Mojaloop TTK PI-12

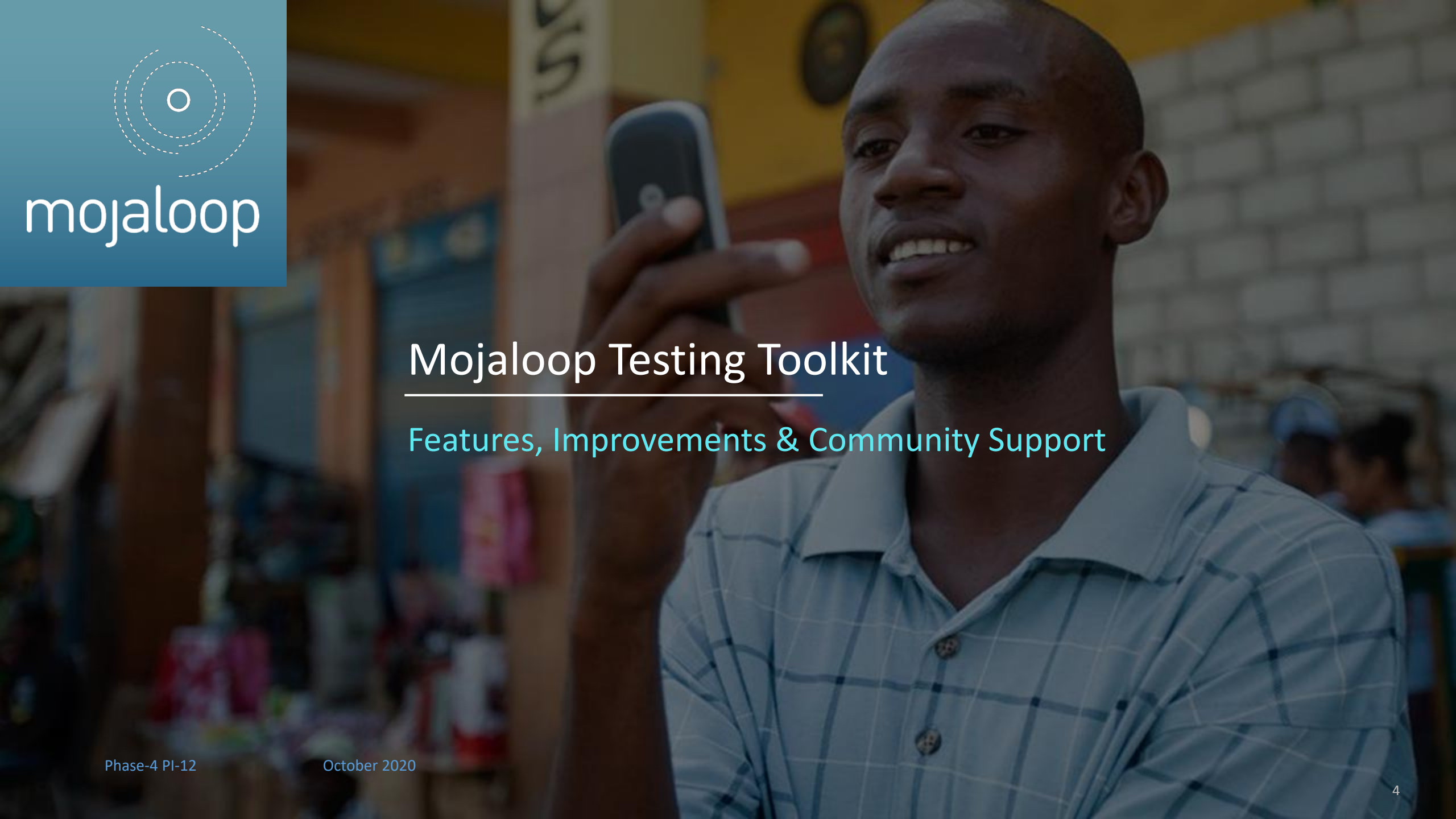
Phase-4 Going live!

Mojaloop PI-12

Phase-4 Going Live!

PI-11 TTK Report-out: Team

1. Georgi Logodazhki
2. Matt Bohan
3. Michael Richards
4. Sam Kummary
5. Sankar Ramakrishnan
6. Vijay Guthi



Mojaloop Testing Toolkit

Features, Improvements & Community Support

Phase-4 PI-12 October 2020

PI-11 TTK: Objectives

1. Support – wider adoption of the testing tool-kit by implementers [by two implementers]
2. Implement a hosted version of the TTK that can be securely & easily hosted by schemes
3. [Stretch] Include a Reference OAuth Server in Helm Chart

PI-11 TTK: General themes

1. Validating (FSPIOP) API implementations
    a. DFSP
    b. Switch / Hub
2. Support schemes (scale) in onboarding FSPs
3. Demonstration capabilities
4. Ease of use by all types of users – reduce business/dev onboarding

Testing toolkit: Enhancements in PI-11

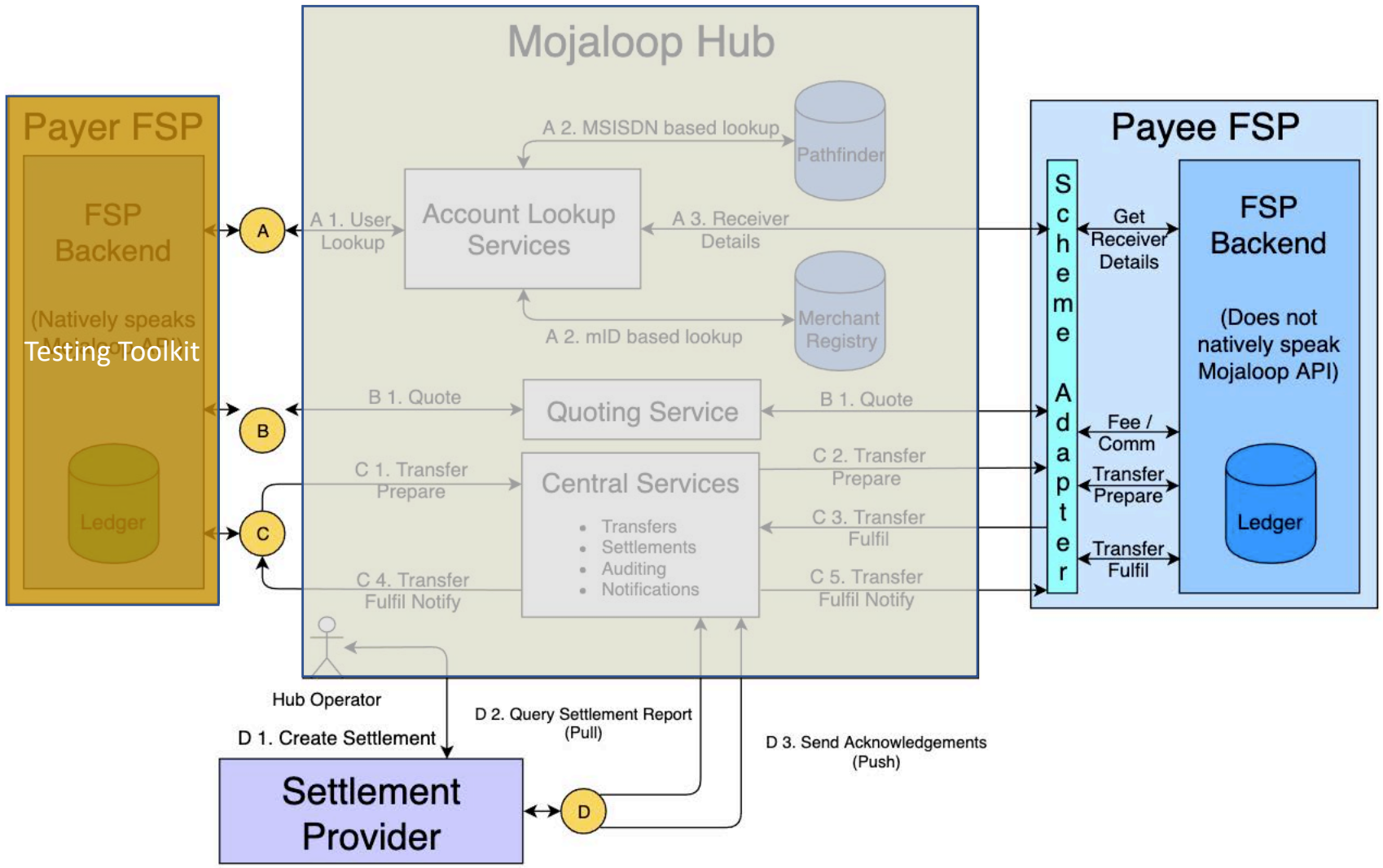
Support wider adoption of the testing toolkit

1. Feature enhancements to support following TTK users
    1. PISP
    2. ATM / POS
    3. Core Team
    4. FXP
2. Hub only mode
3. Divided test cases structure from single JSON file to smaller files and sub folders using sequencing constraints
    a. Support PR Reviews
    b. Collaboration
4. Helm chart for on-demand and scheduled jobs using TTK CLI with AWS S3 and Slack notification support
5. Maintenance and Bugfixes

Hosting capability

1. Implemented database storage in hosted mode
2. Separation of rules, settings, logs and reports per DFSP wise
3. Helm chart for both standalone and hosted mode
4. Integrated keycloak oAuth server

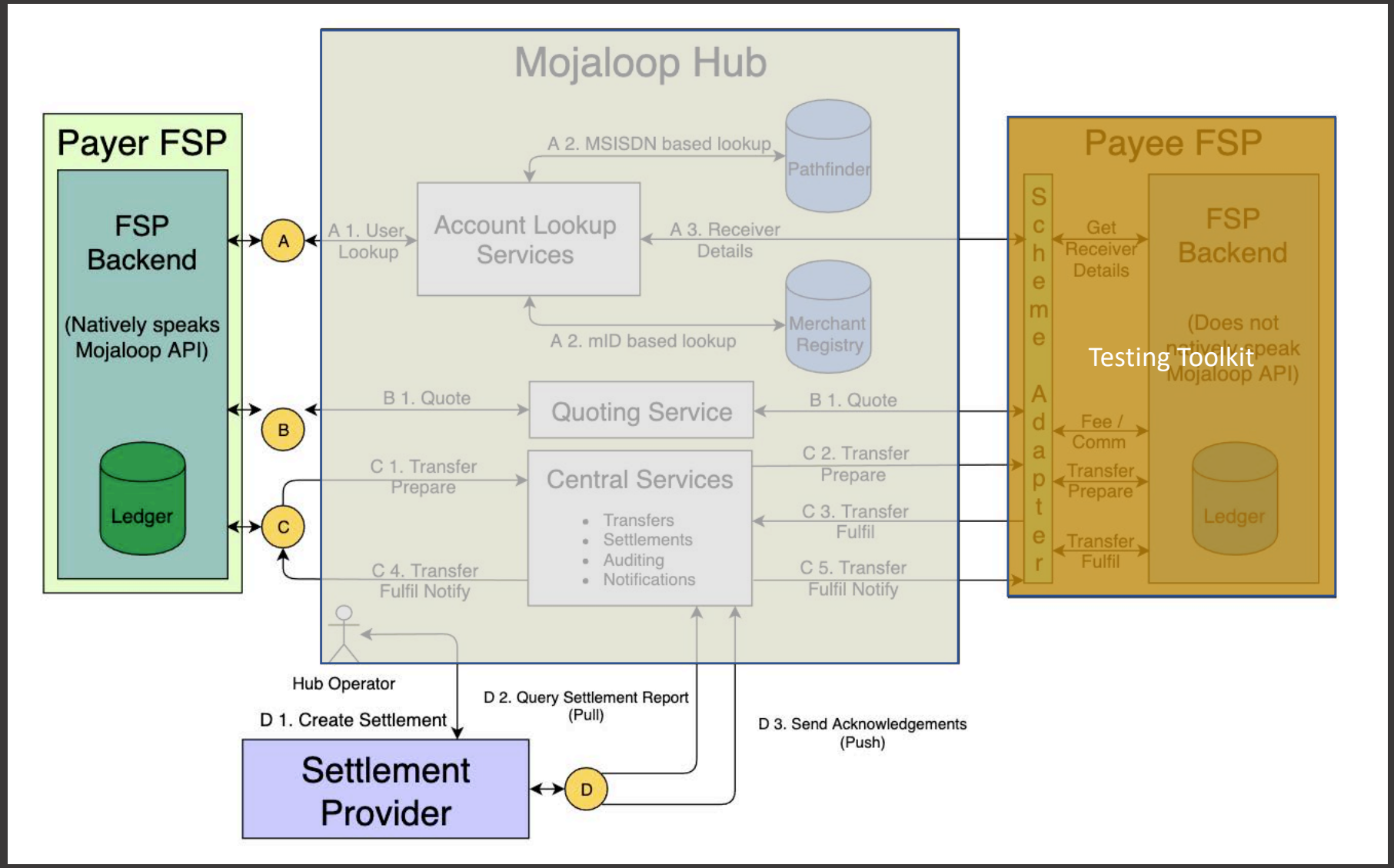
Where does it fit?



Testing Toolkit to HUB

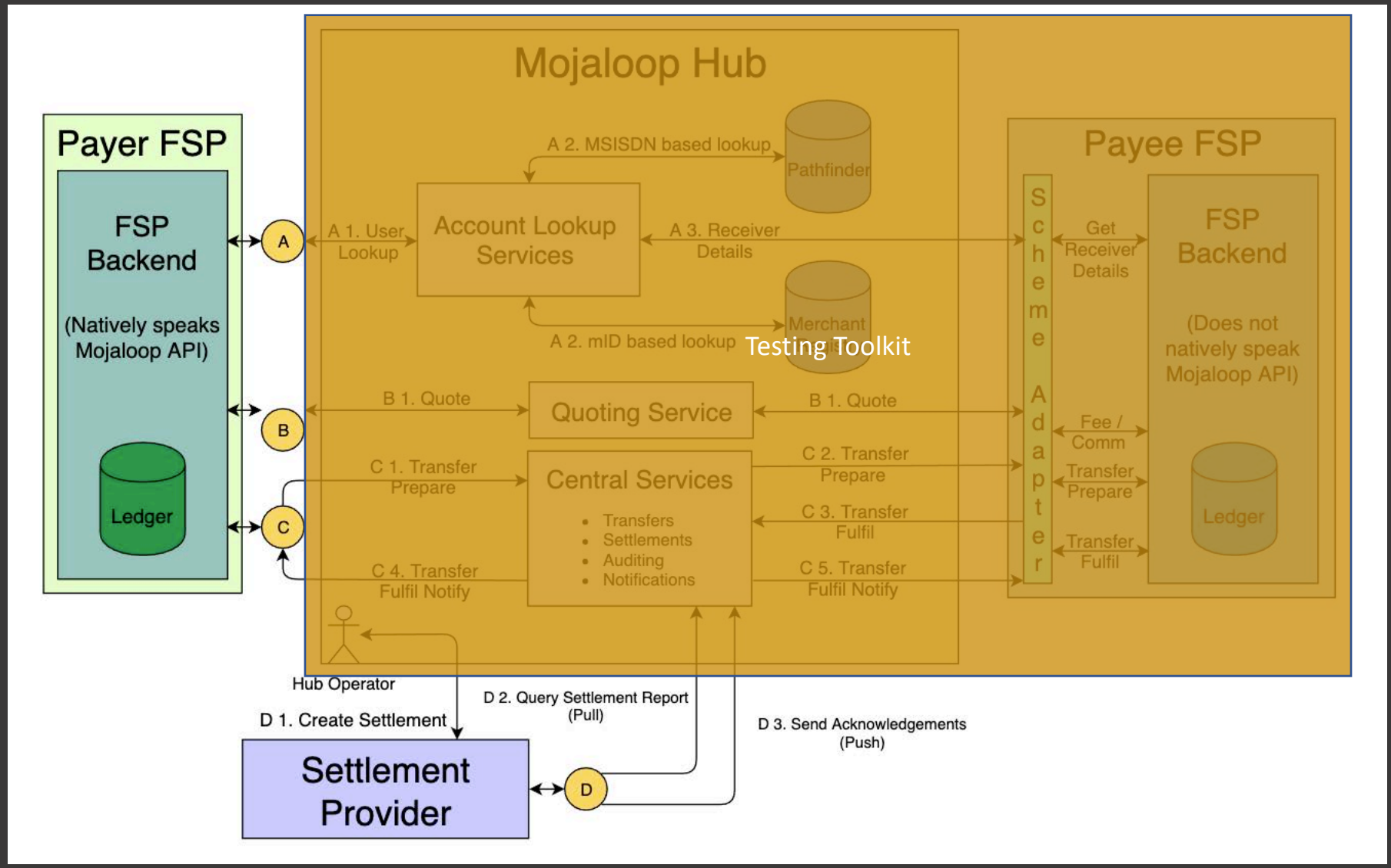
Testing Toolkit to DFSP

Where does it fit?



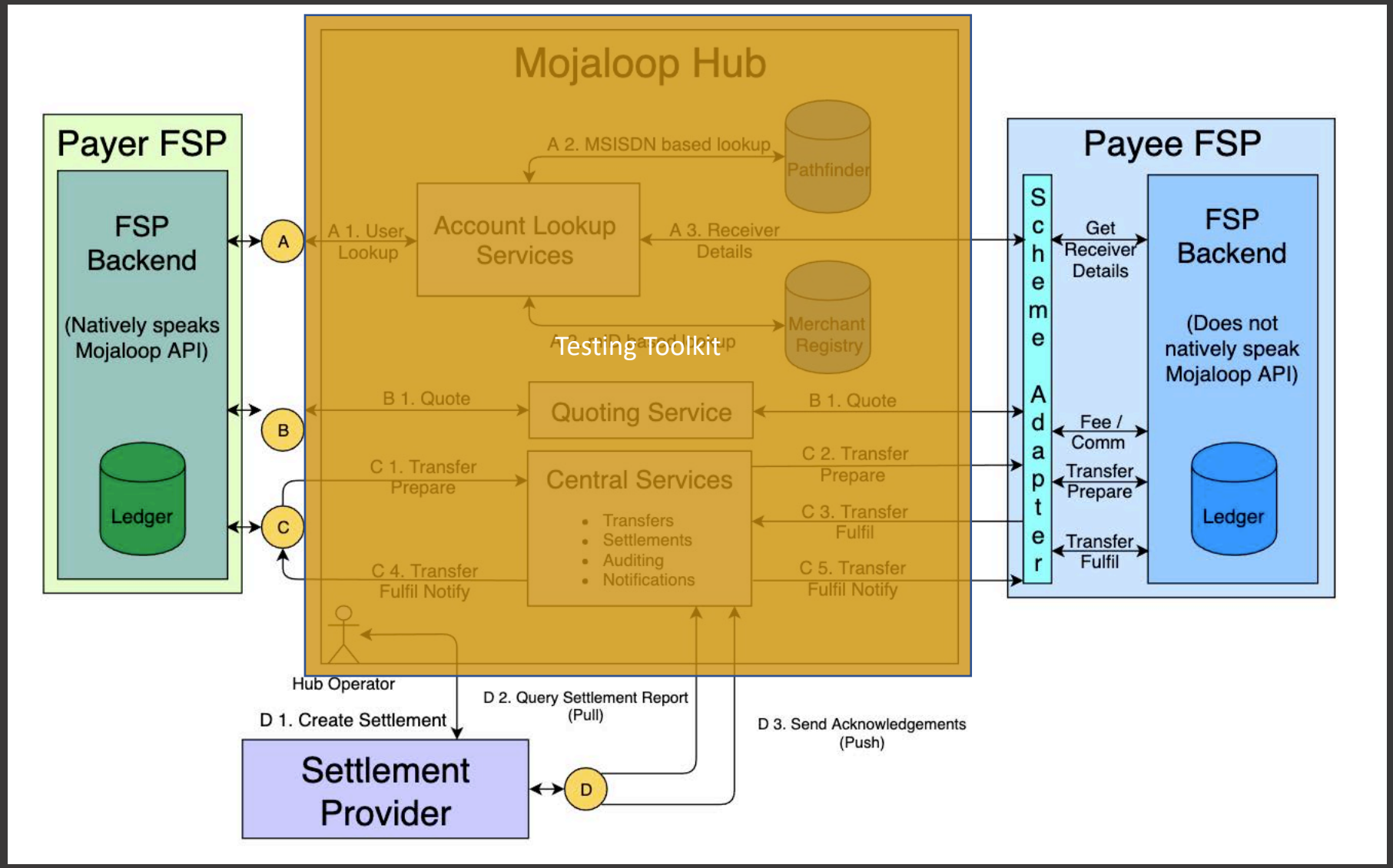
- Testing Toolkit to HUB
- Testing Toolkit to DFSP
- HUB to Testing Toolkit
- DFSP to Testing Toolkit

Where does it fit?



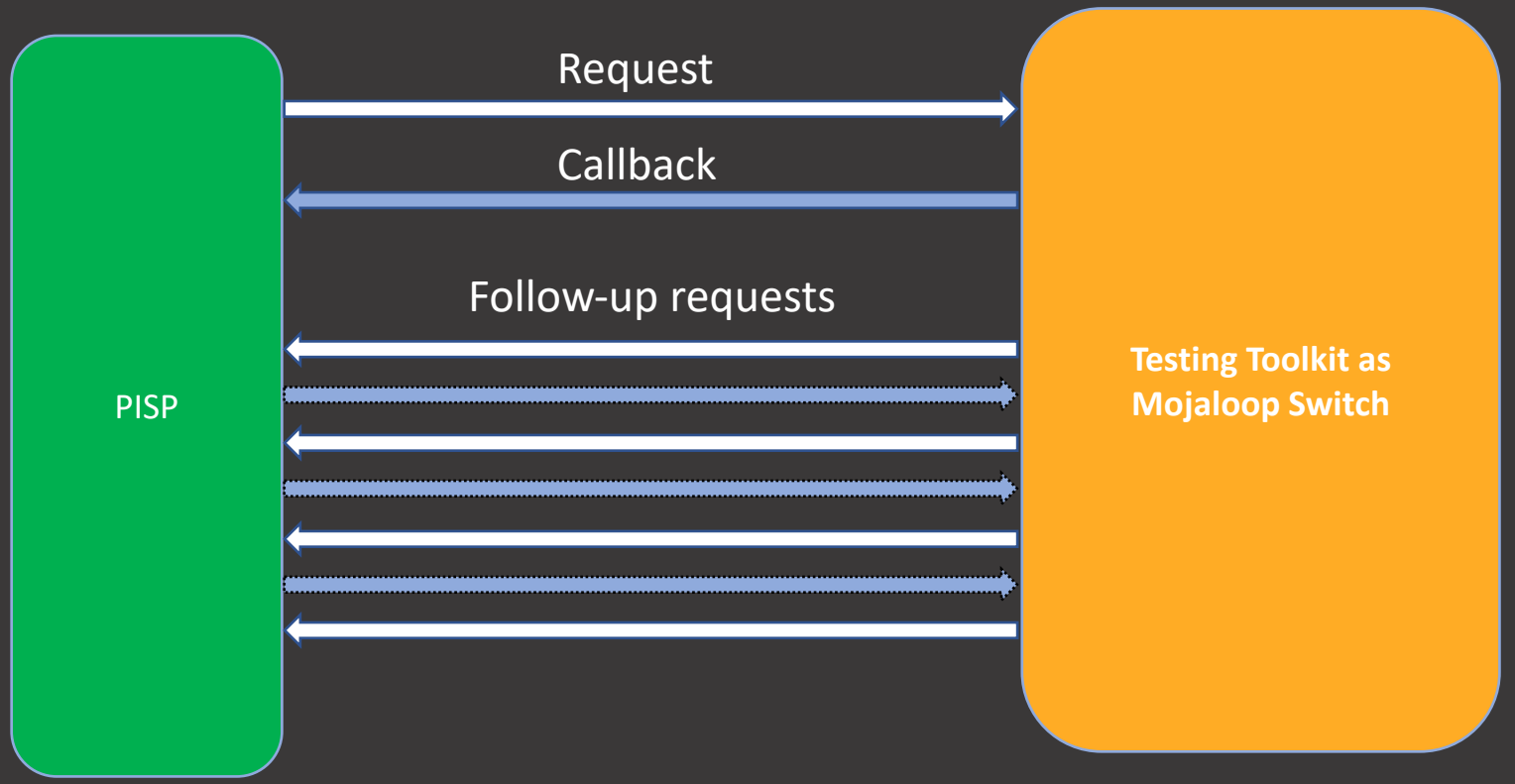
- Testing Toolkit to HUB
- Testing Toolkit to DFSP
- HUB to Testing Toolkit
- DFSP to Testing Toolkit
- As a Hub and Payee FSP

Where does it fit?

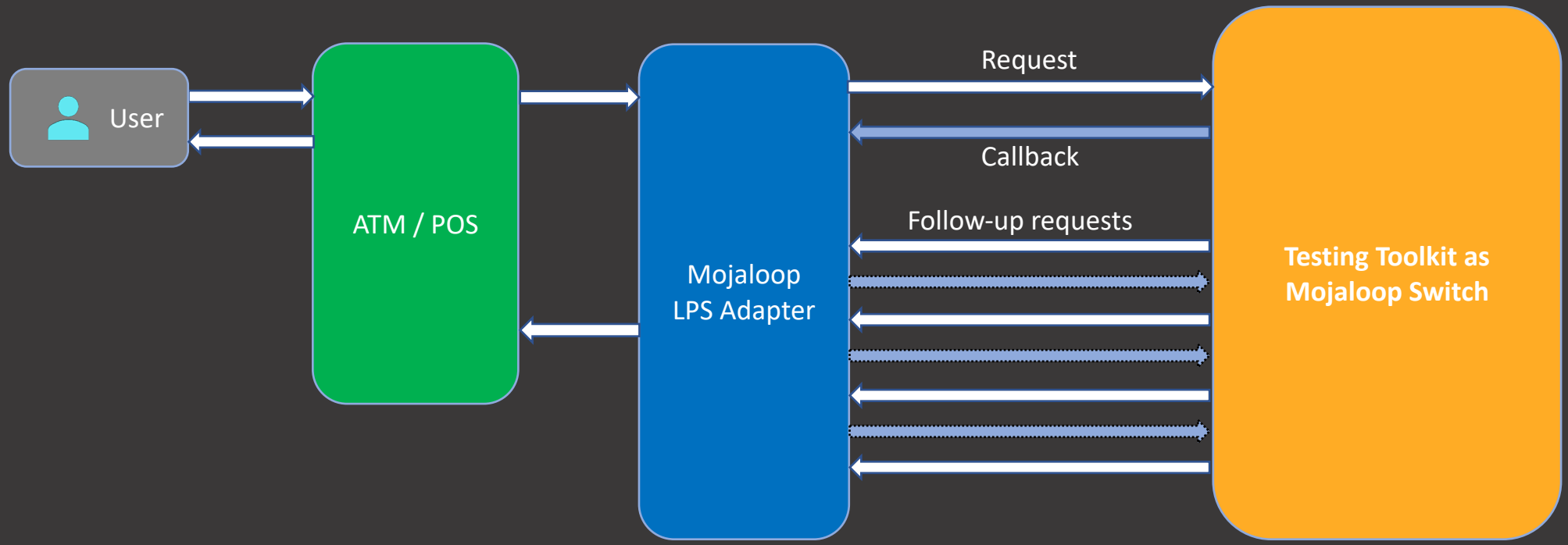


- Testing Toolkit to HUB
- Testing Toolkit to DFSP
- HUB to Testing Toolkit
- DFSP to Testing Toolkit
- As a Hub and Payee FSP
- As a Hub only

Use-case: PISP Team

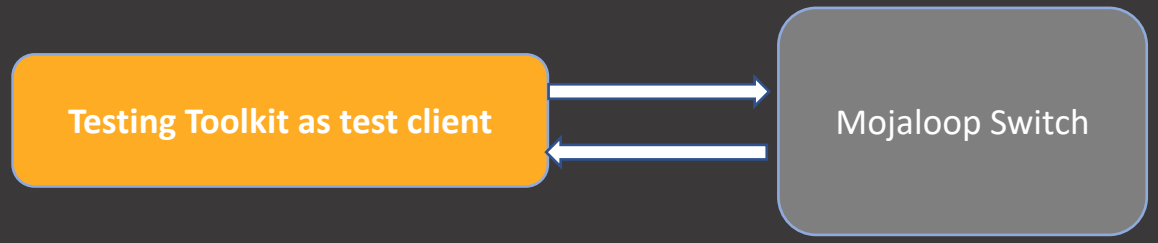


Use-case: ATM / POS

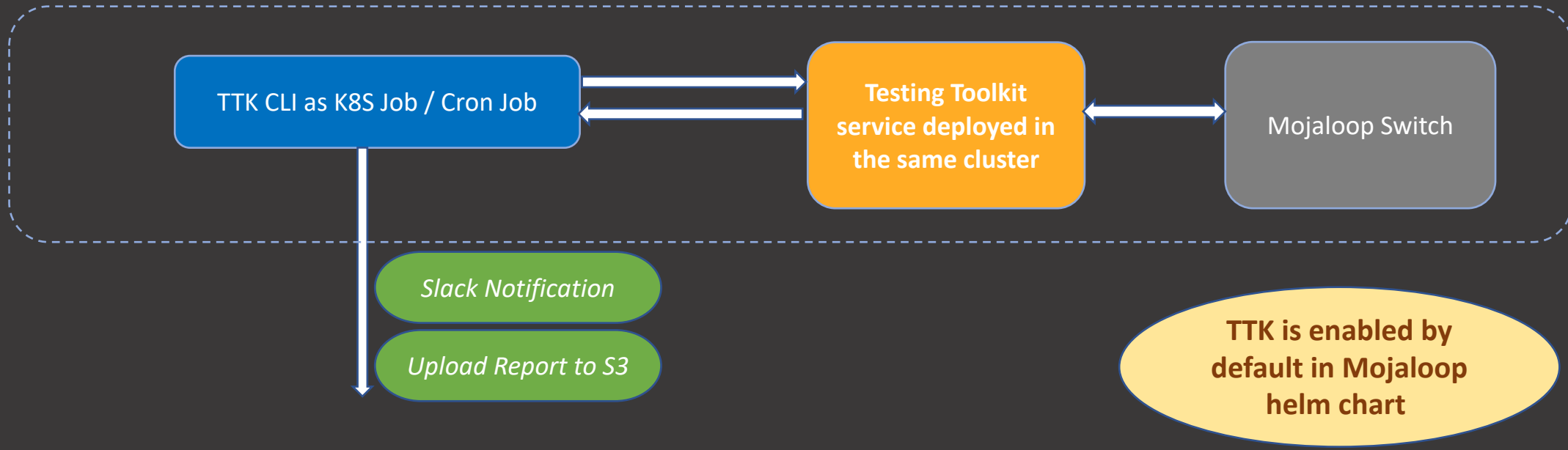


Use-case: Core Team

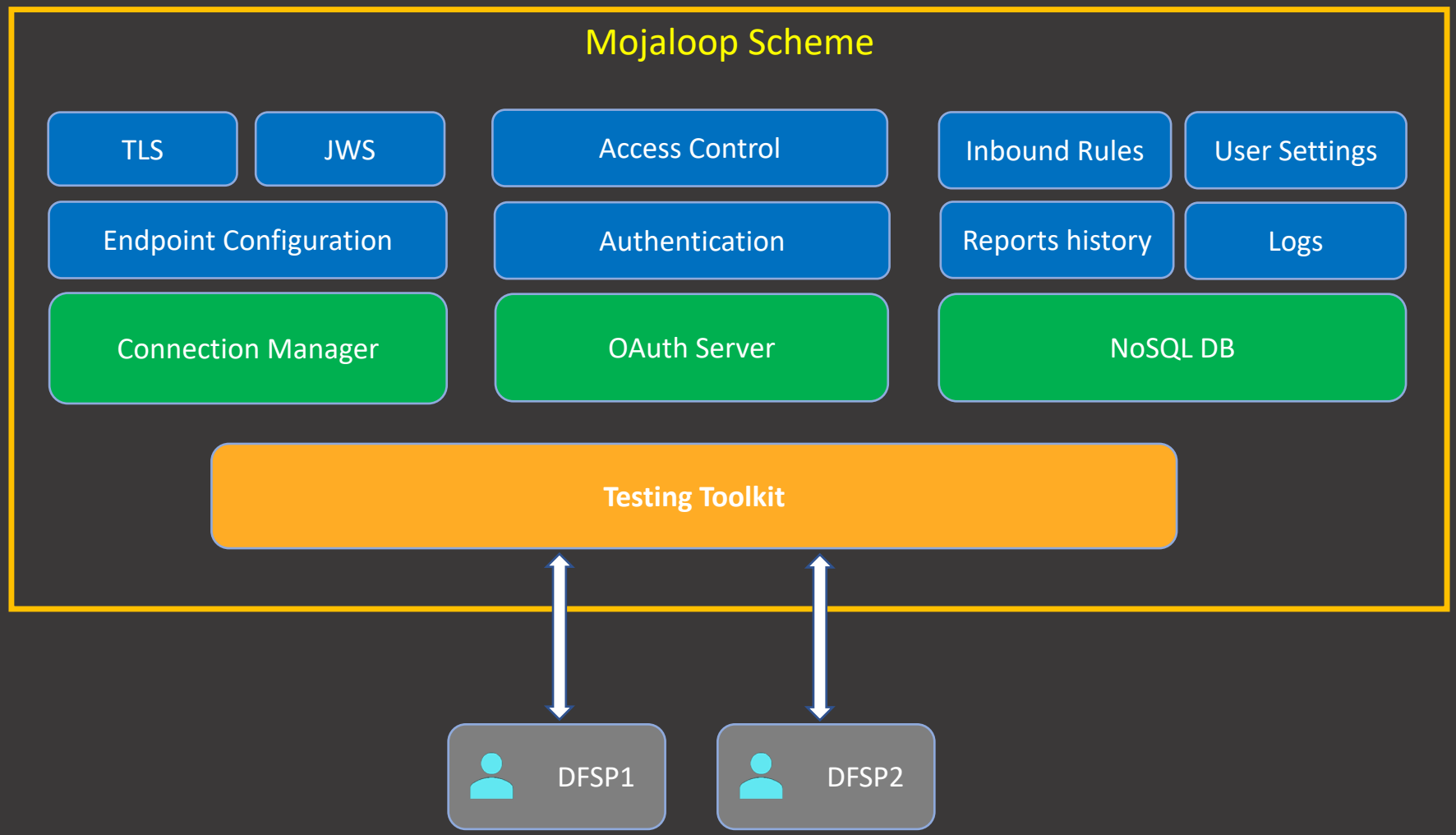
As a replacement for Postman



Scheduled Regression tests



Adding Capability for Hosted Solution

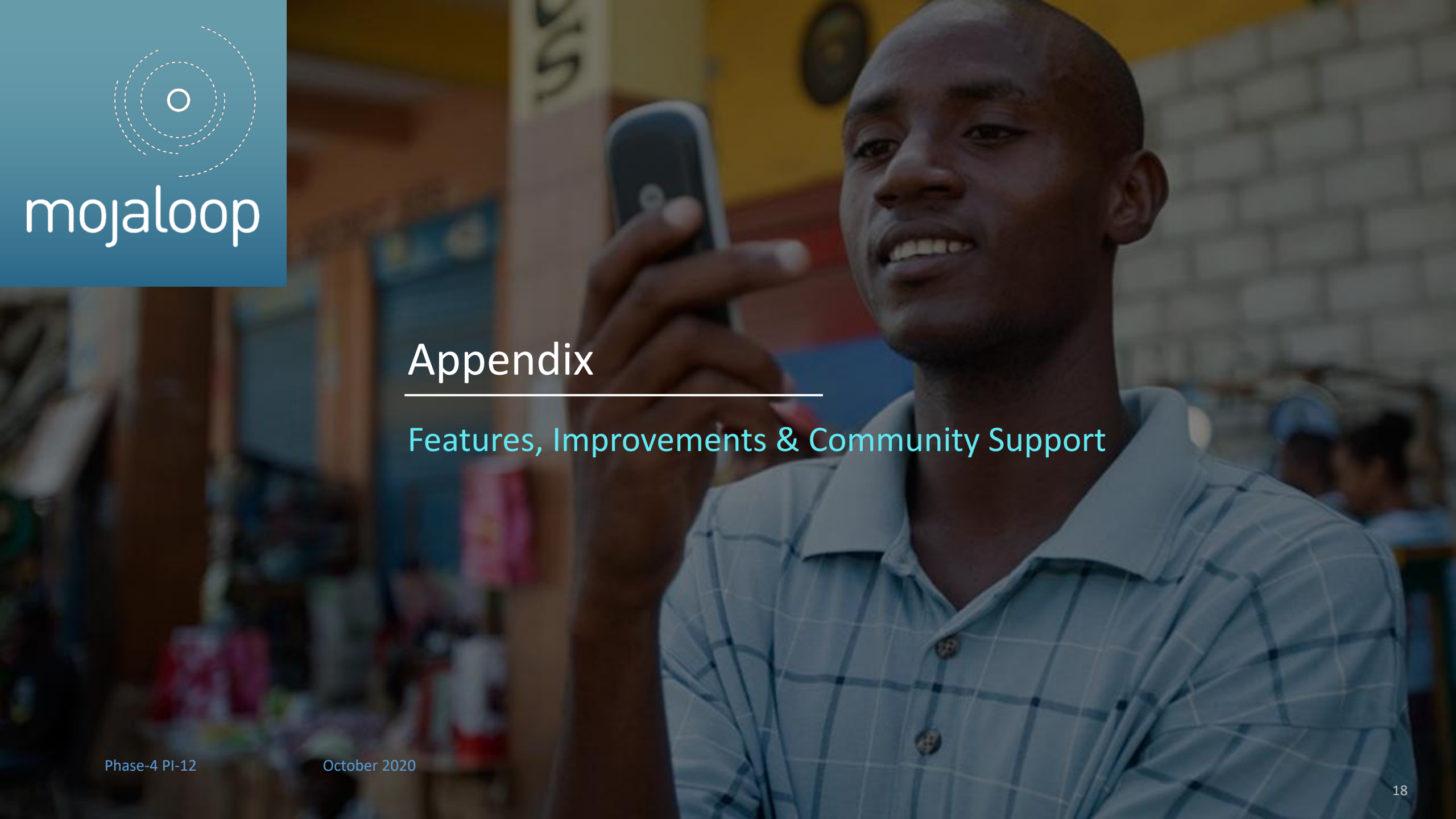


Testing toolkit: Demo

Testing toolkit: Roadmap

1. Support wider adoption
    a. Native javascript support for adding tests
    b. Capability to Validate Hubs for functionality
    c. Interact with implementers for requirements & support adoption for Validating FSPs, Hubs and Onboarding FSPs to Schemes
    d. Usage / UI enhancements based on community feedback
2. TTK enhancements
    a. Event Framework integration
    b. Sequence diagrams for default tests (based on event framework)
    c. Dashboards / home-screen with live data
    d. Distributed architecture: hosted solution micro-service architecture (to enhance load handling)
    e. Scenarios: DFSPs can download and select scenarios with step-by-step instructions for validating their functionality
3. Maintenance & Support
    a. Hardening of the TTK
    b. Training material (After PI-12)

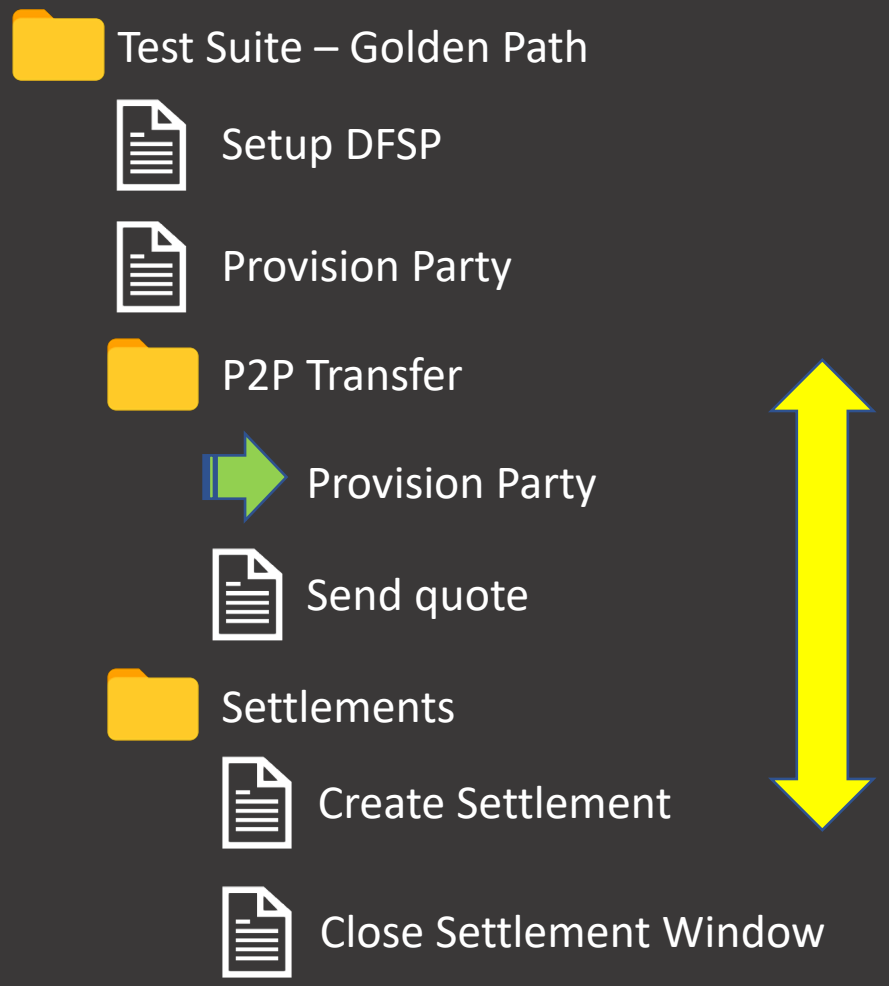
Phase-4 PI-12 October 2020



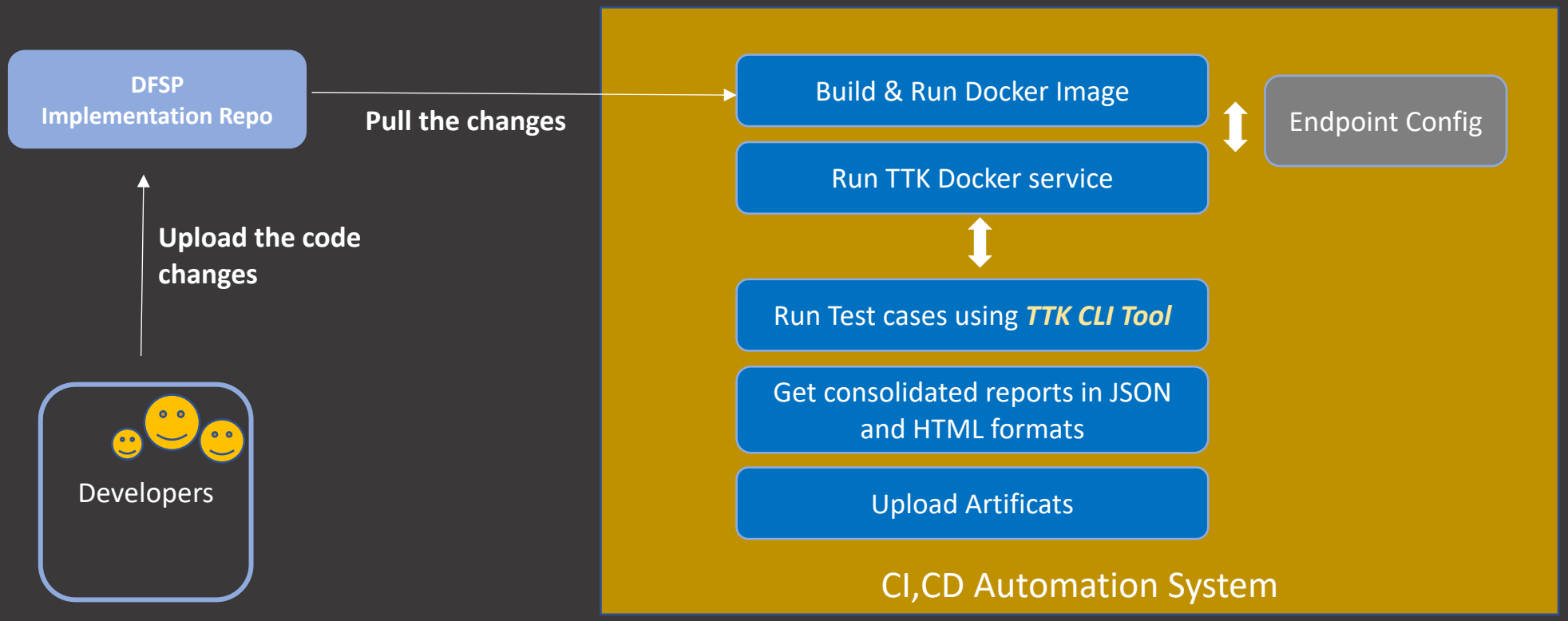
Appendix

Features, Improvements & Community Support

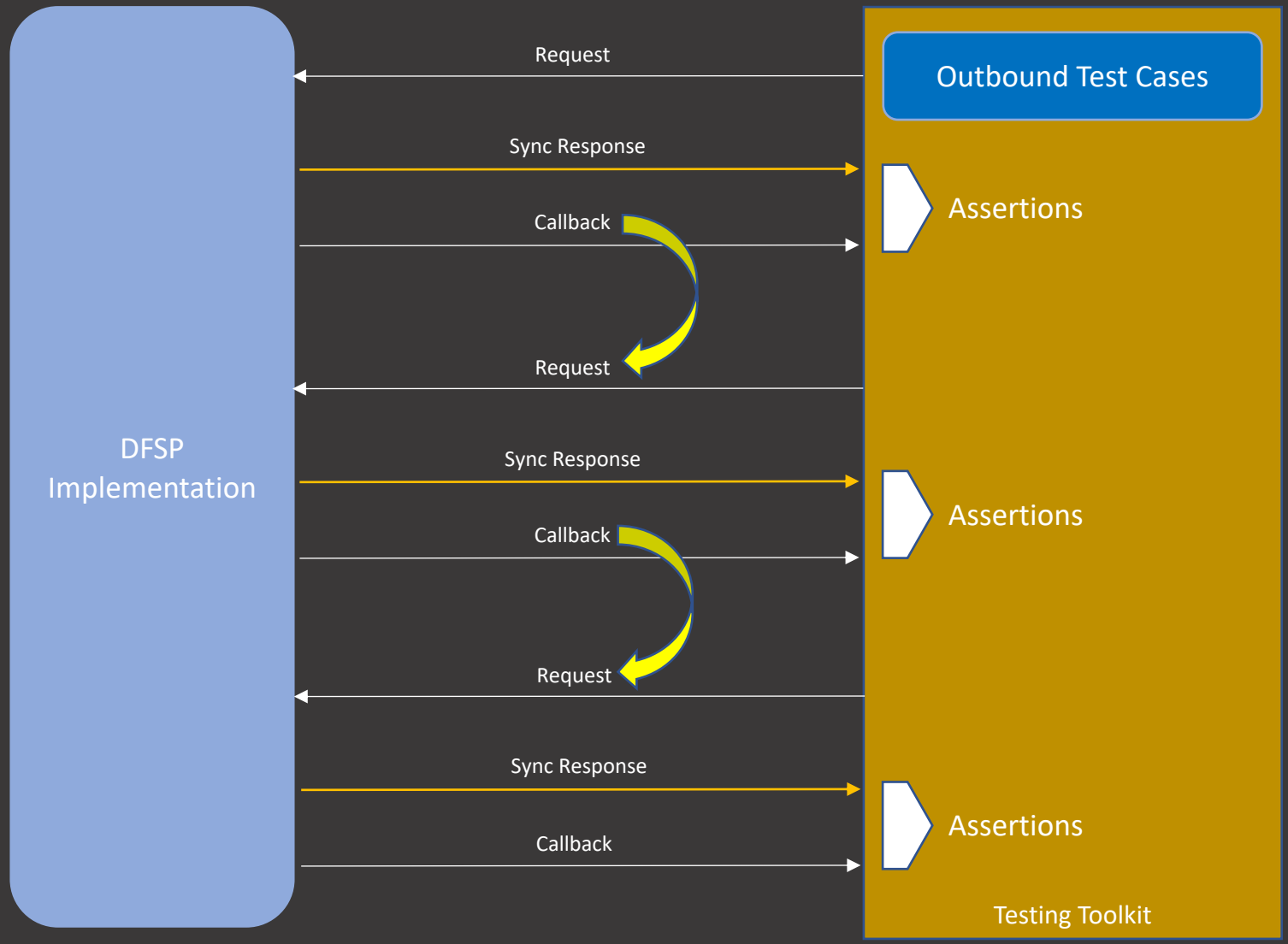
Breakdown the large test case file into folder structure



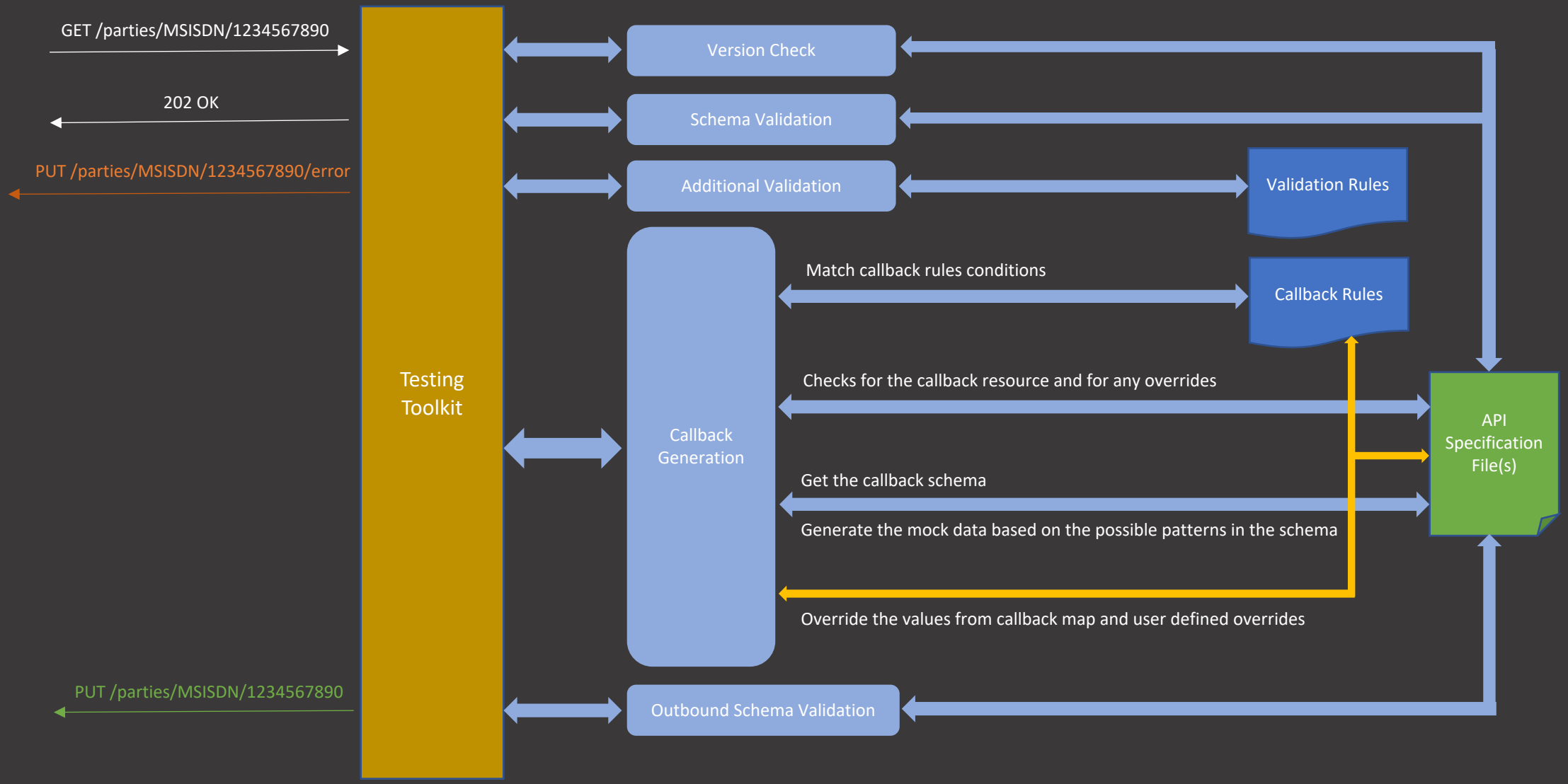
Integrating TTK into DFSP's CI, CD



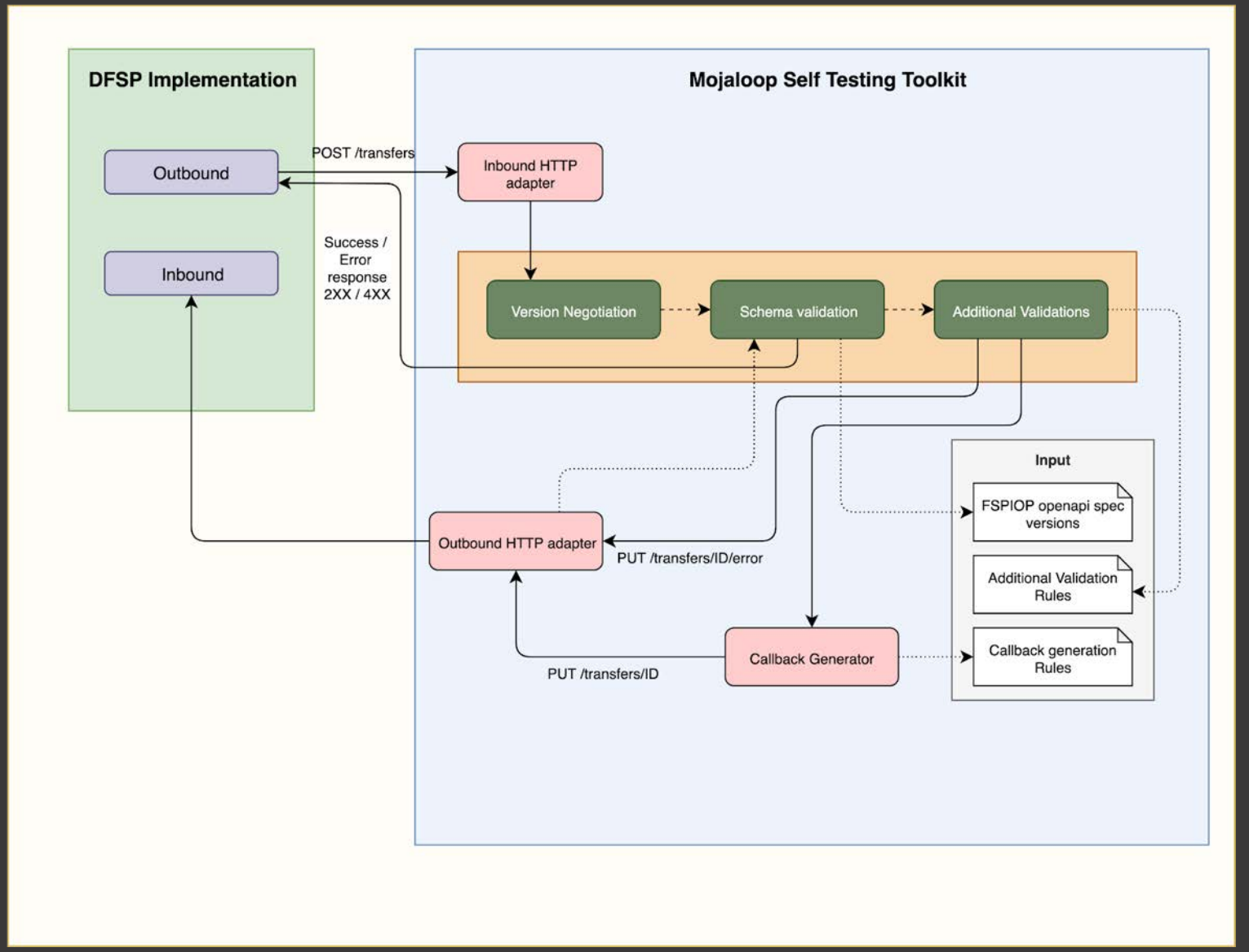
How Testing Toolkit Works - Test Case Initiation and Assertions



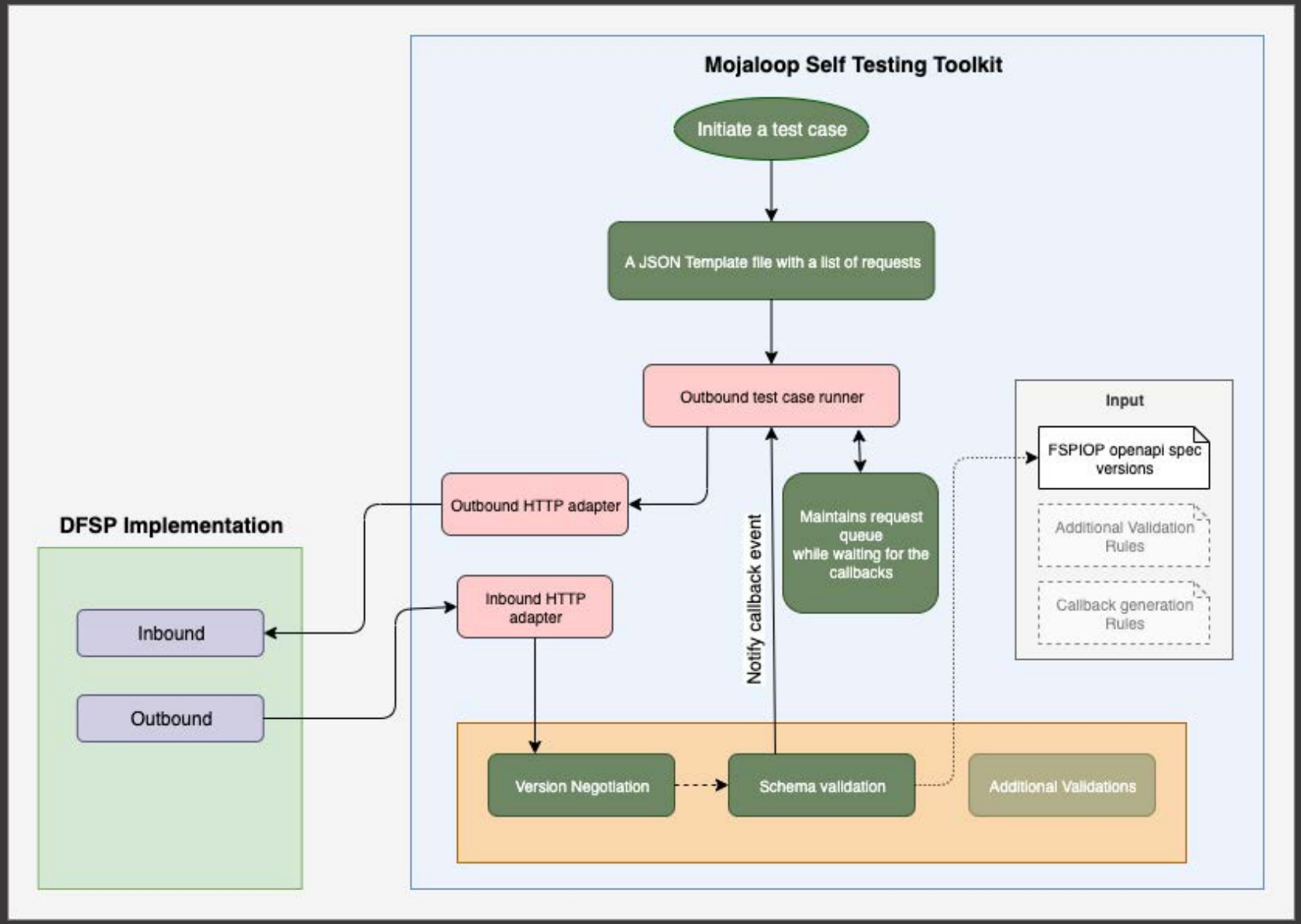
How Testing Toolkit Works - Incoming requests



Architecture Diagram – Part 1



Architecture Diagram – Part 2



PI-11 Toolkit Features

1. User Interface for QA / Product / Business users
2. Version validation and negotiation, schema validation & additional validation
3. Dynamic callback and error callback generation based on rules
4. Initiation of use cases (outbound) - Assertions and report generation
5. Simultaneous support for multiple APIs
6. Separate test sets for Hub and DFSP implementations
7. Synchronous & Asynchronous APIs
8. Supports JWS and mTLS
9. Command line client (CLI) for scheduling & devops automation
10. Easily portable (Light weight and import and export configuration options)

Testing toolkit: Initial goals

1. Test *any* Mojaloop FSPIOP API implementation (initial goal)
2. Simple to use
3. Support different versions of Mojaloop API
4. Highly configurable (Configurations portable)
5. Can validate Inbound requests
6. Can generate Outbound requests

ML OSS Community: Change Control Board

1. Goals, purpose – ML FSPIOP API
2. Current Membership
    i. BMGF (Matt Bohan, Miller Abel)
    ii. Ericsson (Henrik Karlsson)
    iii. Huawei (Chen Hill, Ray C)
    iv. Mahindra Comviva (Ritvik Sinha)
    v. ModusBox [Non-voting] (Michael Richards, Sam Kummary)
    vi. Mowali (John Mark Ssebunya)
    vii. Telepin (Eric Feeley, RJ Wilson)
    viii. BoT TIPS (Mutashobya Mushumbusi)
3. Change requests, Solution proposals, Bugs
4. Restructuring to form Special Interest Groups (SIGs) for each of the APIs in the ML family of APIs and a CCB
5. Working on issues, changes for v2.0 and beyond

Mojaloop FSPIOP API: Evolution