

mojaloop

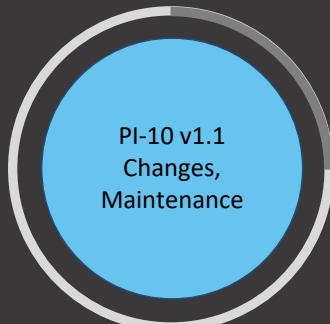
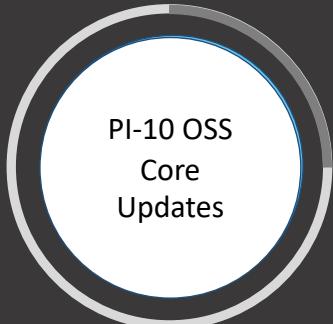
# Mojaloop PI-11

---

Phase-4 Going live!

# Mojaloop PI-11

Phase-4 Going Live!

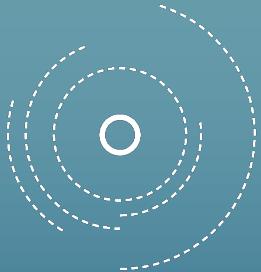






# PI-10 Overview: Agenda

1. PI-10 Overview of changes and progress
2. OSS Community, Collaboration Updates
3. PI-10 Focus Areas



mojaloop

## ML Overview of changes PI-10

---

Features, Improvements & Community Support

# Mojaloop PIs Overview

Timeline	Summary
Phase-1 (2016 - 17)	<b>Level One Project</b> <ul style="list-style-type: none"><li>• Reference Implementation</li><li>• 6 Program Increments (PIs)</li></ul>
Phase-2 (2018)	<b>Road To Productionization</b> <ul style="list-style-type: none"><li>• 1 – 4 Program Increments</li></ul>
Phase-3 (2019 Jan - Dec)	<b>Supporting Adoption &amp; Deployment</b> <ul style="list-style-type: none"><li>• PI-5 (Feb – April): Account lookup, QA Framework, Streamlined CI, Release process, Error endpoints, Documentation, Node Upgrade, Bug Fixes &amp; Community support, Bulk Transfers Design</li><li>• PI-6 Event handling framework, Bulk Transfers PoC, API Gateway, OSS Settlements API, Quoting Service, ALS</li><li>• PI-7 Event &amp; Error Handling framework, Packaging, OSS Settlements, Performance testing capabilities, QA</li><li>• PI-8 Consolidation, Performance, Community Support</li></ul>
Phase-4 (2020 Jan - Dec )	<b>Going Live</b> <ul style="list-style-type: none"><li>• PI-9: Performance Testing &amp; Improvements, Merchant Request to Pay, Operational Monitoring, Testing toolkit, Settlement v2</li><li>• <b>PI-10:</b> Performance PoC, Standardizing Bulk Transfers, Testing toolkit, Settlement v2, Adopting FSPIOP API v1.1, Versioning</li><li>• <b>PI-11 (July – October)</b></li><li>• PI-12</li></ul>

## Mojaloop FSPIOP API v1.0 – Use-cases

### Payer-Initiated Transaction

- [●] P2P Transfers
- [●] Prepares, Fulfils
- [●] Rejections, Timeouts
- [●] Error Endpoints
- [●] Customer-Initiated Merchant Payment
- [●] Customer-Initiated Cash-out - Receive Amount
- [●] Customer-Initiated Cash-out - Send Amount
- [●] ATM-Initiated Cash-out
- [●] Refund

### Bulk Transactions

- [●→●] Bulk Payments

### Payee-Initiated Transaction

- [●] Merchant-Initiated Merchant Payment
- [●] Agent-Initiated Cash-out
- [●] Agent-Initiated Cash-In – Send Amount
- [●] Agent-Initiated Cash-In – Receive Amount

### Payee-Initiated Transaction using OTP

- [●] Merchant-Initiated Merchant Payment Authorized on POS
- [●] Agent-Initiated Cash-out Authorized on POS

#### Key

- [●] Fully implemented
- [●] Supported, not tested
- [●→●] Proof of Concept
- [●] Not implemented
- [○] Out of Scope

# Switch Functionality – Mojaloop (Phase-4 PI-10)

## Mojaloop v1.0: Focus Use-cases

1. P2P
2. Merchant ‘Request to Pay’
3. Bulk Payments

### Payer-Initiated Transaction\*

- [●] P2P Transfers
- [●] Prepares, Fulfils, Query
- [●] Rejections, Timeouts
- [●] Error Endpoints

### Payee-Initiated Transaction\*

- [●] MIMP Transfers
- [●] Transaction Requests
- [●] Prepares, Fulfils, Query
- [●] Rejections, Timeouts
- [●] Error Endpoints

### Bulk Payments\*

- [●] Bulk Transfers
- [● → ●] Prepares, Fulfils, Query
- [● → ●] Rejections, Timeouts
- [● → ●] Error Endpoints

#### Key

- [●] Fully implemented
- [●] PoC / Initial Version
- [●] Partially implemented
- [●] Not implemented
- [○] Out of Scope

# Switch Functionality: Mojaloop End-points (PI-9 → PI-10)

## Mojaloop v1.0 – FSPIOP API Specification

### Transfers^\*

- [●] POST - Prepare
- [●→●] PUT - Response
- [●→●] PUT – Error
- [●→●] PATCH – Notification
- [●] GET - Query

### Parties\*

- [●] GET - Request
- [●] PUT - Response
- [●] PUT - Error

### Quotes

- [●] POST - Request
- [●] PUT - Response
- [●] PUT - Error
- [●] GET - Query

### Participants\*

- [●→●] POST - Create
- [●] PUT - Response
- [●] POST - Bulk Create
- [●] PUT - Error
- [○] DEL - Delete

### Transactions

- [○] PUT - Response
- [○] GET - Query

### TransactionRequests

- [●] POST - Request
- [●] PUT - Response
- [●] PUT - Error
- [●] GET - Query

### Authorizations\*

- [●] GET - Request
- [●] PUT - Response
- [●] PUT - Error

### BulkTransfers\*

- [●→●] POST - Request
- [●→●] PUT - Response
- [●→●] PUT - Error
- [●→●] GET - Query

### BulkQuotes\*

- [●→●] POST - Request
- [●→●] PUT - Response
- [●→●] PUT - Error
- [●→●] GET - Query

### Key

- [●] FSPIOP API v1.1 changes
- [●] Fully implemented
- [●] PoC / Initial Version
- [●] Partially implemented
- [●] Not implemented
- [○] Future Roadmap

# PI-10 Switch Functionality – Operations

## Operational – Use Cases

### Participants

- [●] Manage Participants
- [●] Create Initial Value
- [●] Query
- [●] Update
- [●] Manage Participant Limits
- [●] Create Initial Value
- [●] Query
- [●] Update
- [●] Manage Callback URLs
- [●] Create Initial Value
- [●] Query
- [●] Update

### Oracles (ALS)

- [●] Manage Oracles
- [●] Create
- [●] Query
- [●] Update
- [●] Delete

### Monitoring, Tracing

#### Tracing

- [●] Transfers
- [●] Quotes
- [●] ALS

#### Metrics

- [●] Transfers
- [●] Quotes
- [●] ALS

#### Key

- [●] Fully implemented
- [●] PoC / Initial Version
- [●] Partially implemented
- [●] Not implemented
- [○] Roadmap for PI10

# PI-10 Switch Functionality – Settlement

## Settlements v1.0 [Supports some forms]

- [●] Open, close Settlement Windows
- [●] Query Settlement Windows
- [●] Query Settlement Report
- [●] Create/Trigger Settlement with Windows
- [●] Process successful Settlement Acknowledgements
- [●] Reconcile Positions based on successful Settlements
- [●] Process failed Settlement Acknowledgements

## Positions

- [●] Query Positions
- [●] Manage Positions
- [●] Create Initial Value
- [●] Query
- [●] Update

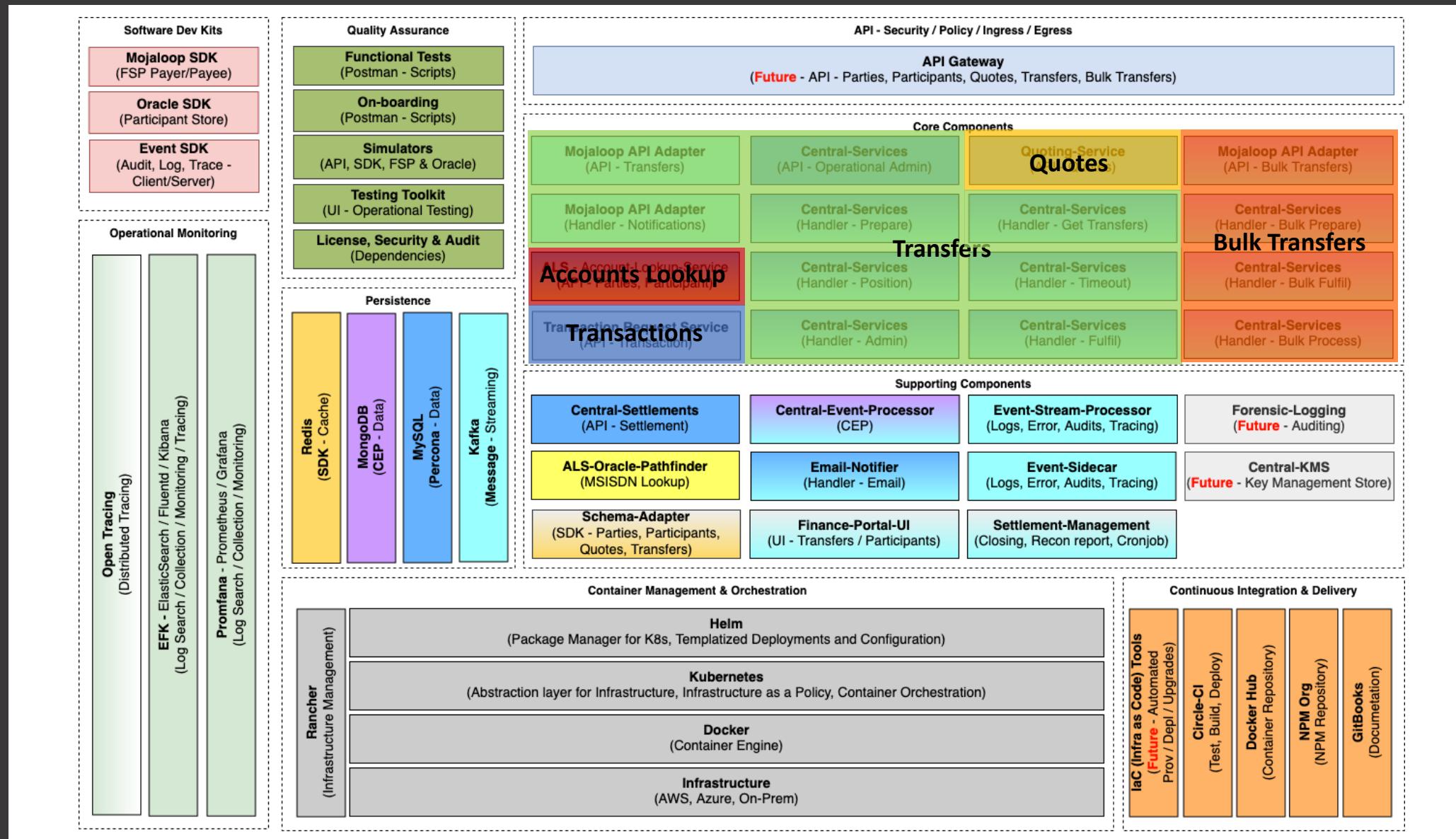
## Settlements v2

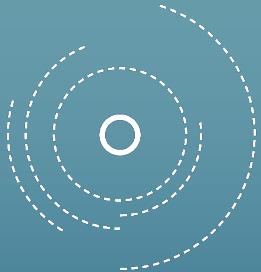
- [●] Settlement models
- [●] Designs
- [●→●] Settlement by Currency
- [○→●] Interchange Fees
- [○→●] Continuous Gross Settlement

### Key

- [●] Fully implemented
- [●] PoC / Initial Version
- [●] Partially implemented
- [●] Not implemented
- [○] Roadmap for PI10

# PI-10: Component Architecture





mojaloop

## ML Community Updates

---

Features, Improvements & Community Support

# PI-10 Overview: Mojaloop OSS Community

## Contributions, Collaboration

1. Cross network/currency
2. Code quality and Security improvements
3. Versioning Standards
4. ISO Integration
5. Quality Assurance (bugs, coverage, Tests with mojaloop-simulator)
6. Enable JWS services on outbound messages
7. Settlement v2
8. PISP Solution

# Mojaloop OSS PI-10: Community

1. Change Control Board (CCB)
2. Design Authority (DA)
3. Weekly scrum-of-scrums
4. Slack channels: `#general`, `#announcements` , `#help-mojaloop`, `#design-authority` , `#ml-oss-devs`

# ML OSS Community: Design Authority

1. Responsibilities
2. Membership – driven by contribution
3. Frequency of meetings, boards used
4. Functioning overview

## DA Issue Board

<https://tinyurl.com/y6bnj7sz>

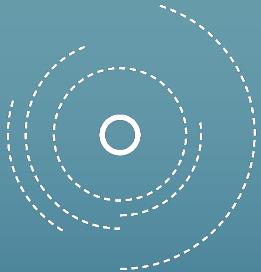
# ML OSS Community: Design Authority

## Responsibilities

- Ensure a uniform Architecture
- Provide a channel for bringing ideas for Design and Verification
- Define Technical Strategies
- Verify Architectural Standards
- Drive Design Methodologies

# ML OSS Community: DA issues overview

4 Issues - 0 Story Points <b>Identified</b>  design-authority #41 Merchant-Initiated Merchant Payment functionality on Mojaloop Simulator   design-authority #49 Recognising enumerations   design-authority #57 Investigate solution to avoid recompiling node-rkafka at each npm install both in docker and in host   design-authority #60 OSS-Core coding discussion points	12 Issues - 0 Story Points <b>Assigned</b>  design-authority #58 central-settlement merge into central-ledger   design-authority #59 Cryptographic Processing Module   design-authority #55 Contingency Plan discussion (Generic) for Open-Source Library Support changes   design-authority #53 Placement of Rules modules   design-authority #45 Mojaloop Post Transfer Processor   design-authority #43 Design a mechanism for informing a PISP on the	10 Issues - 0 Story Points <b>Deferred</b>  design-authority #51 Should the PISP implementation live in the `sdk-scheme-adapter`, or should we make a new `thirdparty-scheme-adapter`?   design-authority #34 Merchant Oracle - Registering Merchant ID   design-authority #31 Oracle Design - ALIAS usage for TIPS implementation   design-authority #11 Async Process Log file creation   design-authority #12 Security	14 Issues - 0 Story Points <b>Decision Made</b>  +2 design-authority #54 zero down-time deployment proposal   design-authority #52 Deprecation of Helm2 support   design-authority #44 Understand and Define Mojaloop Roles for PISP, x-network, etc. use cases   design-authority #46 Discuss the PISP Simulator   design-authority #47 Answer the question of whether to have a separate API for PISP, or simply extend the existing Open API   design-authority #40
--	---	--	--



mojaloop

## PI-10 Core team "Focus Areas"

---

Features, Improvements & Community Support

## PI-10 Overview: Focus Areas

1. FSPIOP API v1.1 changes: Implementation
2. Maintenance: Helm releases, Support for accented characters, QA
3. Bulk Transfers: Standardization, -ve scenarios, Query
4. Testing toolkit: Support for - Hub testing, usability and adoption
5. Performance: Perf PoC, Support streaming improvements

# PI-10: Performance

## Completed/In-Progress

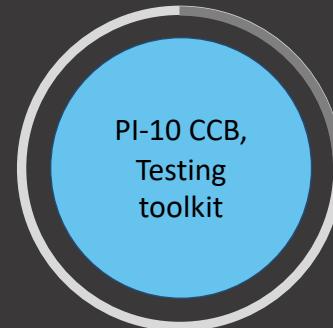
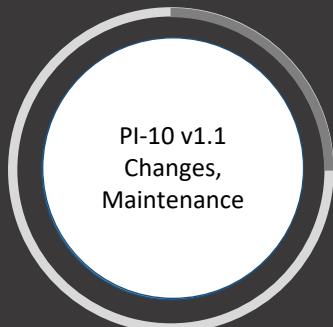
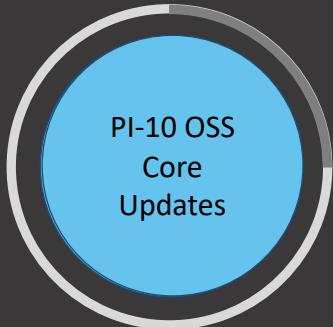
1. Kafka Workshops with Confluent
  - a) Best-practices & Learnings on Kafka Architecture & deployment / infrastructure
2. Perf Environment
  - a) Automated config ([collaboration](#))
  - b) Automated startup/shutdown ([collaboration](#))
3. Mojaloop Kafka Client (Central-Services-Stream)
  - a) Client [Instrumentation](#): Monitoring Producer/Consumer internal queues (fetch, send lag), number of messages read/send, etc
  - b) Provided [support](#) & [verification](#) of client [queue management](#) and [parallel optimizations](#) ([collaboration](#))
4. Proof of Concept ([PoC](#)) utilizing Domain-Driven-Design, Event-driven CQRS & Event-Sourcing Architecture ([collaboration](#))

## Roadmap

1. Perf Environment
  - a) Implement Best-Practices/learnings from Kafka Workshops ([Infrastructure as Code](#))
  - b) Automated config ([roll-out](#))
  - c) Automated startup/shutdown ([roll-out](#))
2. Mojaloop Kafka Client ([Central-Services-Stream](#))
  - a) Optimized Producer from ([PoC](#))
    - a) High-level producer implementation ([improved resilience](#)) & config ([improved performance](#))
  - b) Optimized Consumer from ([PoC](#))
    - a) Simplified consumer implementation ([improved maintainability & performance](#)) & config
  - c) Instrumentations
    - a) Client Monitoring ([Pull-request pending](#))
    - b) Dashboards

# Mojaloop PI-11

Phase-4 Going Live!



## FSPIOP API v1.1 Changes - 1: Support Payee FSP Notification

- [●] Option to notify completion of a transfer to a Payee - [Story#1334](#)

### Behaviour in v1.0

The Payee FSP needs to send a *GET /transfers/<ID>* message to the Switch to know the status

### Desired behaviour

The Payee FSP can send the *PUT /transfers/<ID>* to the Switch with an option to indicate that it would like to receive a notification when the transfer is finalized in the Switch.

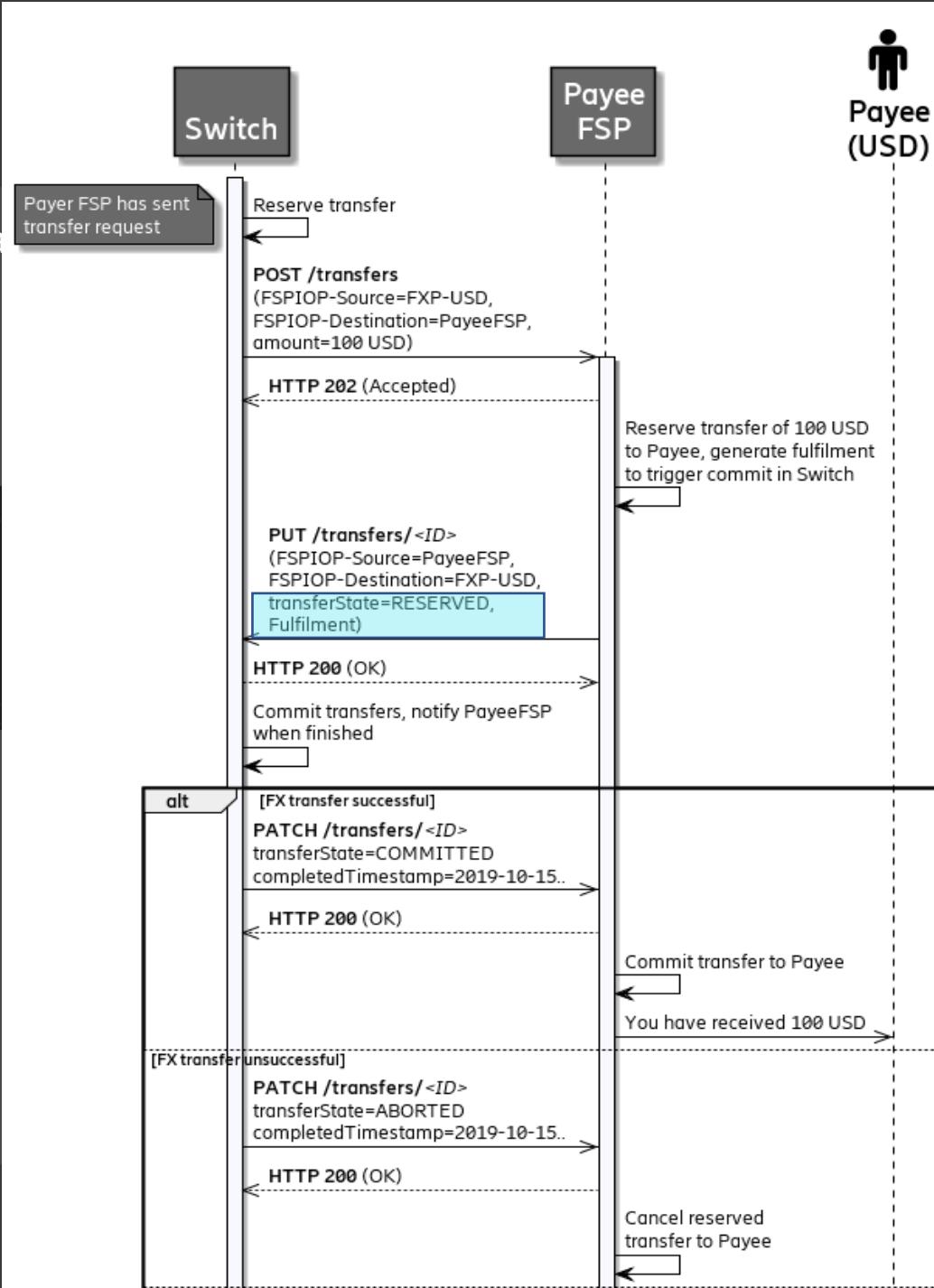
# FSPIOP

- [●] Option to notify completion of a transfer

```
PATCH /transfers/85feac2f-39b2-491b-817e-4a03203d4f14 HTTP/1.1
Content-Type: application/vnd.interoperability.transfers+json;version=1.0
Content-Length: 166
Date: Tue, 15 Oct 2019 08:14:03 GMT
FSPIOP-Source: Switch
FSPIOP-Destination: DFSP2
FSPIOP-Signature: {"signature": "YXBwbGlj...JzK2pzb2"}
{
  "completedTimestamp": "2019-10-15T08:14:03.113+01:00",
  "transferState": "COMMITTED",
}
```

## Updated central services

1. central-services-shared
2. central-ledger
  1. Fulfil handler
  2. Position handler
3. ml-api-adapter



## Notification

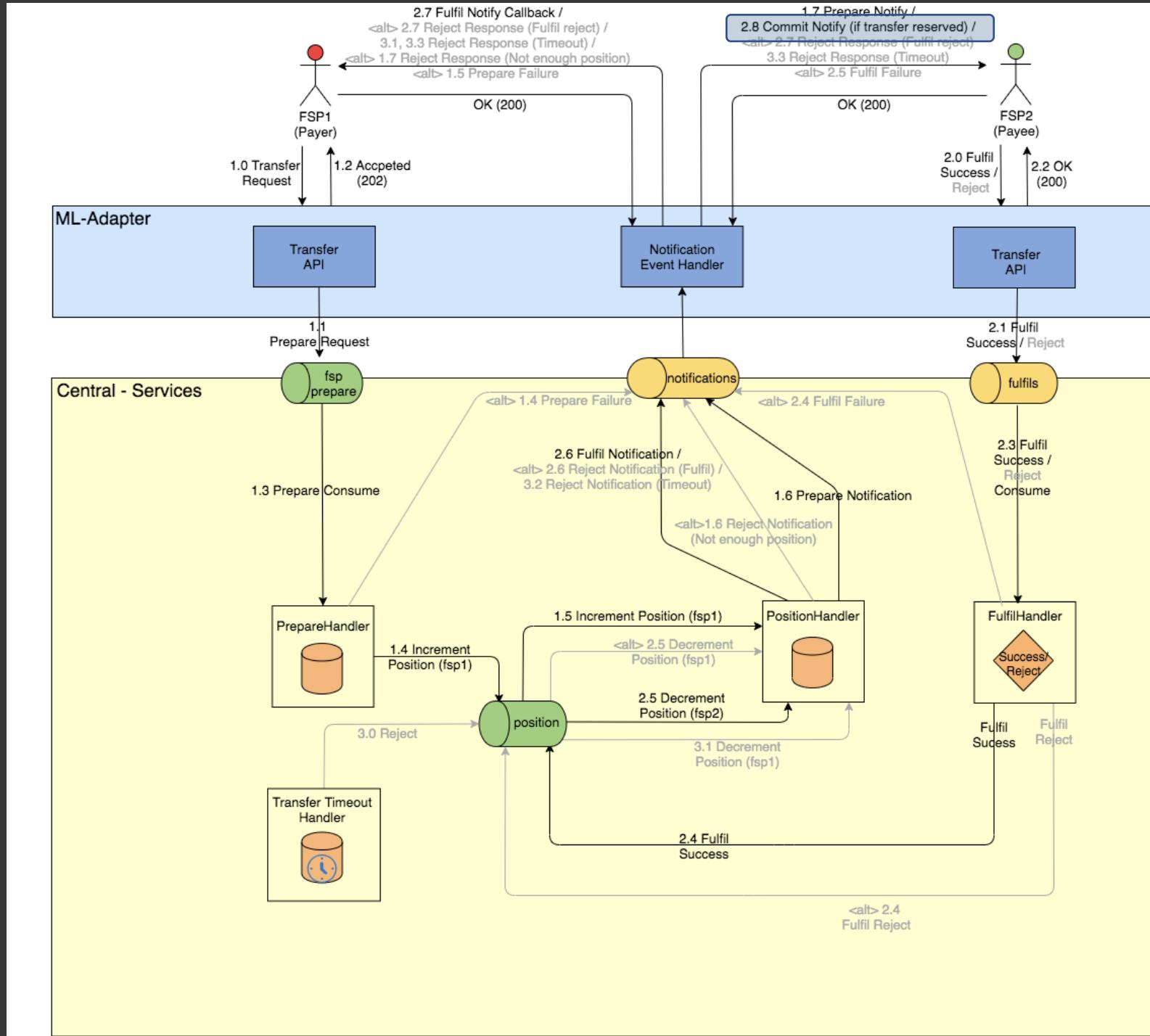
```
PUT /transfers/85feac2f-39b2-491b-817e-4a03203d4f14 HTTP/1.1
Content-Type: application/vnd.interoperability.transfers+json;version=1.1
Content-Length: 166
Date: Tue, 15 Oct 2019 08:14:02 GMT
FSPIOP-Source: DFSP2
FSPIOP-Destination: DFSP1
FSPIOP-Signature: {"signature": "YXBwbGlj...JzK2pzb2"}
{
  "fulfilment": "mhPUT9ZAwd-BXLfeSd7-YPh46rBWRNBiTCSWjpk90s",
  "transferState": "RESERVED",
}
```

## Supporting updates

1. Simulator(s)
2. sdk-scheme-adapter
3. Postman collections
4. Documentation

# PI-10: Transfers Architecture v1.1

Phase-4 PI-11



## FSPIOP API v1.1 Changes - 2: Use error callbacks for rejections

- [●] Clarify usage of ABORTED state in the Fulfil step of a transfer - [Story#1335](#)

### Behaviour in v1.0

Transfer could be aborted by two ways

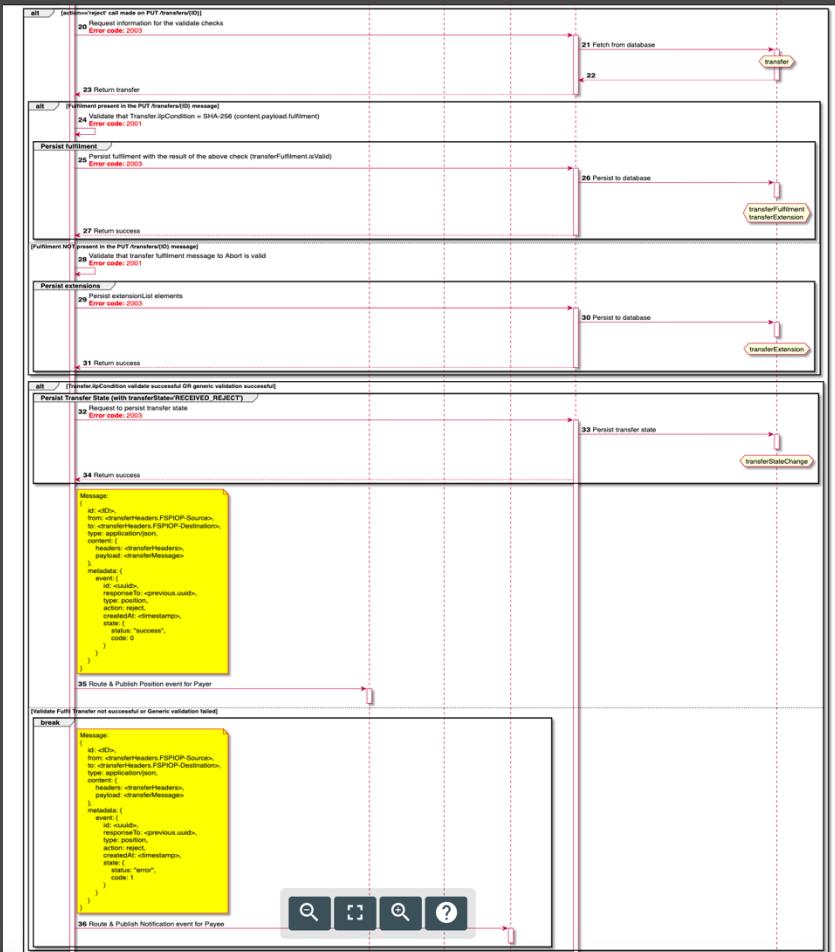
1. *PUT /transfers/{ID}* endpoint with 'transferState' as ABORTED.
2. *PUT /transfers/{ID}/error* endpoint with additional error information.

### Requested Behavior

FSPs to ONLY use *PUT /transfers/{ID}/error* to abort / reject transfers during a Fulfil step

# FSPIOP API v1.1 Changes - 2: Use error callbacks for rejections

[●] Clarify usage of ABORTED state in the Fulfil step of a transfer - [Story#1335](#)



## Updates

1. Central-ledger
2. Postman golden path collection
3. Documentation

## FSPIOP API v1.1 Changes - 3: ExtensionList element to */participants* requests

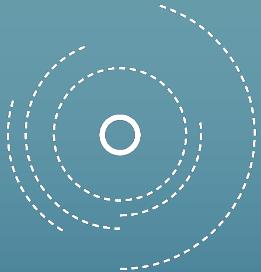
- [●] Support for extensionList element in 'partyIdInfo' Object - [Story#1336](#)

### Behaviour in v1.0

'*partyIdInfo*' object and POST */participants/{TYPE}/{ID}*,  
*/participants/{TYPE}/{ID}/{SubID}* calls did not have an *ExtensionList* element

### Requested Behavior

Additional data is required for some existing data items in order to meet the requirements of individual implementations. An optional *ExtensionList* element has been added to '*partyIdInfo*' object and POST */participants/{TYPE}/{ID}*, */participants/{TYPE}/{ID}/{SubID}* calls



mojaloop

## PI-10 Maintenance

---

Releases, Bug Fixes & Upgrades

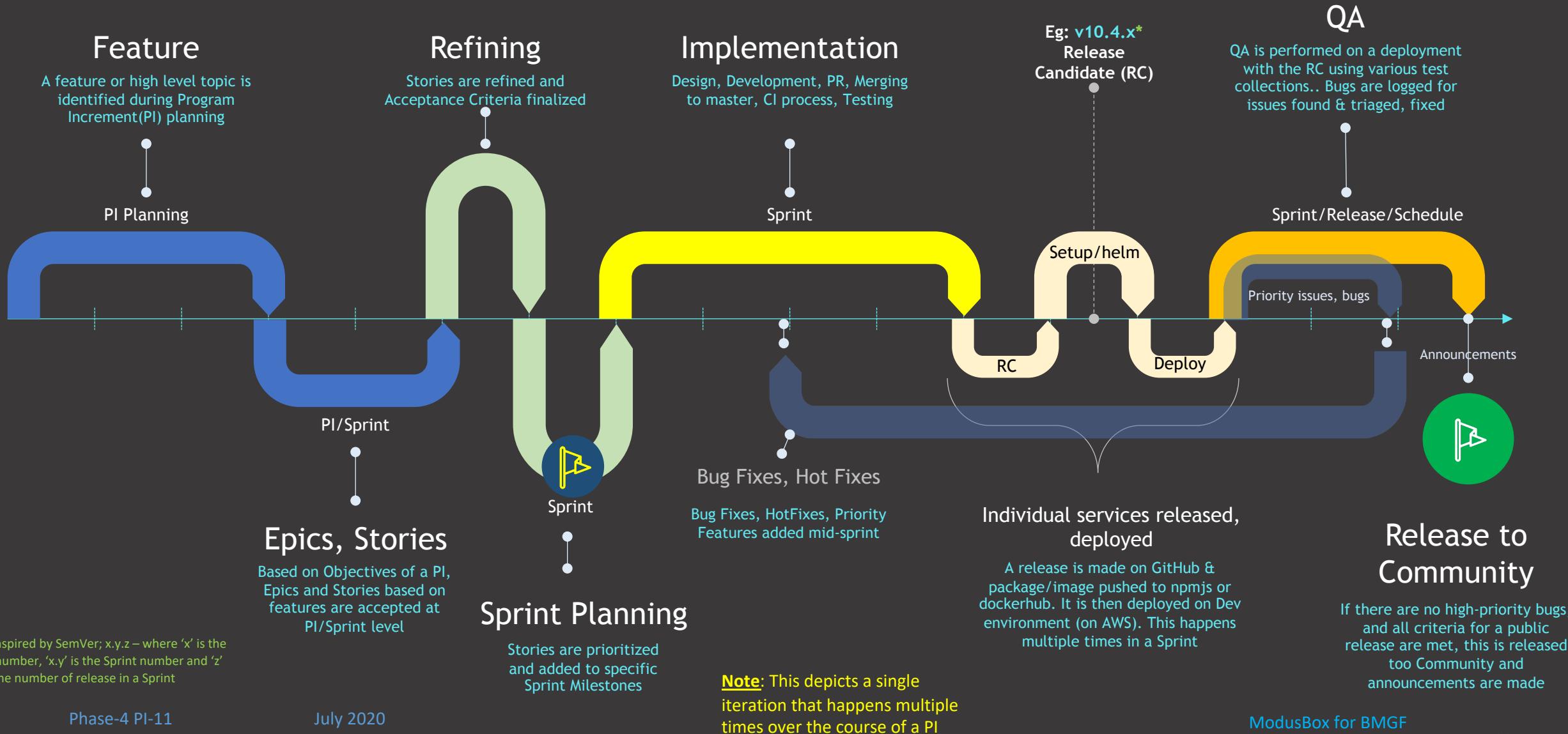
# PI-10: Maintenance Updates

1. Helm Releases, maintenance
2. Support for accented characters in names
3. JWS signing enabled for outward facing interfaces (quoting service, ml-api-adapter, ALS)
4. QA, Bug Fixes
5. On-us transfers support

# PI-10: Helm Updates

1. Helm Release [v10.1.0\\*](#)
2. Helm Release [v10.4.0](#)
3. Current charts support Helm v3 and v2
4. Backwards compatibility of charts with v2 verified on ML OSS environments

# ML OSS: (Helm) Release Mechanism



# PI-10: Accented characters in names

1. Allowing names with accent characters in names
  - a. Updated regex to attempt match any letter/number from any language script
  - b. Current issues with some scripts investigating alternative libraries for regex validation
2. Library: *hapi-openapi -> openapi-backend*
  - a. Allows for more control over validations libraries
  - b. OpenAPI 3.0 compatible
  - c. Works with our current libraries and plugins

# QA Updates – Bug Fixes

## Major Bug Fixes in PI – 10 (*A total of ~20 bugs fixed*)

#1331 – Unsupported HTTP Method for all resources – fix for 405 status code

#1168 – Timeout error in quoting service

#1378 - Extension List missing when unsupported version requested for services

#1381 - PAYEE flag disables notification to PAYER

#1404 - Accept header is sent in the callbacks for Post/Quotes

#1408 - FSPIOP-URI header missing in PUT /transfers callback

#1412 - Span finished. no further actions allowed ALS

# PI-10 QA Updates – Test collections

## Postman collections

1. [Mojaloop Simulator](#) collections are used for validation, referenced as part of Helm release with Mojaloop-simulator
2. *JWS signing enabled for Switch services, postman collections, mojasims*
3. [Legacy simulator collections](#) are used for validation, referenced as part of Helm releases as well
4. Authorizations based tests are included in the golden path test.
5. [Bulk transfers collection](#) is provided and used for validating Bulk transfers functionality

## Testing toolkit

1. Testing toolkit used to validate and demonstrate
  - a. Hub implementation in generalOn-us transfers, v1.1 changes, Bulk transfers

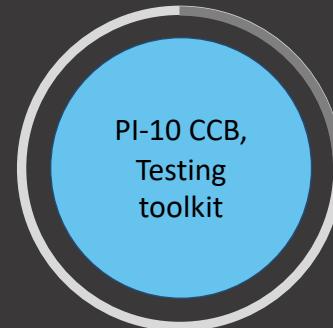
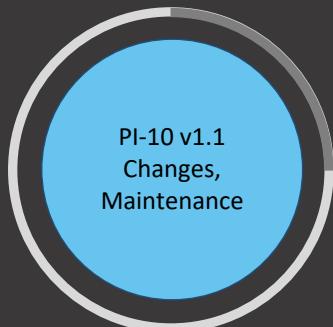
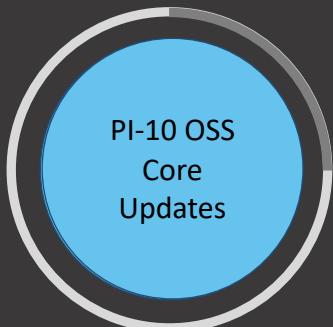
# QA Roadmap for PI-11

## QA Stream for PI 11 – If prioritized as a high-level item

1. Update test collections to correlate with Sequence diagrams for functionality; Update sequence diagrams as needed to add sufficient detail.
2. Improve coverage on features, API Services (double number of tests / coverage)
3. Streamlining non-golden path collections
4. Improve reliability of tests (setting correct timeouts, using variables correctly)
5. Using testing toolkit to provide at least as much as the current test coverage
6. Port the delta from Legacy Simulator collection to MojaSim collection
7. Cron job regression testing setup with multiple collections (Legacy, MojaSim, Bulk)

# Mojaloop PI-11

Phase-4 Going Live!



# PI-10: Bulk Transfers standardization

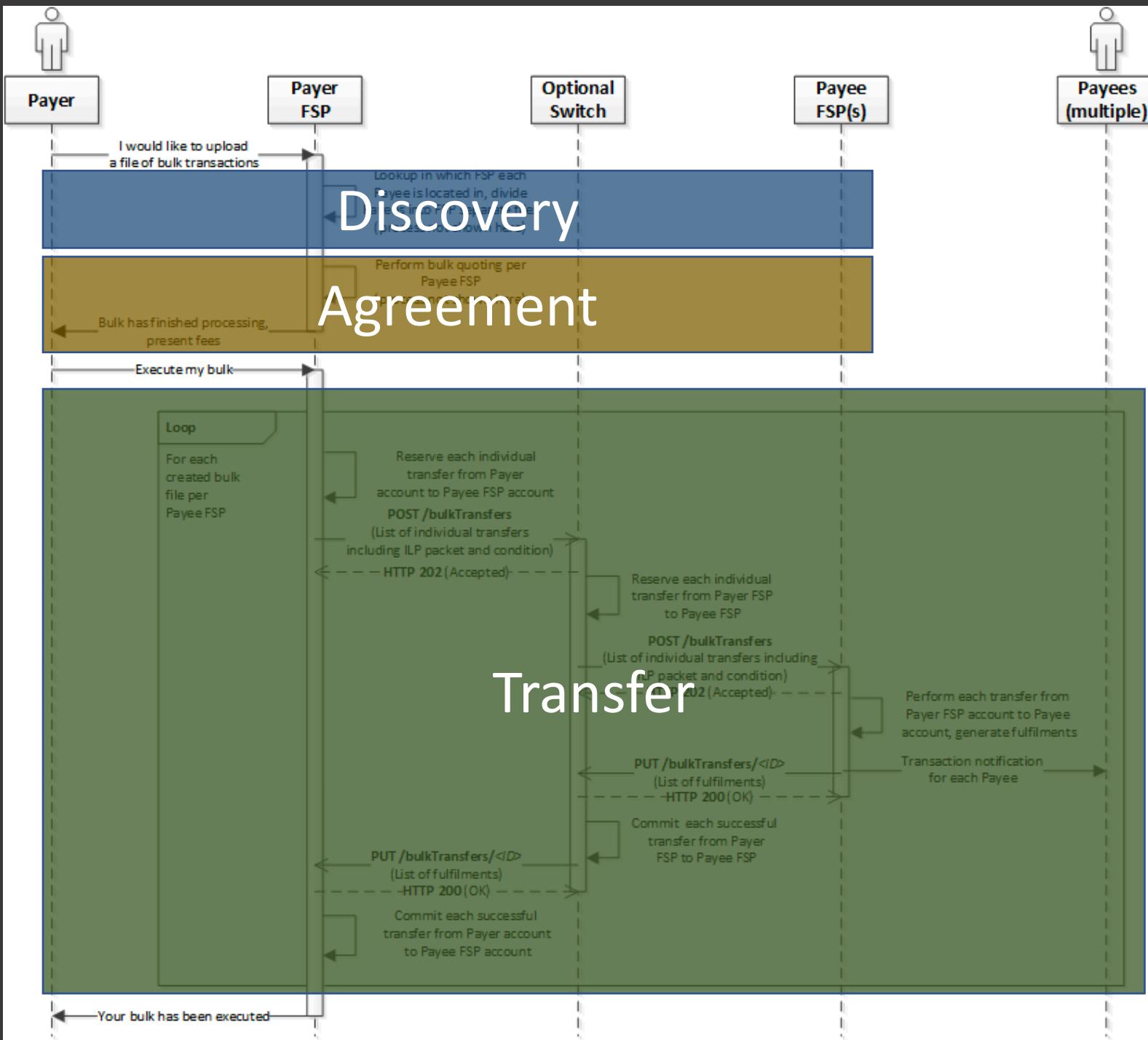
1. Bulk Transfers
  - a. *PoC done in PI-7*
  - b. PI-9: Happy Path standardized with tests and Simulator support, Postman tests
2. Standardization in PI-10
  - a. Support for error scenarios: *PUT bulkTransfers/{ID}/error*
  - b. *GET bulkTransfers/{ID}* functionality with individual responses for Payer and Payee FSPs
  - c. *POST /bulkQuotes* and *PUT bulkQuotes/{ID}* implemented now
  - d. Support added in Simulators
  - e. End to end tests with simulators
3. *Bulk transfer design and processing*

# PI-10: Bulk Transfers Steps

## Bulk Transfer Steps v1.0

Payer FSP performs

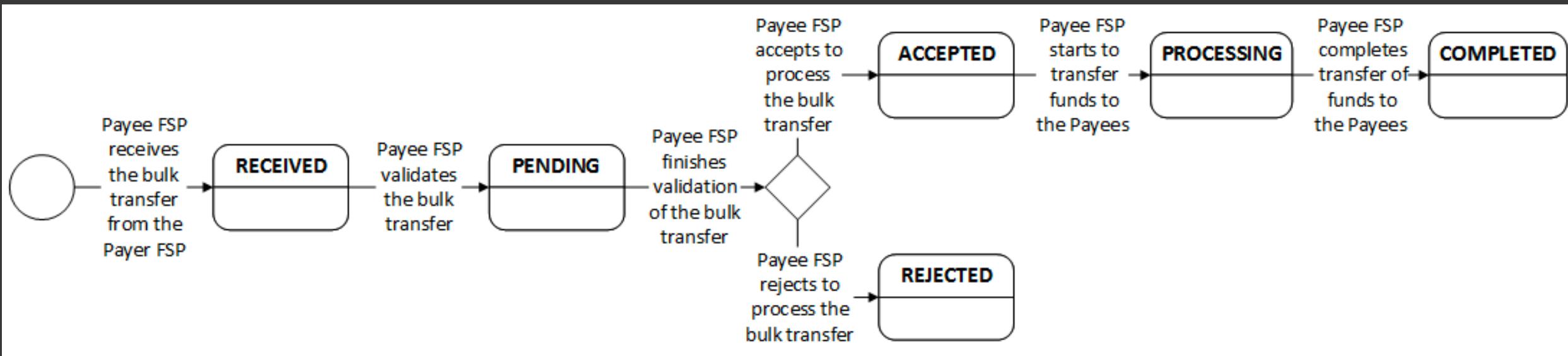
1. **Lookup** the Payee FSP for individual transfers in the bulk, using the API Resource [`/participants`](#)
2. Perform the **bulk quote** process using the API Resource [`/bulkQuotes`](#). The bulk quote callback should contain the required ILP Packets and conditions needed to perform each transfer.
3. Perform **bulk transfer** process using [`POST /bulkTransfers`](#). This performs each hop-to-hop transfer and the end-to-end financial transaction.



# PI-10 Bulk Transfers: Bulk handlers

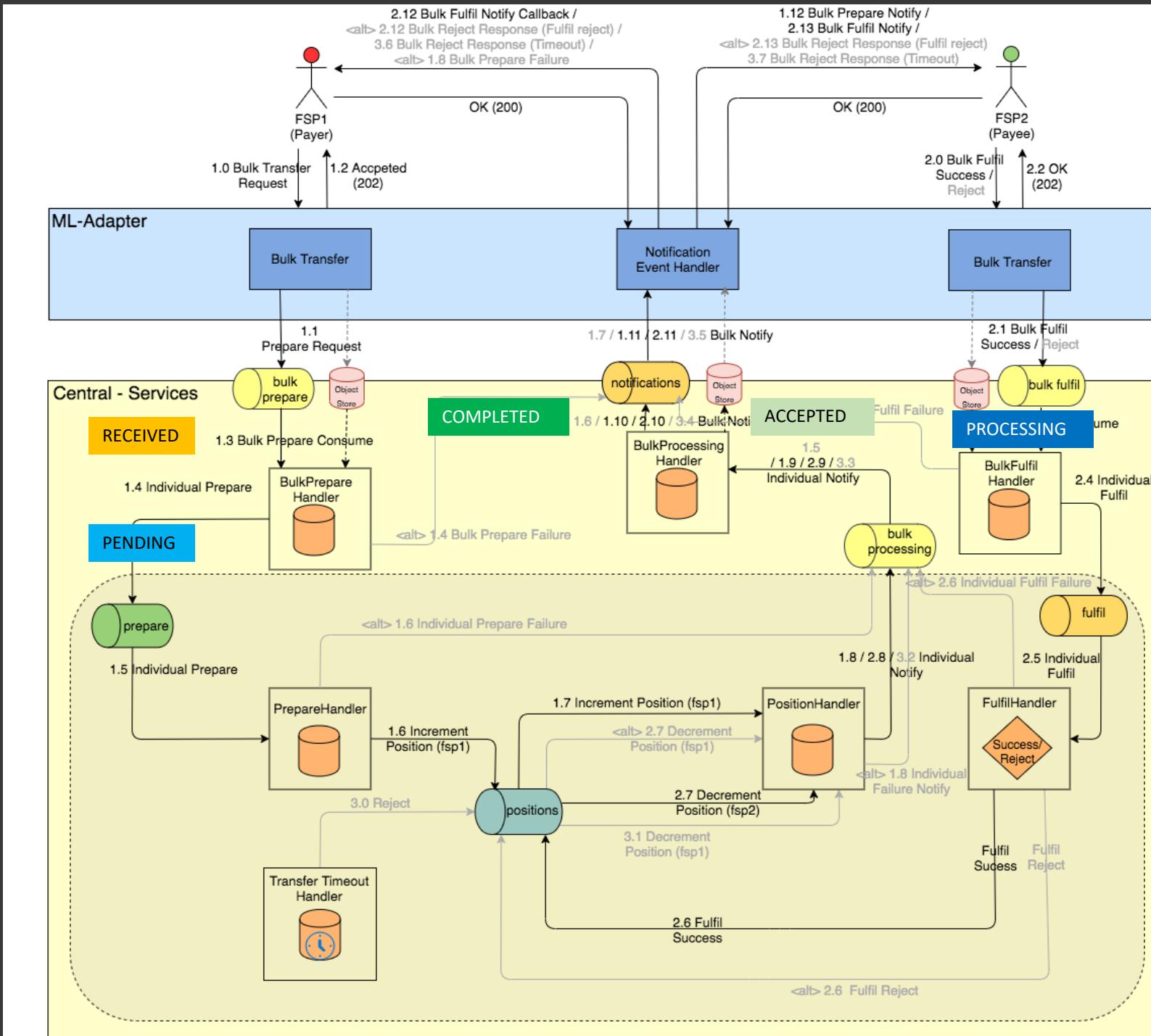
1. Bulk API Adapter
2. Bulk Prepare Handler
3. Bulk Fulfil Handler
4. Bulk Processing Handler
5. Bulk Get Handler

## PI-10: Bulk Transfers States



Pl-10: Bulk Transfers Architecture v1.0

Phase-4 PI-11

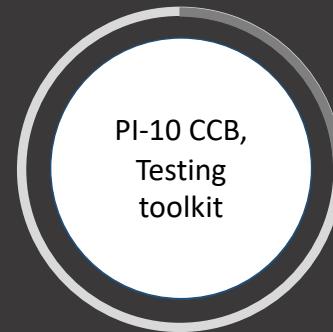
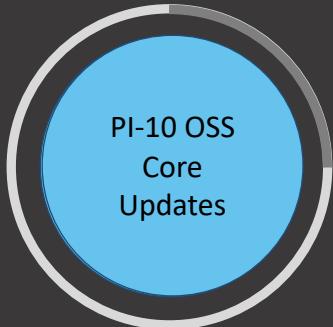


# Bulk Transfers: Roadmap

1. Bulk Lookup
  - a. If prioritized, design a solution (API level, sequence)
  - b. Propose to the CCB as a Change request
2. Bulk make?
  - a. An FSP provides a bulk with individual transfers at one go
  - b. The Switch takes up the responsibility for all three steps;
    - a. Breaking it out into smaller bulks based FSPs after Lookup
    - b. Perform bulk quoting (With option to aggregate them and send to Payer FSP)
    - c. Perform bulk transfer
3. Configurable options such as all or nothing, etc.

# Mojaloop PI-11

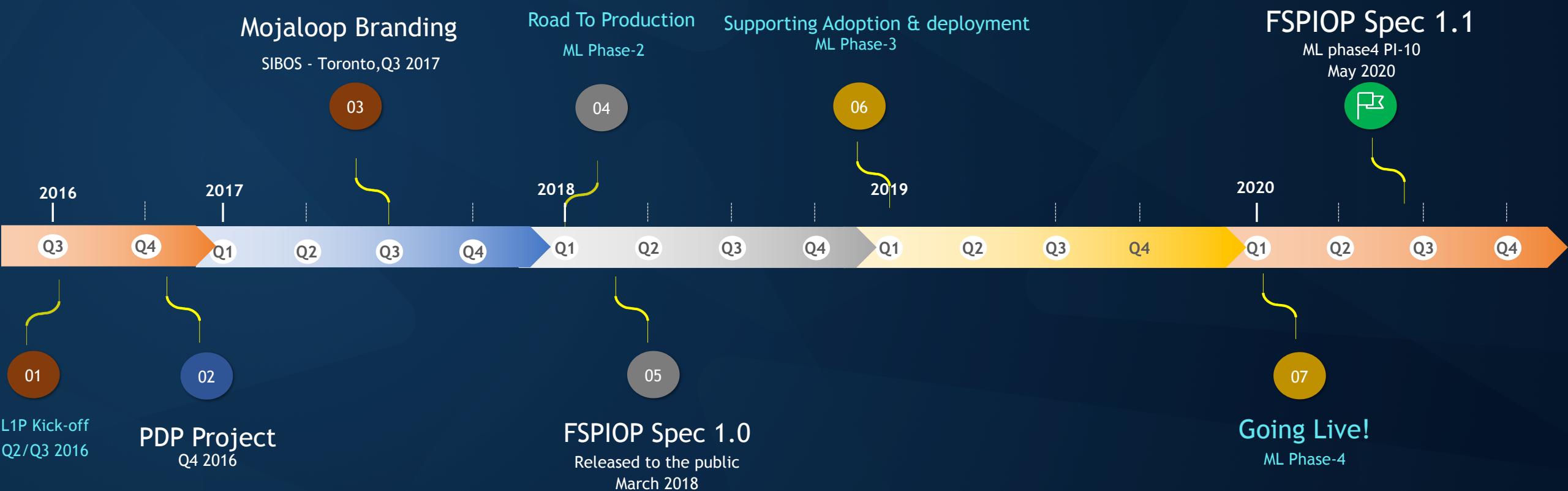
Phase-4 Going Live!



# ML OSS Community: Change Control Board

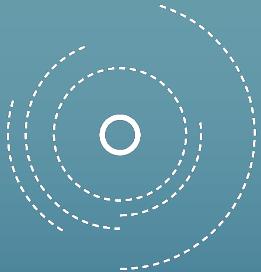
1. Goals, purpose – ML FSPIOP API
2. Current Membership
  - i. BMGF (Matt Bohan, Miller Abel)
  - ii. Ericsson (Henrik Karlsson)
  - iii. Huawei (Chen Hill)
  - iv. Mahindra Comviva (Ritvik Sinha)
  - v. ModusBox [Non-voting] (Michael Richards, Sam Kummerly)
  - vi. Mowali (John Mark Ssebunya)
  - vii. Telepin (RJ Wilson)
  - viii. BoT TIPS (Mutashobya Mushumbusi)
3. Frequency of meetings, boards used
4. Change requests, Solution proposals, Bugs

# Mojaloop FSPIOP API: Evolution



# CCB – ML FSPIOP API: v1.1 changes

1. Current status:
  - a. FSPIOP API v1.1 published in May 2020
  - b. Changes in pipeline for v2.0 (PISP, Cross network/currency, other breaking changes)
  - c. Separate APIs and boards proposal
2. Overview of changes in v1.1
  - a. List of changes: <https://github.com/mojaloop/mojaloop-specification/issues/52>
  - b. Option to notify completion of a transfer to a Payee
  - c. Updating description, examples to address inaccuracies and omissions
  - d. Clarify usage of ABORTED state in the Fulfil step of a transfer
  - e. Describe quote rejection flow
  - f. Add notes to indicate the case-insensitive nature of HTTP headers
  - g. Clarify usage of FSPIOP-Destination header
  - h. Adding ExtensionList element to several items in the data model



mojaloop

# Mojaloop Testing Toolkit

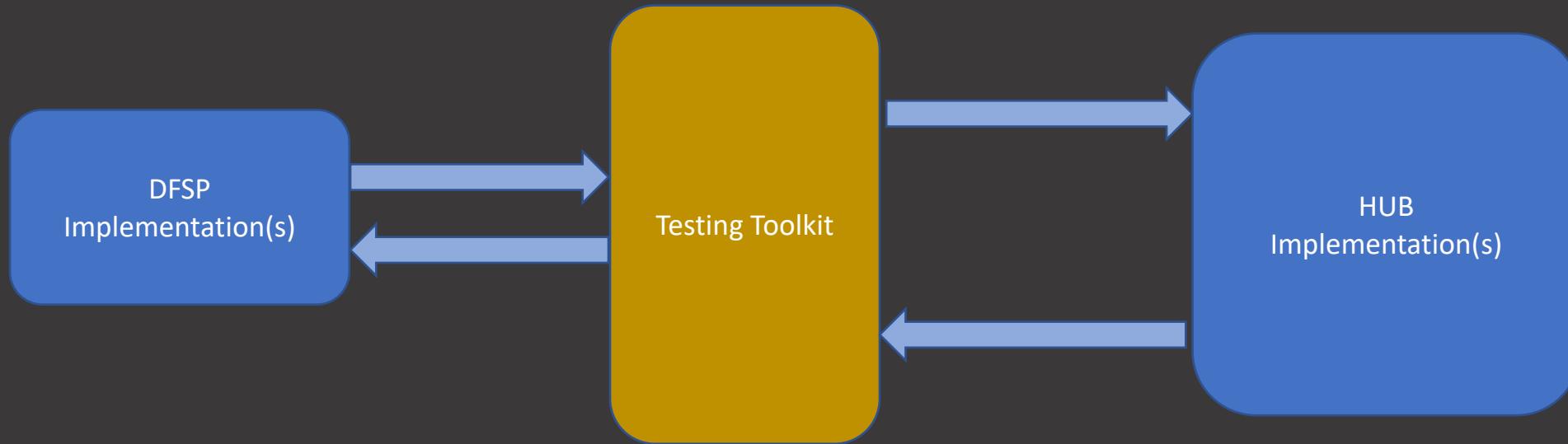
---

Features, Improvements & Community Support

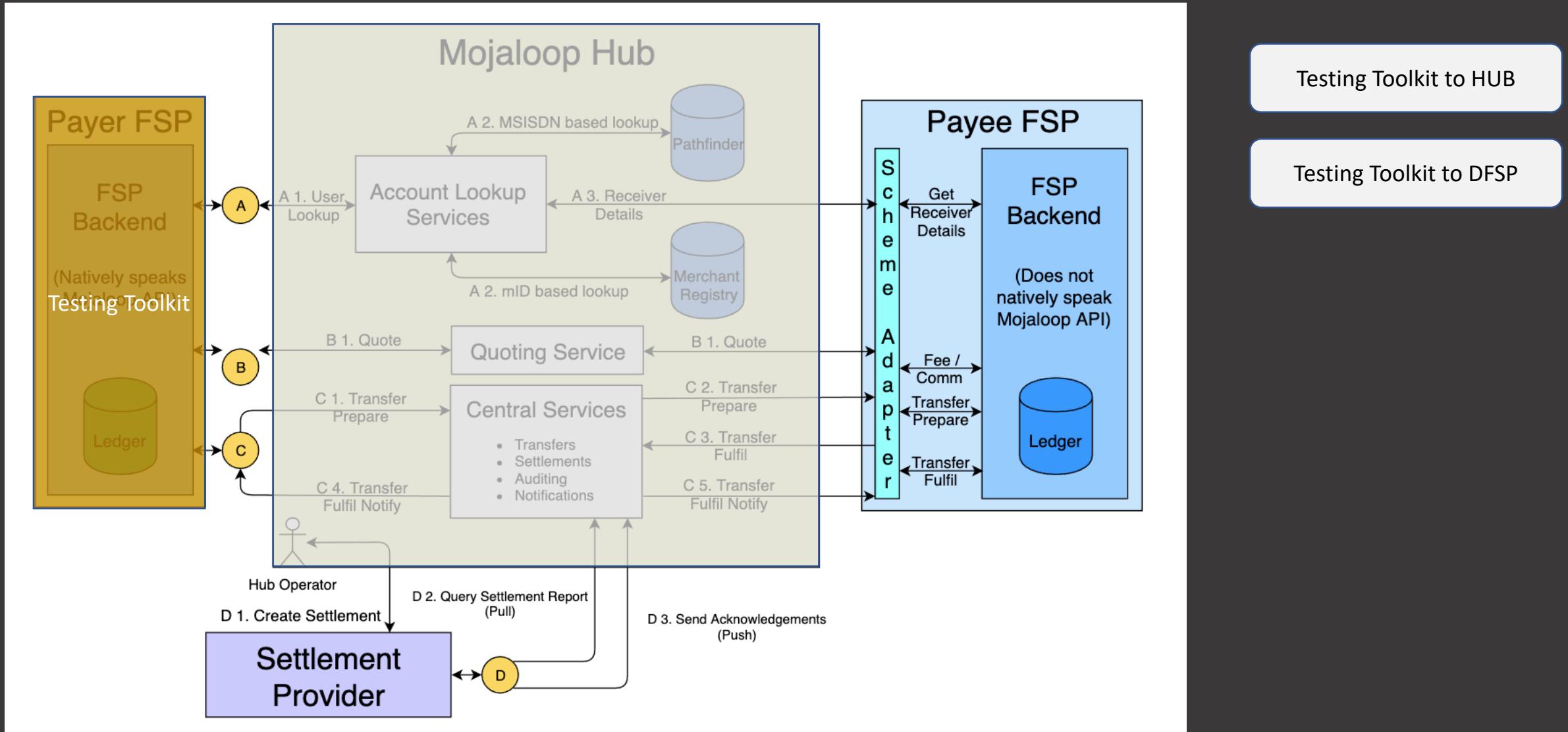
# Testing toolkit: Goals

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

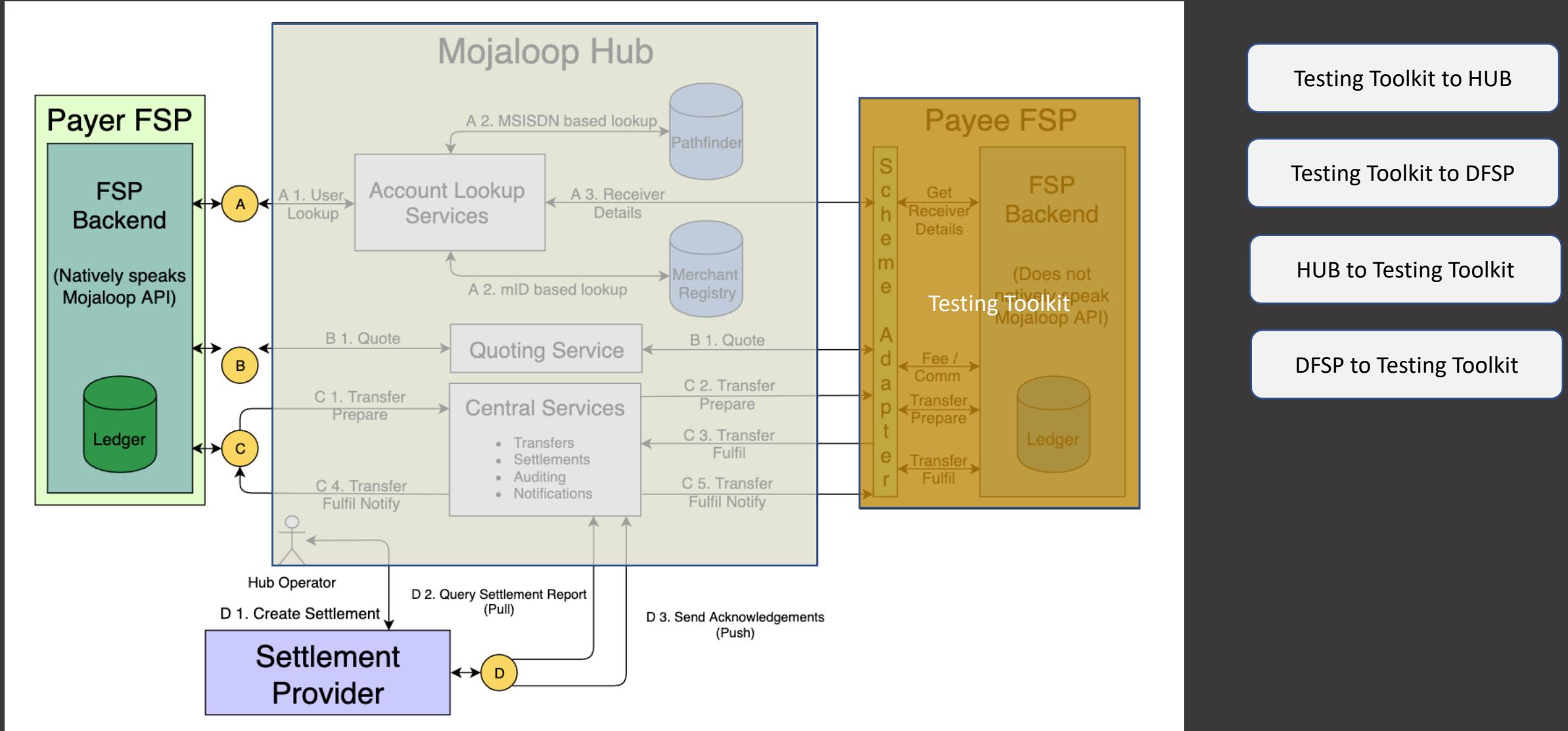
# What is the Testing Toolkit?



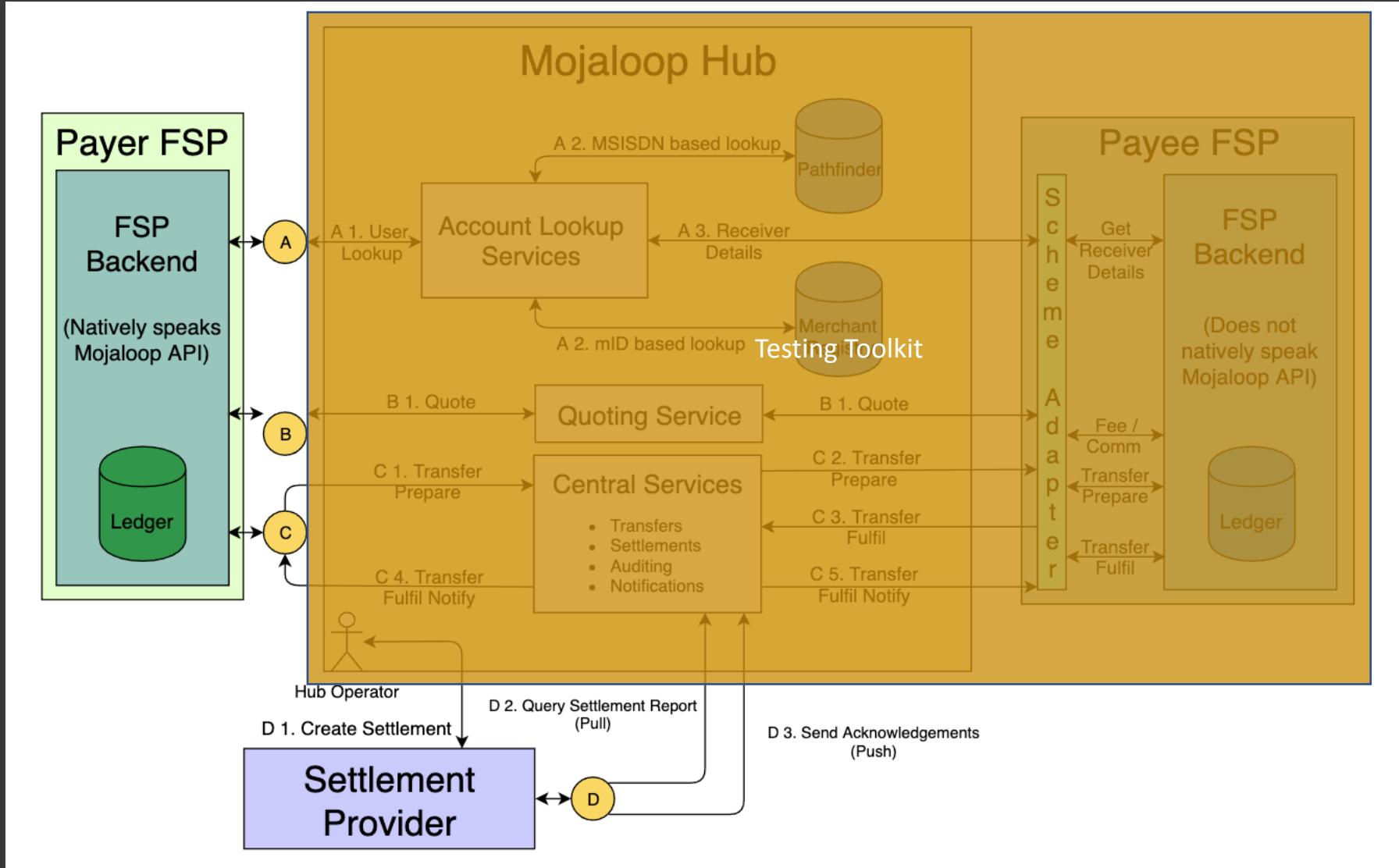
# Where does it fit?



# Where does it fit?



# 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

# PI-10 Toolkit Features

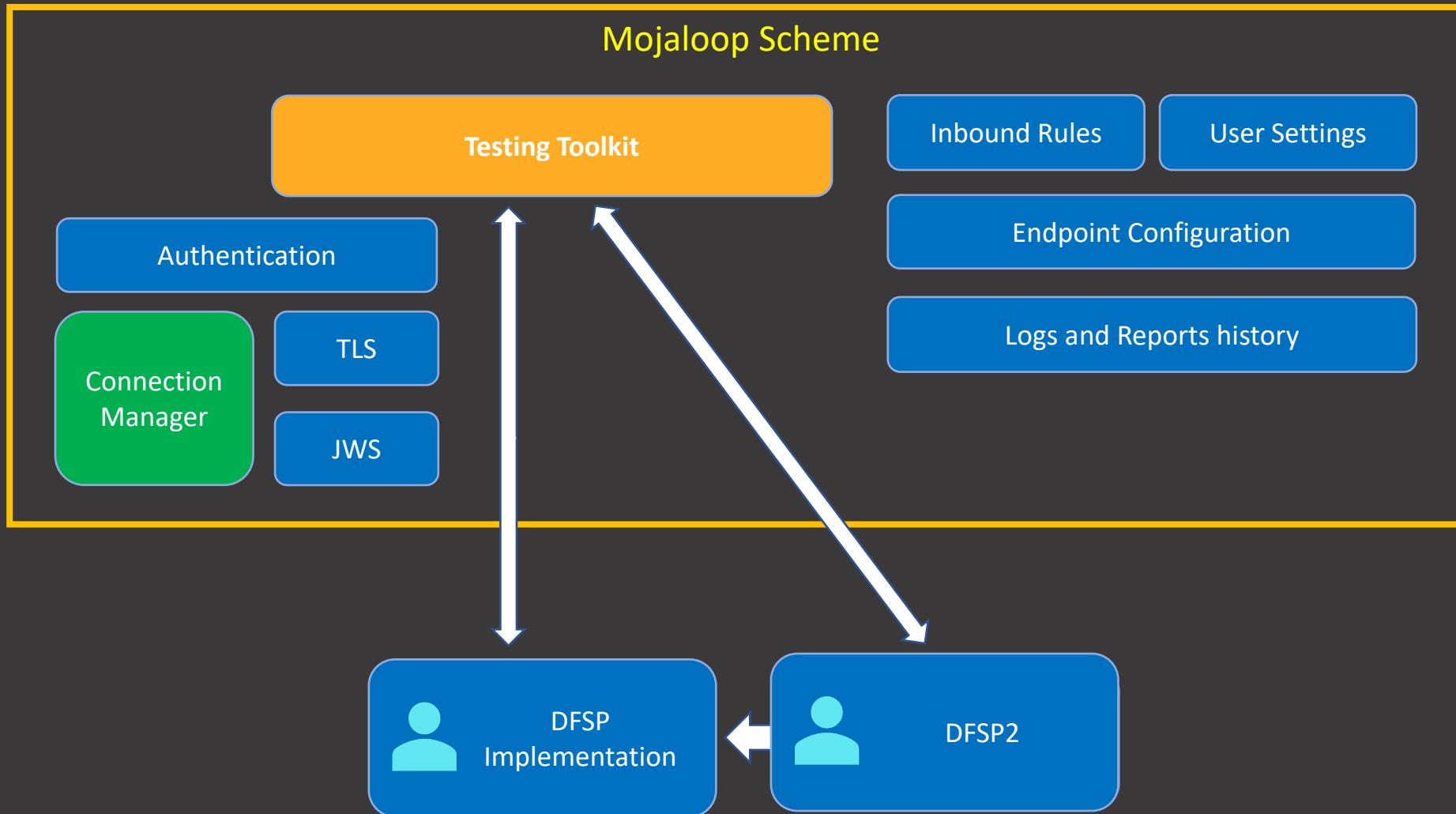
1. User Interface for QA / Product / Business users
2. Version validation and negotiation, schema validation & additional validation
3. Dynamic callback and error callback generation based on rules
4. Initiation of use cases (outbound) - Assertions and report generation
5. Simultaneous support for multiple 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: Enhancements in PI-10

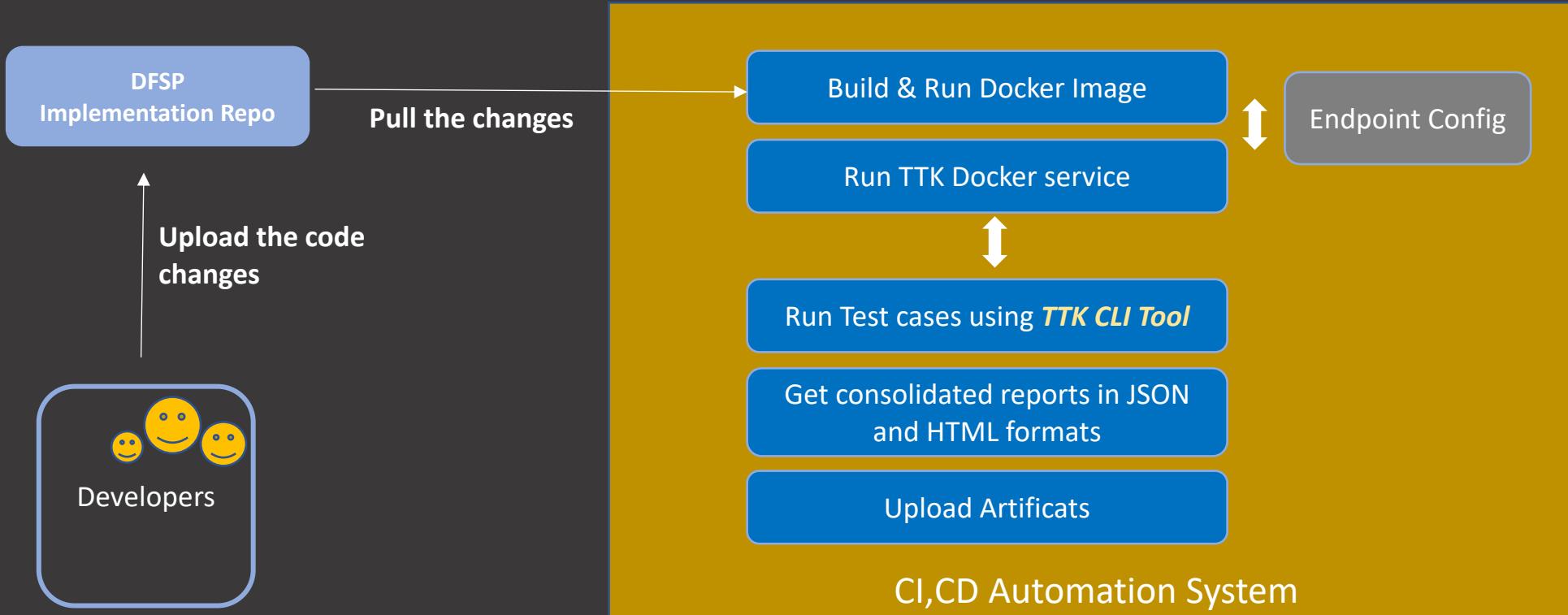
1. Expand support for all the resources in the FSPIOP API
2. Command line client (CLI)
3. Import and Export capability of rules & settings
4. Improved Test case execution
  - a. Separate test case sets for Hub & FSPs
  - b. Test Reports and exporting options
  - c. Curl (code) option for outbound messages
  - d. Scripting capability for outbound request editor
5. Helm chart for testing toolkit
6. Verifying condition and fulfilment
7. Prioritize rules & delay simulation
8. Caching functionality in rules
9. QA: >98% test coverage
10. Basic hosting capability

# Testing toolkit: Demo

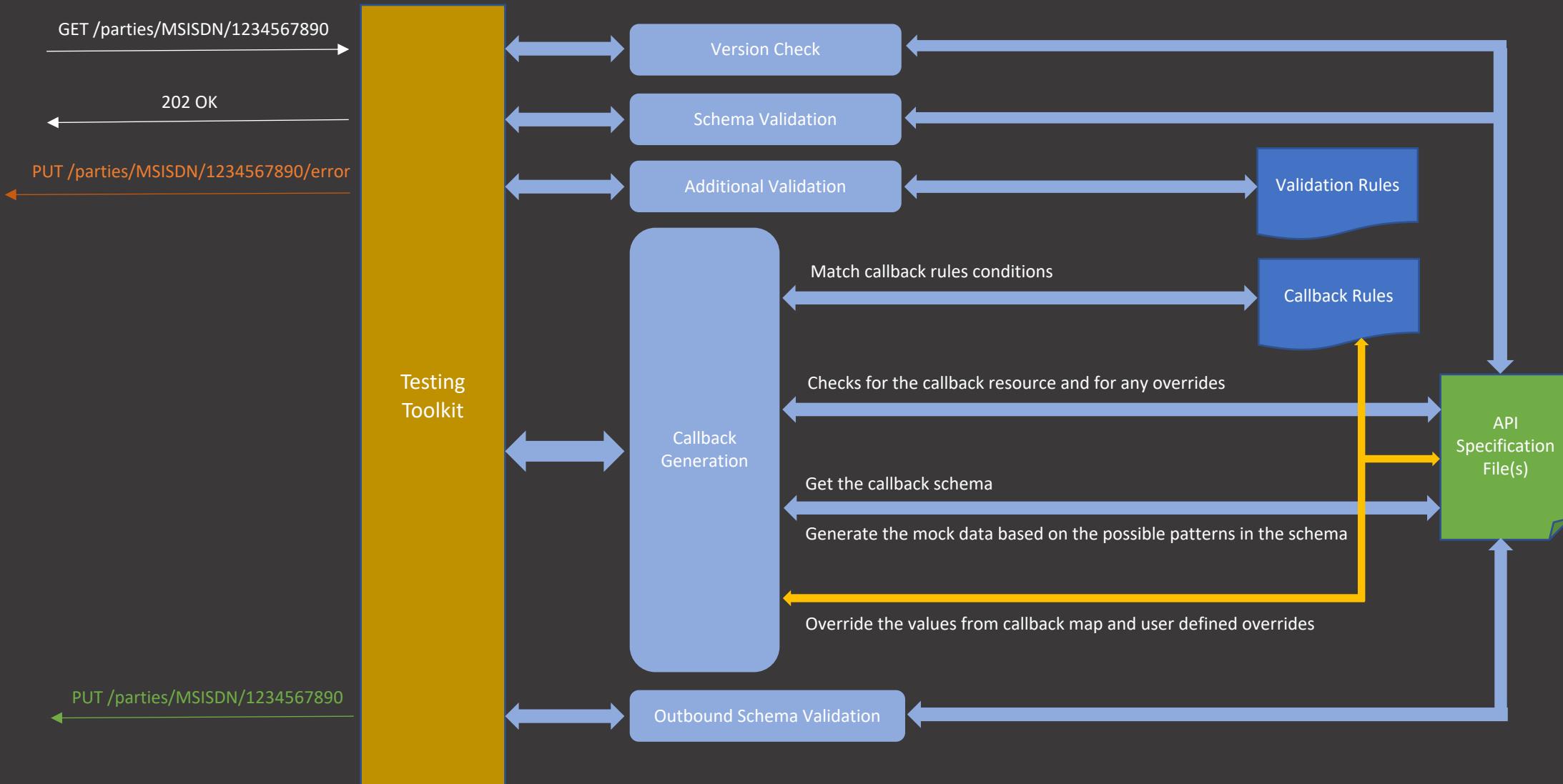
# Adding Capability for Hosted Solution



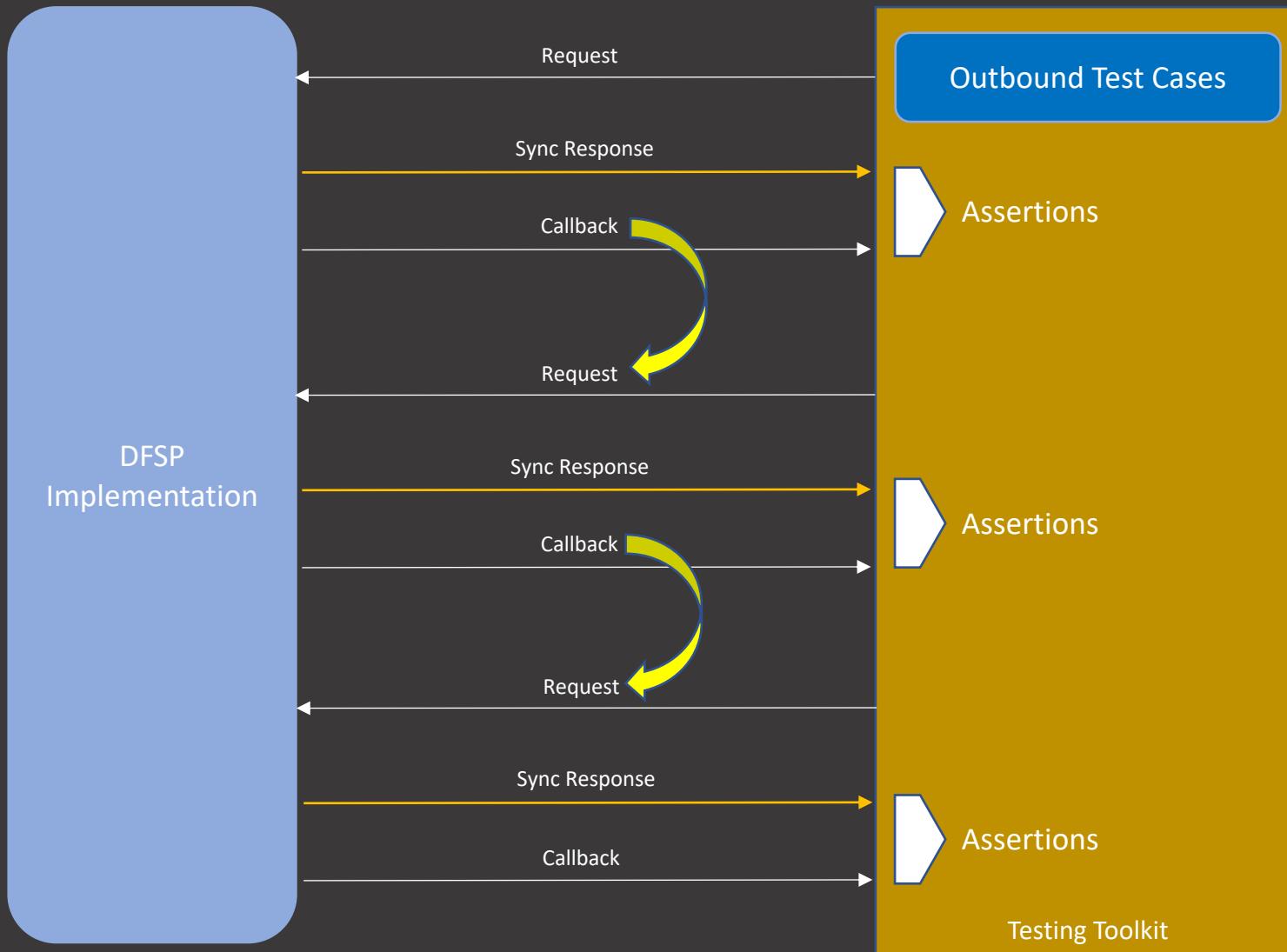
# Integrating TTK into DFSP's CI, CD



## How Testing Toolkit Works - Incoming requests



# How Testing Toolkit Works - Test Case Initiation and Assertions

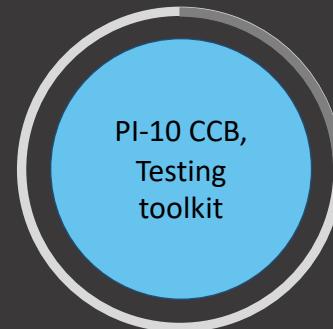
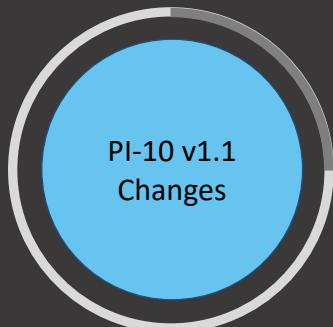
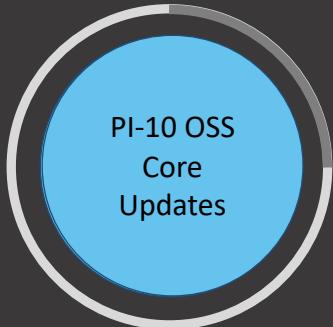


# Testing toolkit: Roadmap for PI-11

1. Enable easy and ***secure hosting*** of testing toolkit and ensure ***scaling*** with separation of – 1) event handling, 2) Rules, 3) User Settings and 4) Logs & Reports storage
2. Play a “**Hub**” role, so that FSPs can test Payer and Payee FSP side functionality simultaneously
3. ***Postman collection conversion*** to TTK test cases
4. Improve coverage to test Hubs
5. Templates, Rules for all use cases from the Mojaloop Specification
6. Performance Tuning to withstand load (not for perf testing)
7. Event framework

# Mojaloop PI-11

Phase-4 Going Live!



## Phase-4 PI-10: Objectives - 1

- 1. Bulk Payments:** Deliver a standardized bulk transfer service for golden path which includes error & negative scenarios
- 2. FSPIOP v1.1 changes:** Enhance the ML OSS implementation to support v1.1 changes including other maintenance items.
- 3. Testing Toolkit:** Stand-alone version that can be used with features to export, import rules and settings is made available, with capability for hosting the toolkit.
- 4. Settlement:** A flexible and configurable settlement service.

## Phase-4 PI-10: Objectives - 2

5. **Cross Currency/Network:** Finalize a design for the Mojaloop to non-Mojaloop flow including API changes and getting buy in from CNPs and the CCB
6. **Performance:** Deliver productionized Mojaloop deployment using Kafka best-practices; Evaluate Event-Sourcing PoC and Optimize SQL report and enhanced dashboards
7. **Fraud & Risk Mgmt:** Define, investigate and validate a backlog and MVP for a FRM system/service against the APRICOT modelling for existing/prospect operators; and identify partners to build/implement a FRM system/service
8. **PISP:** Ability to do an end to end PISP transfer (from demo PISP) and present a well-defined relationship between USER, PISP, and their DFSP
9. **OSS onboarding:** Updated documentation and end to end review of the onboarding process
10. **Versioning:** Propose Versioning standards for Mojaloop OSS releases focusing on implementation, deployments that include details on OSS support for LTS releases, branching & upgrade strategies.

## Phase-4 PI-11: Roadmap – To be prioritized

1. *Forensic Logging, Secure auditing*
2. Performance: PoC learnings implementation
3. Ticketing system
4. Portals for Onboarding & Managing DFSPs, etc
5. Rules API
6. Split Payment Capability (SNAPP)
7. Standardize operations (Admin) API
8. Streamline Testing & QA
9. ISO 20022 adaptation
22. Bulk Transfers extensions

## Phase-4: Roadmap – Ongoing

10. Code quality and Security
11. Cross Network / Currency (CNP/FXP)
12. Fraud Management
13. Leadership & Community Management (events, governance, communication, tools, etc.)
14. LPS Adapter Enhancements and future Use Cases (ATM, POS)
15. Payment Initiation Service Provider – Solution
16. Payment Hub (MIFOS)
17. Settlement v2
18. Testing tool-kit
19. Versioning standards, version maintenance
20. Web Payments (Coil)
21. GSMA Lab

# Streamlining Opportunities in Mojaloop

For improved sharing and transparency

Sam Kummary, Sankar Ramakrishnan

# Opportunities to Streamline



ZenHub

## Backlog Management

- Backlog structure

- Are we interchangeably using Feature, Use cases, Journeys?
- Do we have a definitive list of features for everyone to refer to?

## Product Management

- Common / consistent understanding of Product Feature
- Well defined set of personas

Are we converging Definition of Done for a story and the Business Acceptance Criteria?

## Execution

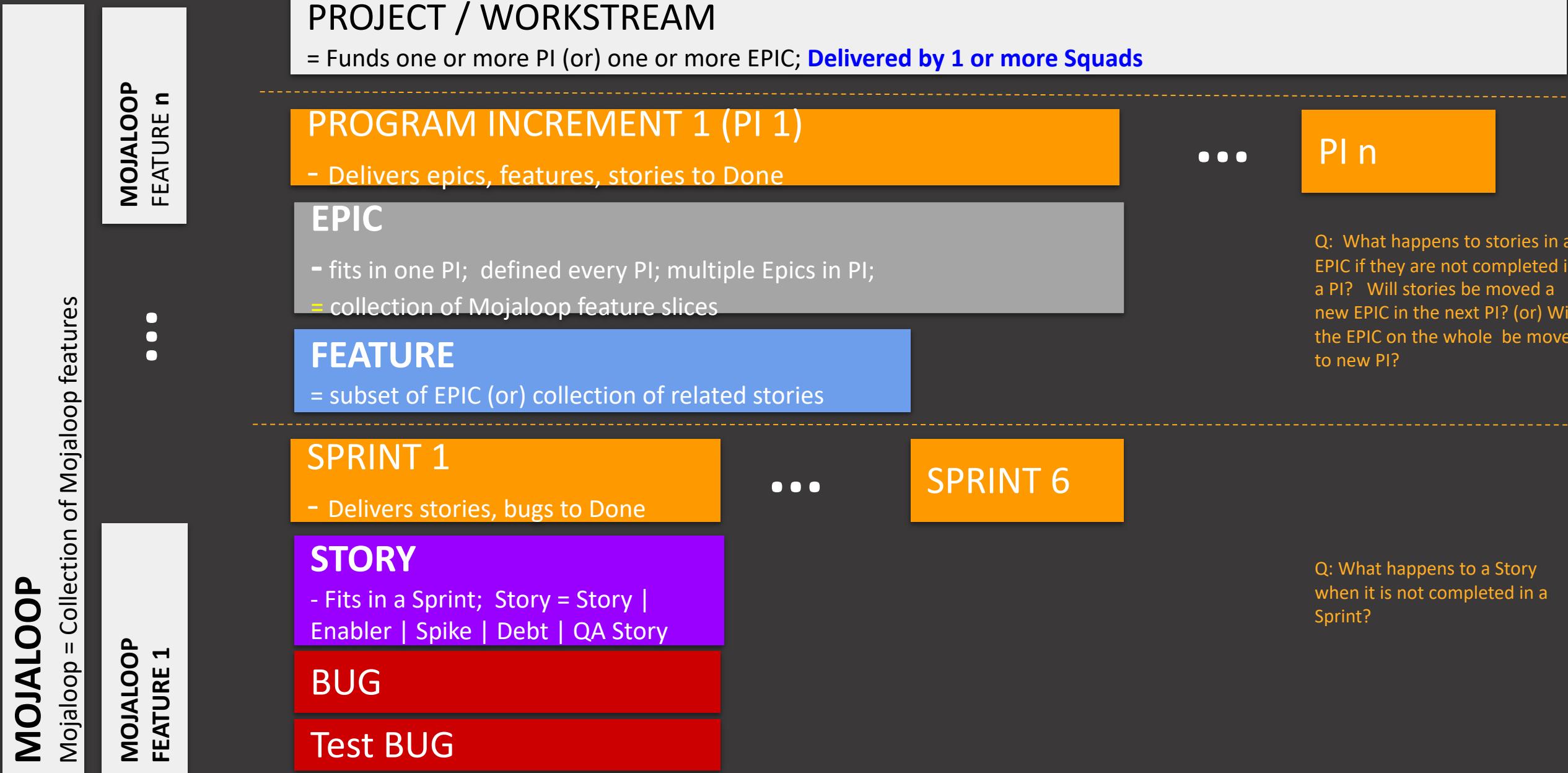
- Acceptance Criteria / Definition of Done for Epics, Stories
- Usage of Release Numbers
- Operational efficiency / simplicity

Dimensions: Program Increment Objectives, Workstream / Project Envelops, Epics, Sprints

## Transparency

- Further transparency on progress towards commitments & improvements

# AS-IS: ML BACKLOG & PROCESS CONCEPTS



# AS-IS : Backlog

project-poc ★ ↴

Board Reports Roadmap Create... Edit Workspace Invite

View tutorials Shortcuts Open in web app Support and training Changelog

Repos (1/33) Labels Milestones Assignees Epics Releases Find Issues (f+i) New Issue +

Customize pipeline name and descriptions to match your team's workflow. Click on the ⓘ next to each pipeline to make changes. [Learn more](#) X

Add a Pipeline ...

	<b>Closed</b>	50+ Issues - 256 Story Points
	<b>Review/QA</b>	7 Issues - 16 Story Points
	<b>In Progress</b>	23 Issues - 123 Story Points
	<b>Sprint Backlog</b>	35 Issues - 103 Story Points
	<b>Blocked Stories</b>	7 Issues - 23 Story Points
	<b>Triage</b>	22 Issues - 80 Story Points
	<b>Performance-PIO</b>	13 Issues - 65 Story Points
	<b>PI10-backlog</b>	75 Issues - 245 Story Points
	<b>Product Backlog</b>	61 Issues - 237 Story Points
	<b>Product Epics</b>	28 Issues - 29 Story Points

# PROPOSED: ML BACKLOG & PROCESS STRUCTURE

## MOJALOOP PRODUCT

Mojaloop = Collection of Mojaloop features

### PRODUCT FEATURE EPIC

- defined for every feature; collection of stories that implement the feature

...

### PROGRAM INCREMENT 1 (PI 1)

- Delivers product features, epics, stories and bugs to Done

...

PI n

### PI OBJECTIVE EPIC

- Captures a PI Objective; A collection of PI Epics

### PI EPIC

- fits in one PI; defined for every PI; multiple Epics in PI;
- = collection of Mojaloop feature slices (or) collection of stories/bugs

### SPRINT 1

...

### SPRINT 6

- Delivers stories, bugs to Done

### STORY

- Fits in a Sprint; Story Feature (slice) | Enabler | Spike | Debt | QA Story

### BUG

### Test BUG

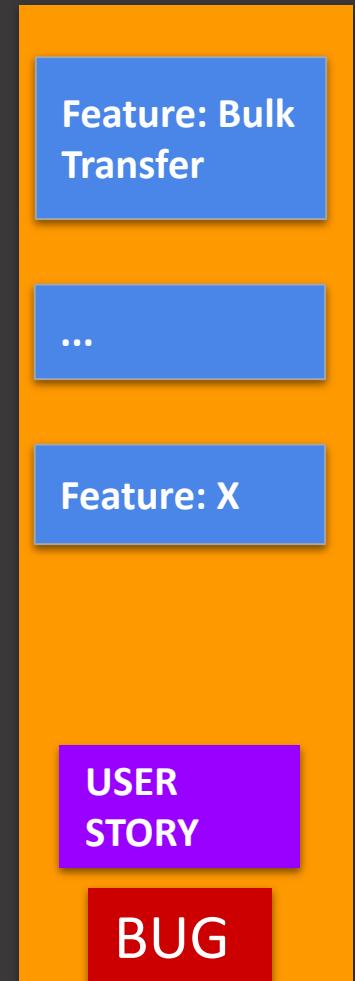
## PROJECT / WORKSTREAM

= Funds one or more PI (or) one or more EPICS;  
**Delivered by 1 or more Squads**

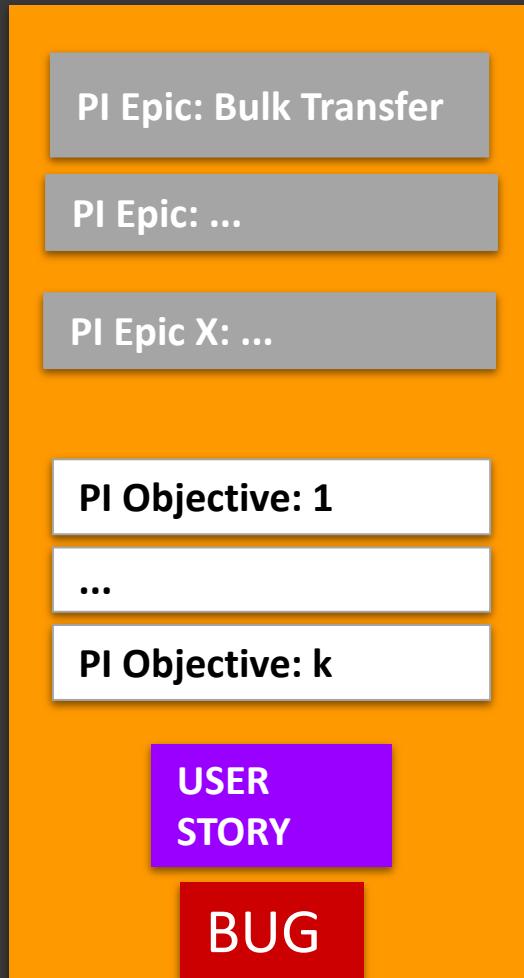
Label: Squads - 'oss-core', 'oss-pdp'

# SIMPLIFIED VIEW

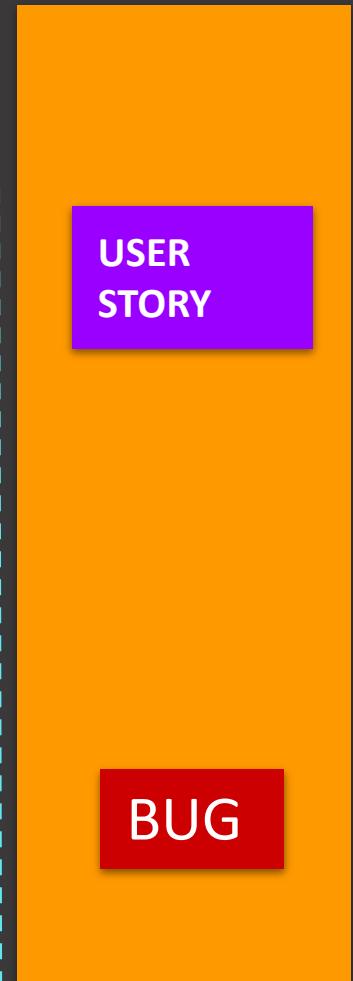
## PRODUCT BACKLOG



## PI BACKLOG



## SPRINT BACKLOG



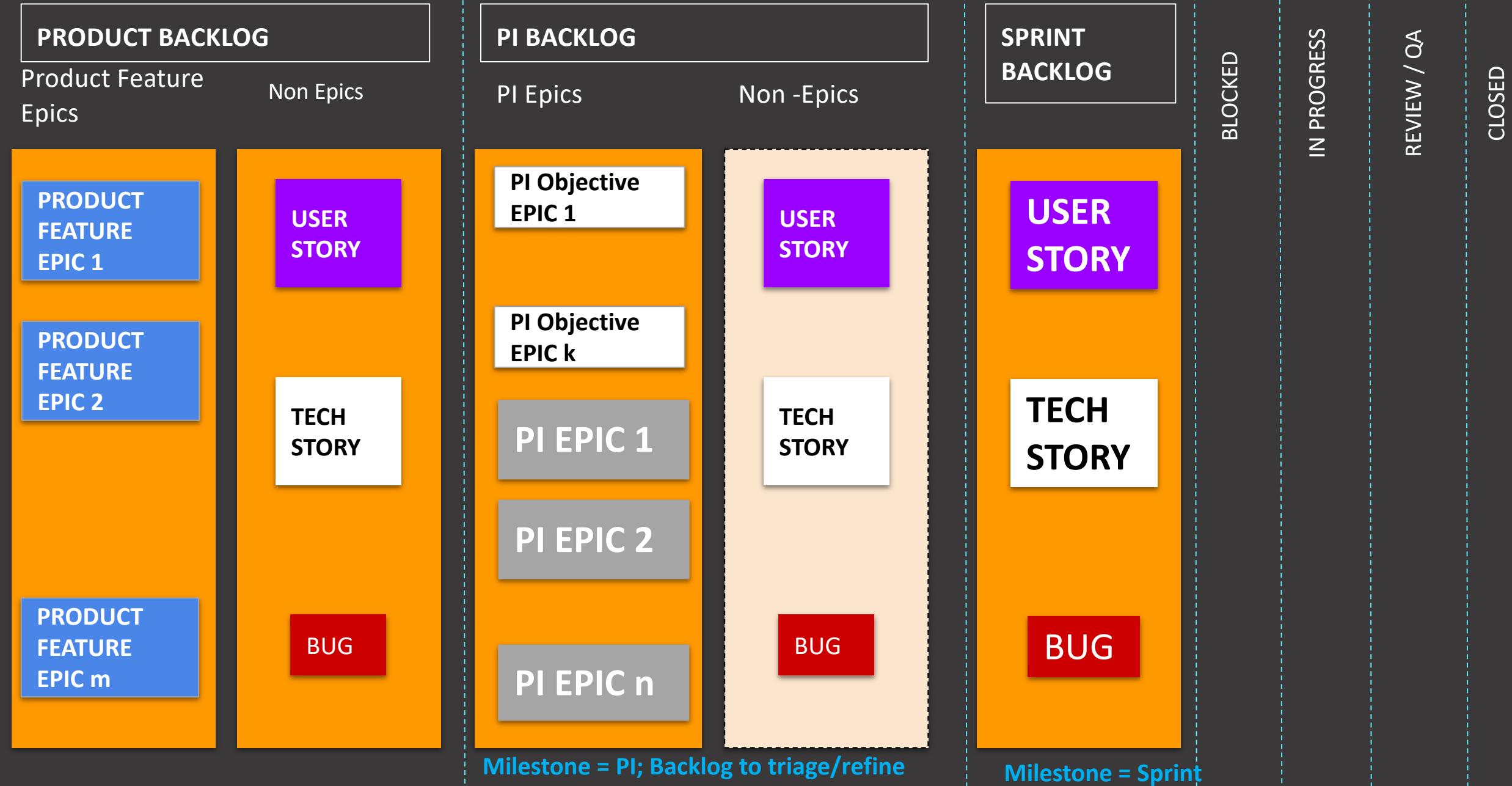
BLOCKED

IN PROGRESS

REVIEW / QA

CLOSED

# PROPOSED : SIMPLIFIED WORKFLOW



Zenhub limitations / feature:

A story can't have more than 1 milestone

A story can have more than 1 epic

## PROPOSED : BACKLOG GUIDELINES

**USER  
STORY**

**BUG**

**PI EPIC**

**In general**, every PI Epic will be associated with one Product Feature Epic

*Rarely, however, a dummy PI specific EPIC called PI-x-sundry may be created to hold issues that are :*

- either not directly related to a product feature
- Or only has a handful of disparate stories from multiple product features

**PI OBJECTIVE EPIC**

**Mandated tags for issues in Sprint Backlog**

Epics	<ul style="list-style-type: none"><li>- Product Feature Epic</li><li>- PI Epic</li><li>- PI Objective Epic</li></ul>
Milestone	<ul style="list-style-type: none"><li>- Sprint Number - Milestone</li></ul>
Labels	<ul style="list-style-type: none"><li>- Workstream / Squad identifier</li><li>- Story type identifier ( User Story, Technical Story, QA Story, Spike, Enabler, Debt )</li></ul>

**Mandated tags for Epics in PI Backlog**

Epics	<ul style="list-style-type: none"><li>- Product Feature Epic</li><li>- PI Objective Epic</li><li>- PI Epic (needed only for PI Objective Epic)</li></ul>
Milestone	<ul style="list-style-type: none"><li>- PI number ( tbd, Zenhub constraints )</li></ul>
Labels	<ul style="list-style-type: none"><li>- Workstream / Squad identifier</li></ul>

# BENEFITS

1. Improved Transparency
  1. Snapshots of progress against dimensions
    1. Program Increment
    2. Sprint
    3. Workstream
2. Consistency across workstreams involved in a PI
3. Consistency across squads within a workstream

**Requesting everyone's support to implement these changes in PI-11  
Experiment in PI-11; Learn & Adapt**

Your feedback is welcome. Please reach out to Sankar Ramakrishnan, Sam Kummary