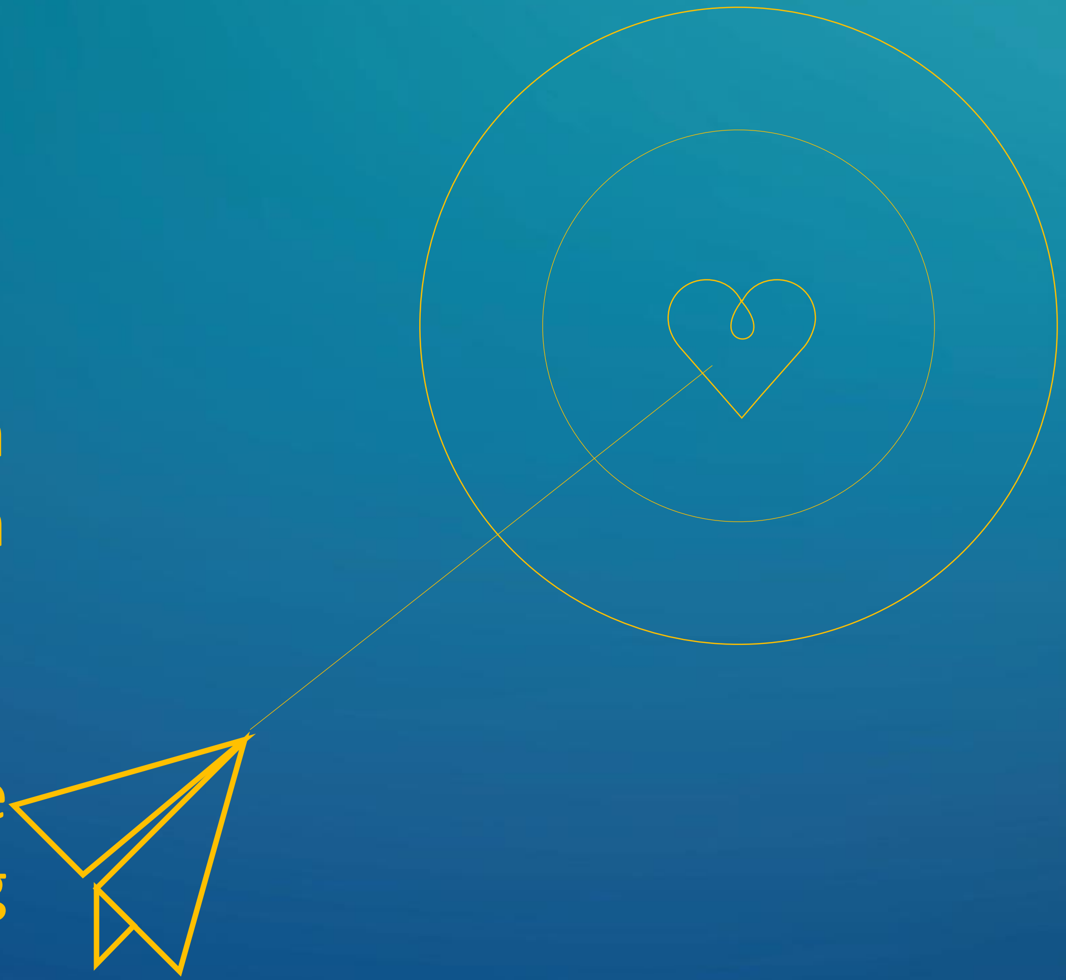
Enabling faster releases with higher Quality

about me

- Member of Technical Staff at PayPal
- 12+ years of experience in Software Development and Quality Assurance in distributed, highly scalable systems
- Strong passion for innovating new test methodologies and frameworks including my latest SmartTest framework (patent pending)

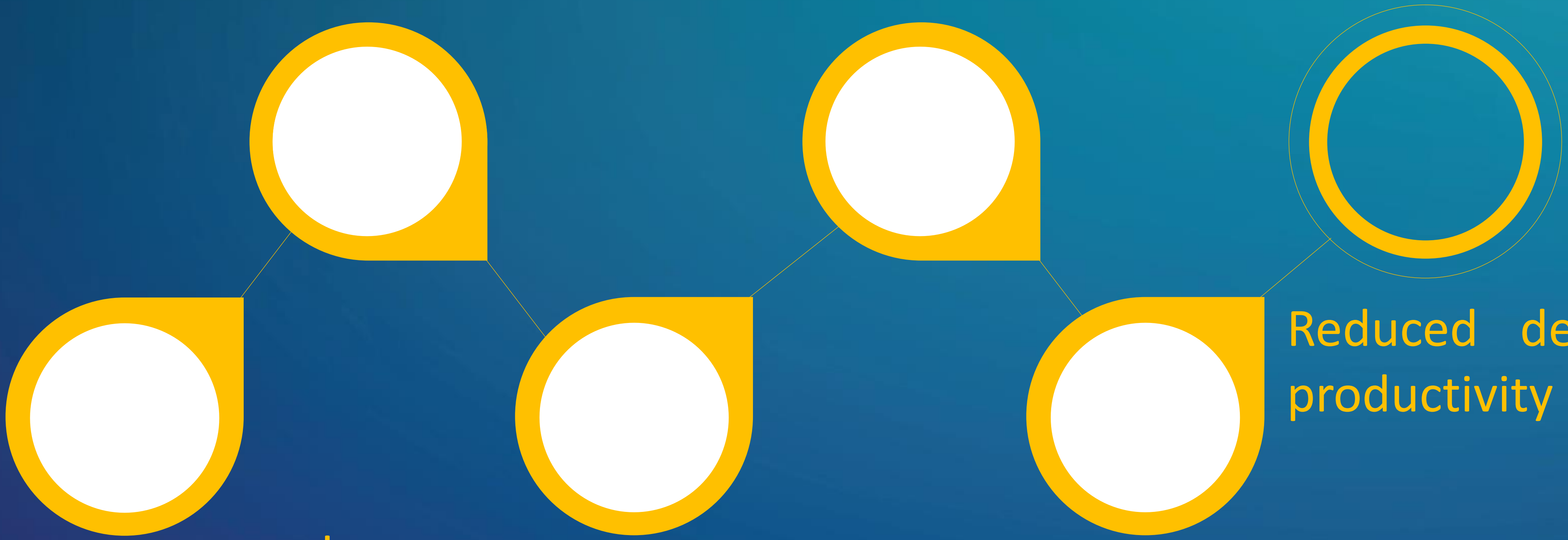
Take aways

- Overview of Behavioral Driven and Property based Testing
- Insight into API Specification Driven testing and how a framework has been developed using this new strategy
- How this test strategy helps to ensure high quality releases without spending a ton of time writing test code



Hmm.. a new strategy .
I wonder what problems does it help solve?

Do you face any of these problems



Test Duplication

Lack of visibility

Reduced developer productivity

Too many test data permutations

Scalability/Maintenance overhead

Tests not in sync with the specification

Behavioral Driven Testing(BDT)

BDT format helps to define scenarios in simple human readable format – Given, When, Then .

Scenario: Add an address to a user's profile
Given a US home address
When the address is added to the user profile
And the address is verified by ZOOT
Then the address is added successfully

Property-based Testing (PBT)

PBT helps to validate the system under test for a range of data points.

API Specification Driven Testing

Coupling the Behavioral Driven and Property based testing through a specification file unlocks a new test strategy

Behavioral Driven Testing

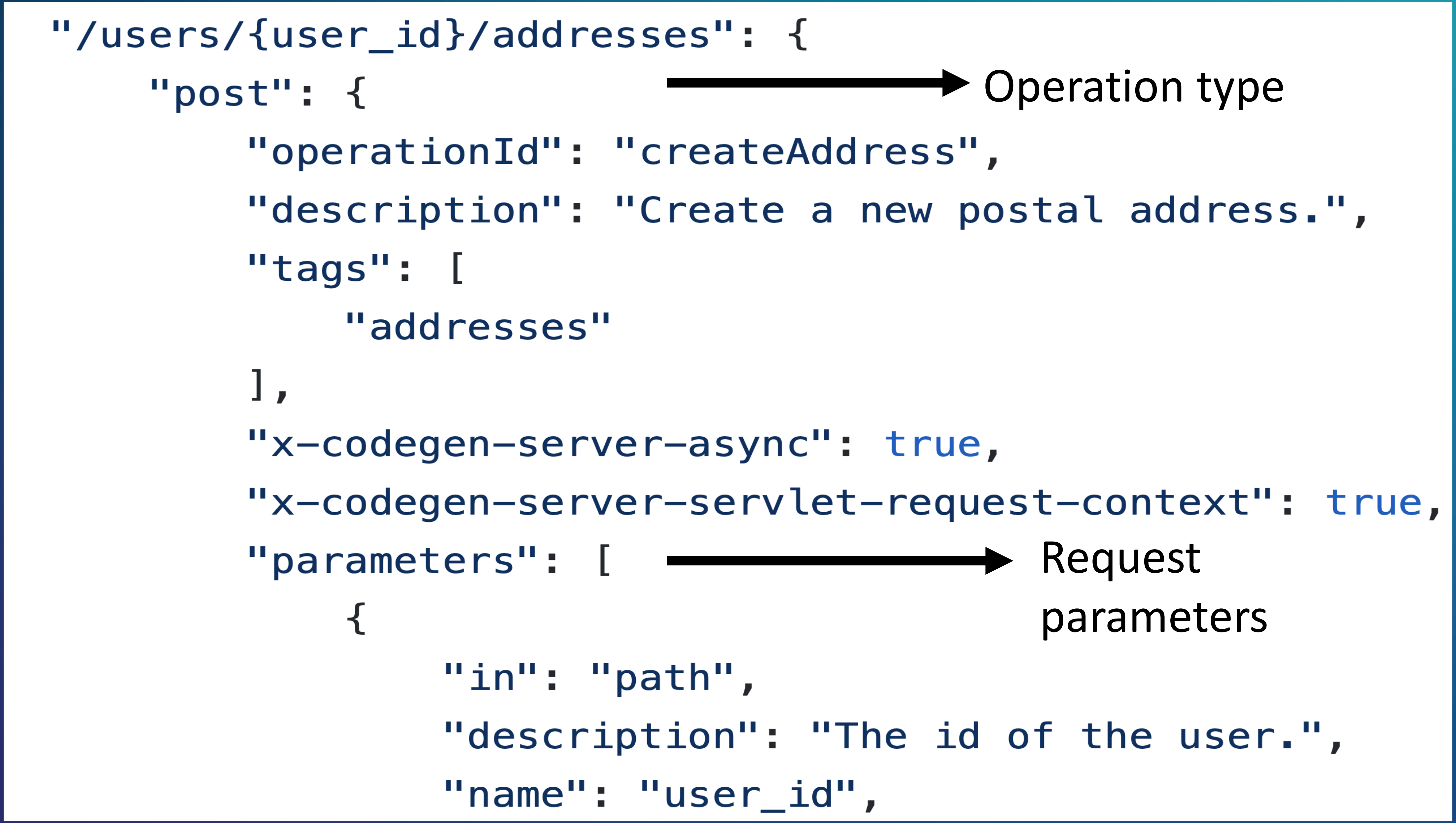
Property-based Testing

Specification

PayPal

#GHC19

Sample Specification File



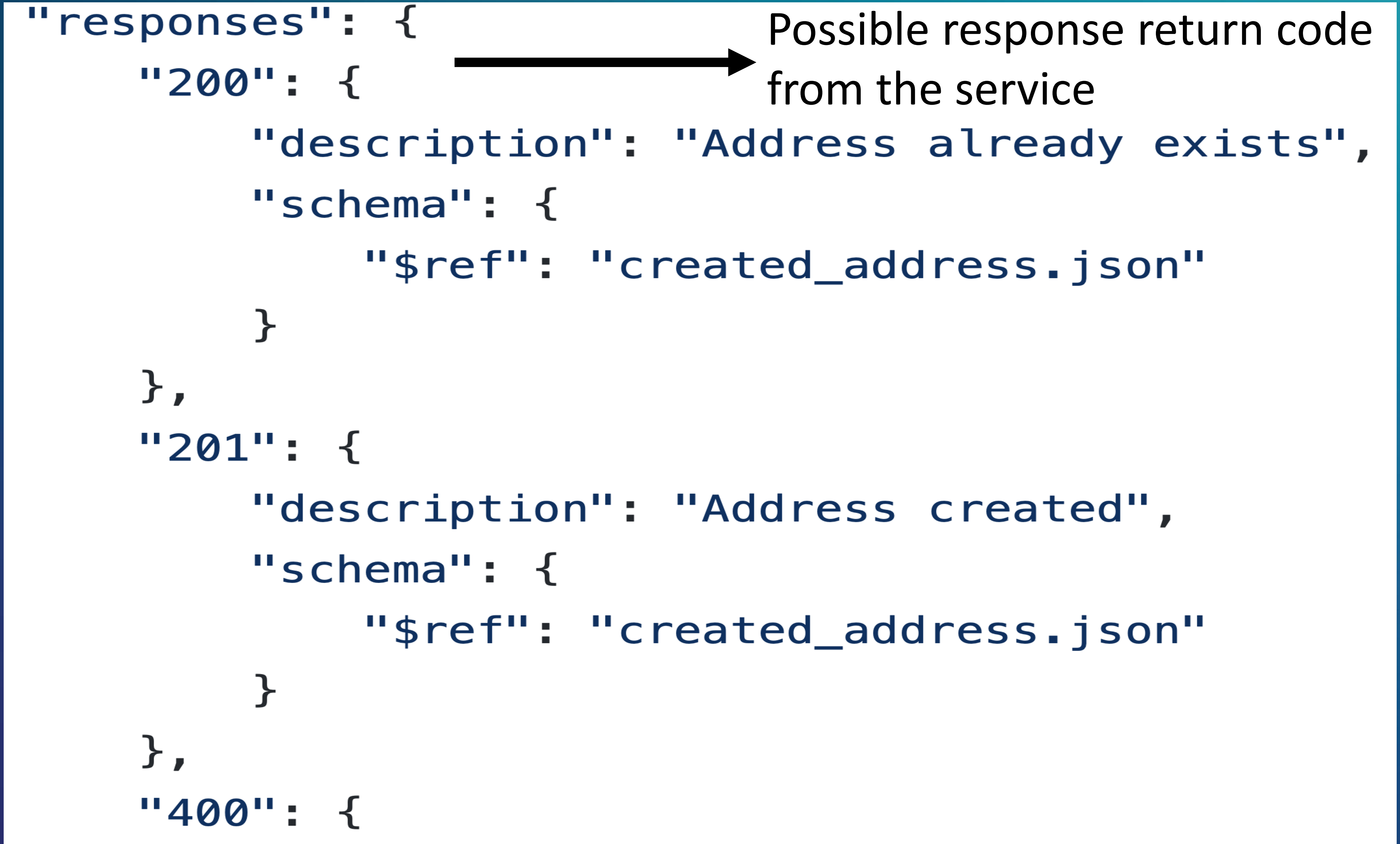
```
"/users/{user_id}/addresses": {
    "post": {
        "operationId": "createAddress",
        "description": "Create a new postal address.",
        "tags": [
            "addresses"
        ],
        "x-codegen-server-async": true,
        "x-codegen-server-servlet-request-context": true,
        "parameters": [
            {
                "in": "path",
                "description": "The id of the user.",
                "name": "user_id",
```

Operation type

Request parameters

#GHC19

Sample Specification File



```
"responses": {
    "200": {
        "description": "Address already exists",
        "schema": {
            "$ref": "created_address.json"
        }
    },
    "201": {
        "description": "Address created",
        "schema": {
            "$ref": "created_address.json"
        }
    },
    "400": {
```

Possible response return code from the service

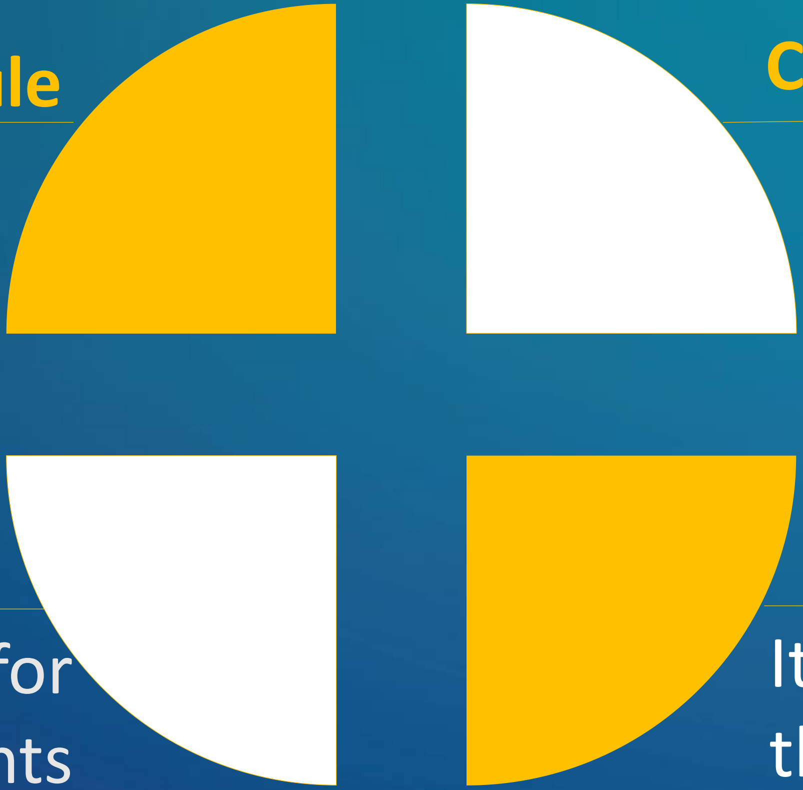
#GHC19

API Specification Driven Testing in Action – SmartTest framework

PayPal

#GHC19

Core Components of the framework



Scenario Interpreter module

Scenarios defined in BDT format – Given, When, Then

Call generation module

Constructs the service call based on the path and parameters from the specification

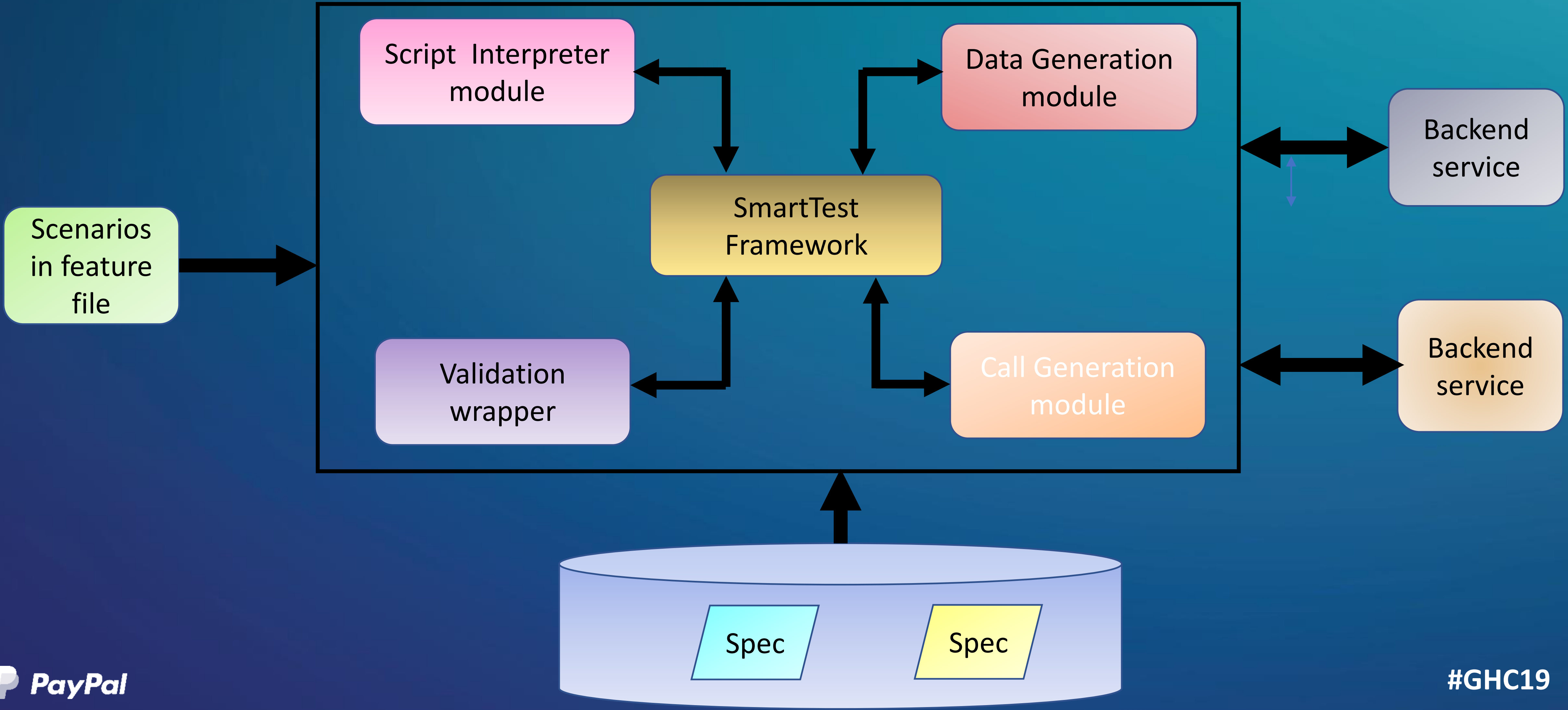
Data Generation Module

Dynamic data generation for a wide range of data points based on the specification

Validation wrapper

It is responsible for handling the “Then” clause of the test scenario definition

SmartTest framework- Architectural model

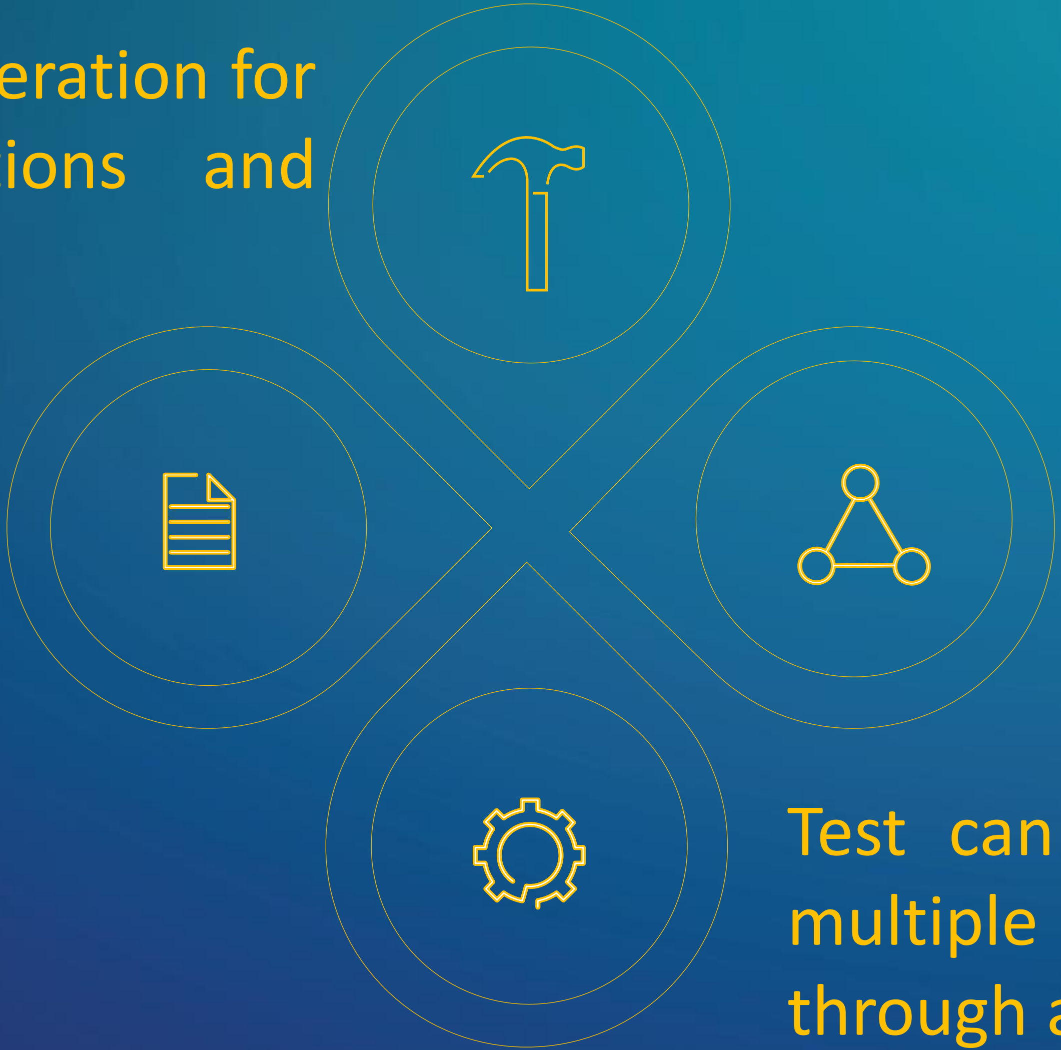


PayPal

#GHC19

Additional Capabilities of the framework

Dynamic data generation for boundary conditions and enumerations



Reusable generic interpreters

End to End testing with multiple backend services

Test can be executed against multiple stacks controlled through a configuration

Sample scenario definitions

Scenario: Add any country based address to a user’s profile
Given 100 US, DE, IN home address
When the address is added to the user profile
Then the address is added successfully

Leveraging a single scenario to validate the use case for a wide range of data

Sample error scenario

Scenario: Add a partial address to a user’s profile
Given a US home partial address without state and zip code
When the address is added to the user profile
Then the address is not added

How the framework enables faster releases with Quality?

- ● Increased test effectiveness
- ● Less churn in product development
- ● Enhanced developer productivity

#GHC19

PayPal

- Reduces friction of test code getting out of sync with specification file

- Huge savings during replatforming initiatives

- Reduction in the testing efforts

Questions?
Carte

References

- https://cucumber.io
- http://www.scalatest.org/user_guide/property_based_testing
- https://www.leadingagile.com/2018/04/step-by-step-toward-property-based-testing/

Please remember to complete the session survey in the mobile app.

THANK YOU

YOU CAN *FOLLOW ME* @

in Sivabalambikai Narayanaswamy

@sivabalambikai

GRACE HOPPER CELEBRATION

ANITA B.ORG

PayPal

#GHC19