



Mifos Lab Updates

Edward Cable, The Mifos Initiative
Istvan Molnar, DPC Consulting

Agenda

- Community Need
- Benefits of Payment Hub EE
- Accelerating Mojaloop Adoption
- Payment Hub EE Traction
- Progress During the Past PI
- Demo
- Roadmap Items

Mifos Initiative & DPC

- 501(c)3 non-profit guiding Mifos OS community advancing Apache Fineract
- Stewards of roadmap & collaborative center
- Industry thought leader & HFOSS pioneer since 2006
- Guide Ecosystem of Solutions & Network of Partners
- 15 million clients reached across 350 orgs



Edward Cable

mojaloop
foundation



Mifos Initiative

**FINANCIAL
INNOVATION
AWARDS
2019**

Highly Commended

Best financial inclusion or
outreach initiative

Presented by

140 The London Institute
of Banking & Finance



István Molnár



Kristóf Józsa



Ádám Sághy

Zoltán Nébli

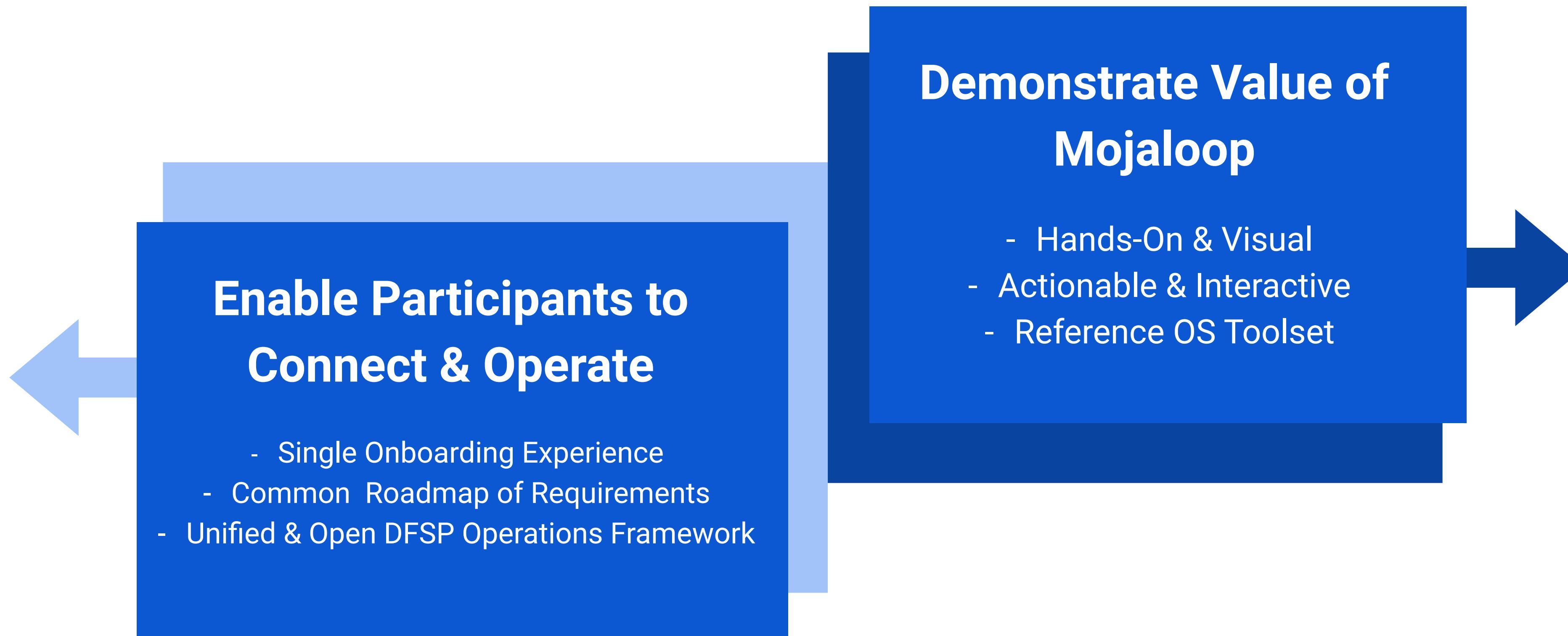
Zoltán Mezei

- 20+ years in IT training, consultancy, software development
- Experience with Instant Payment Systems (Singapore FAST, Hungary HCT Inst, SEPA Instant)
 - Including clearing house solutions, payment hubs, shadow balance solutions from key vendors
 - Developing central clearing house prototype, simulator for participants, payment hub

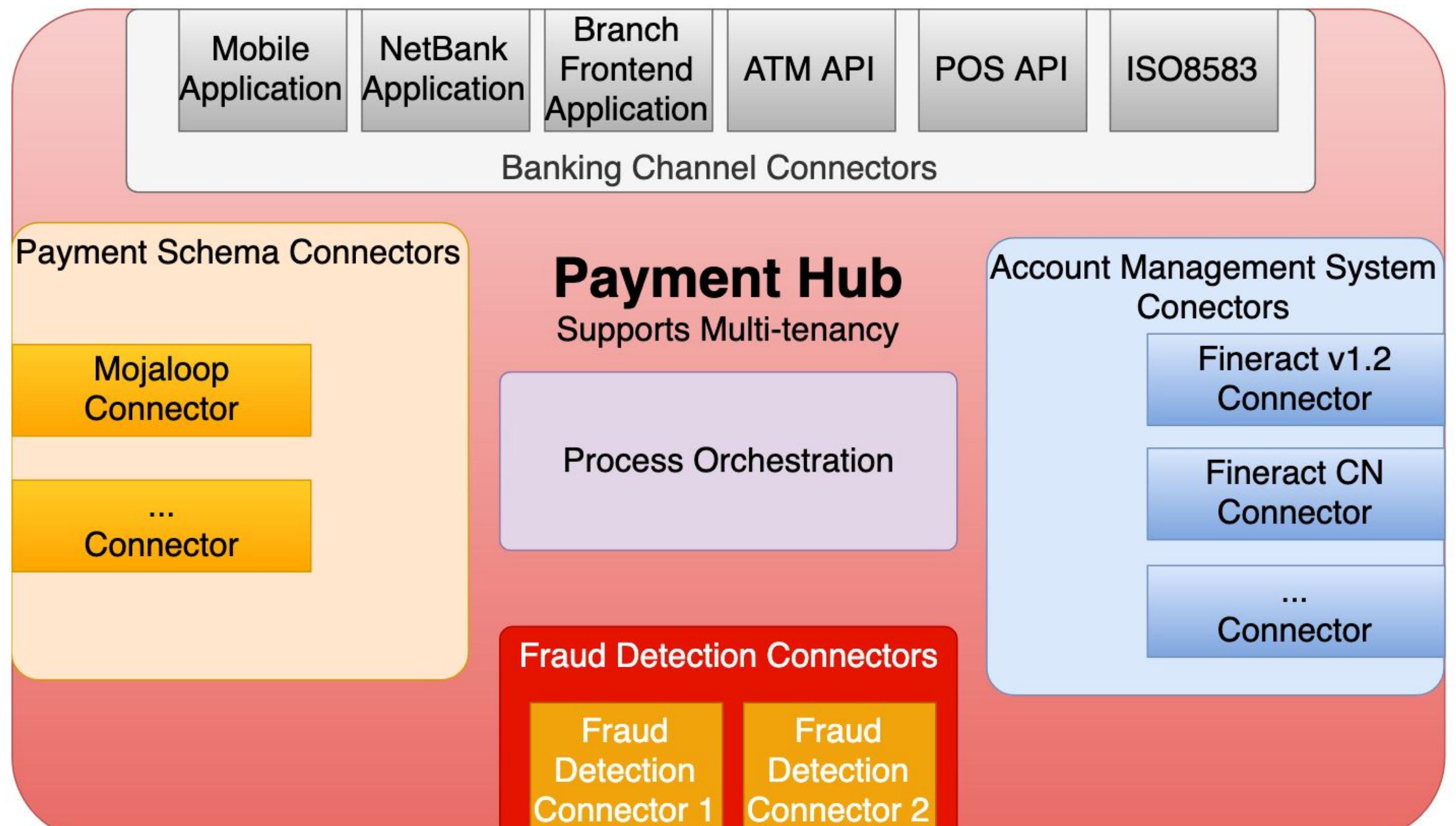


Community Need

Accelerate Adoption of Mojaloop



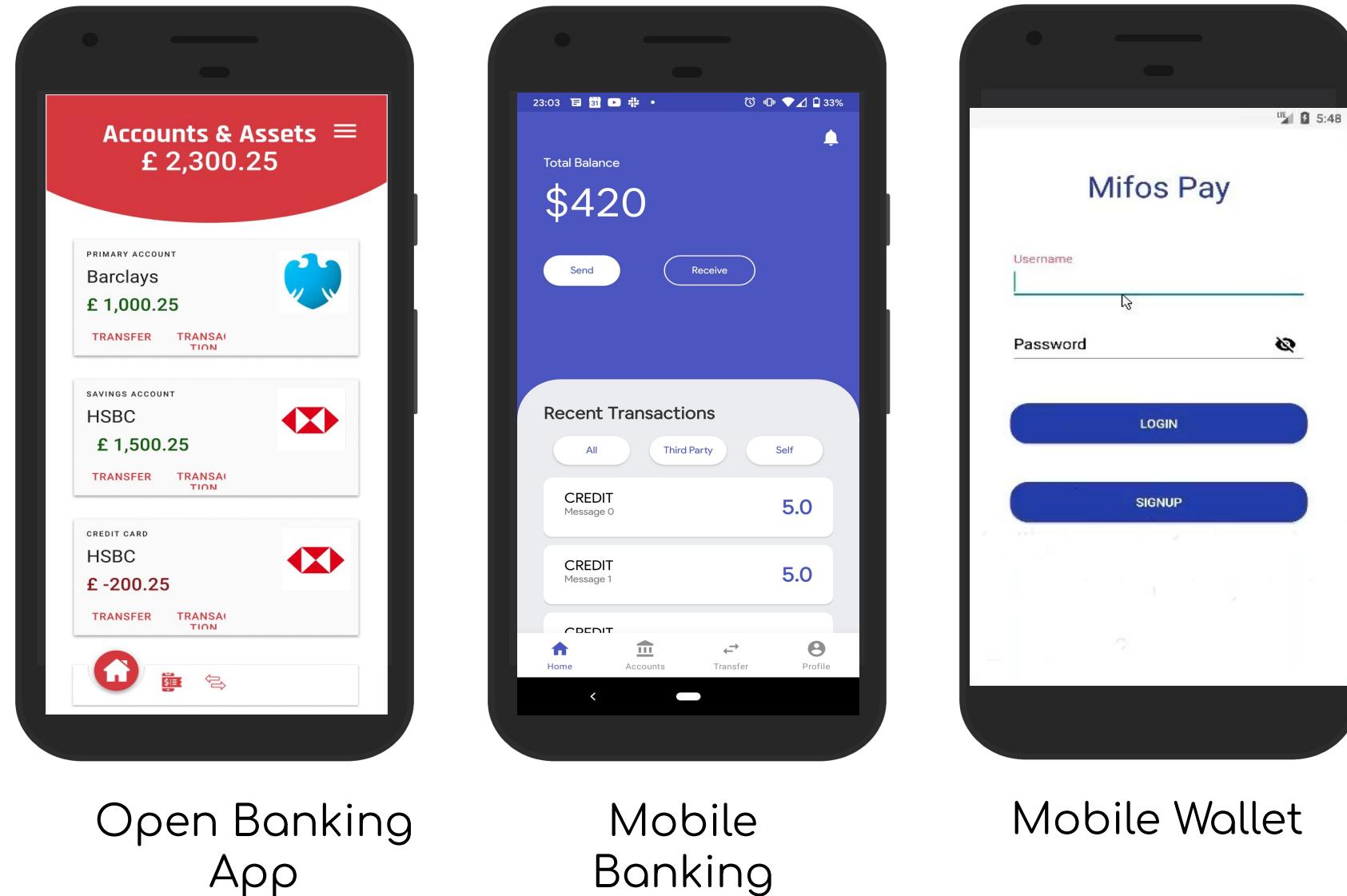
Payment Hub EE



Enable Participants to Connect & Operate

- Single Onboarding Experience
- Common Roadmap of Requirements
- Unified & Open DFSP Operations Framework

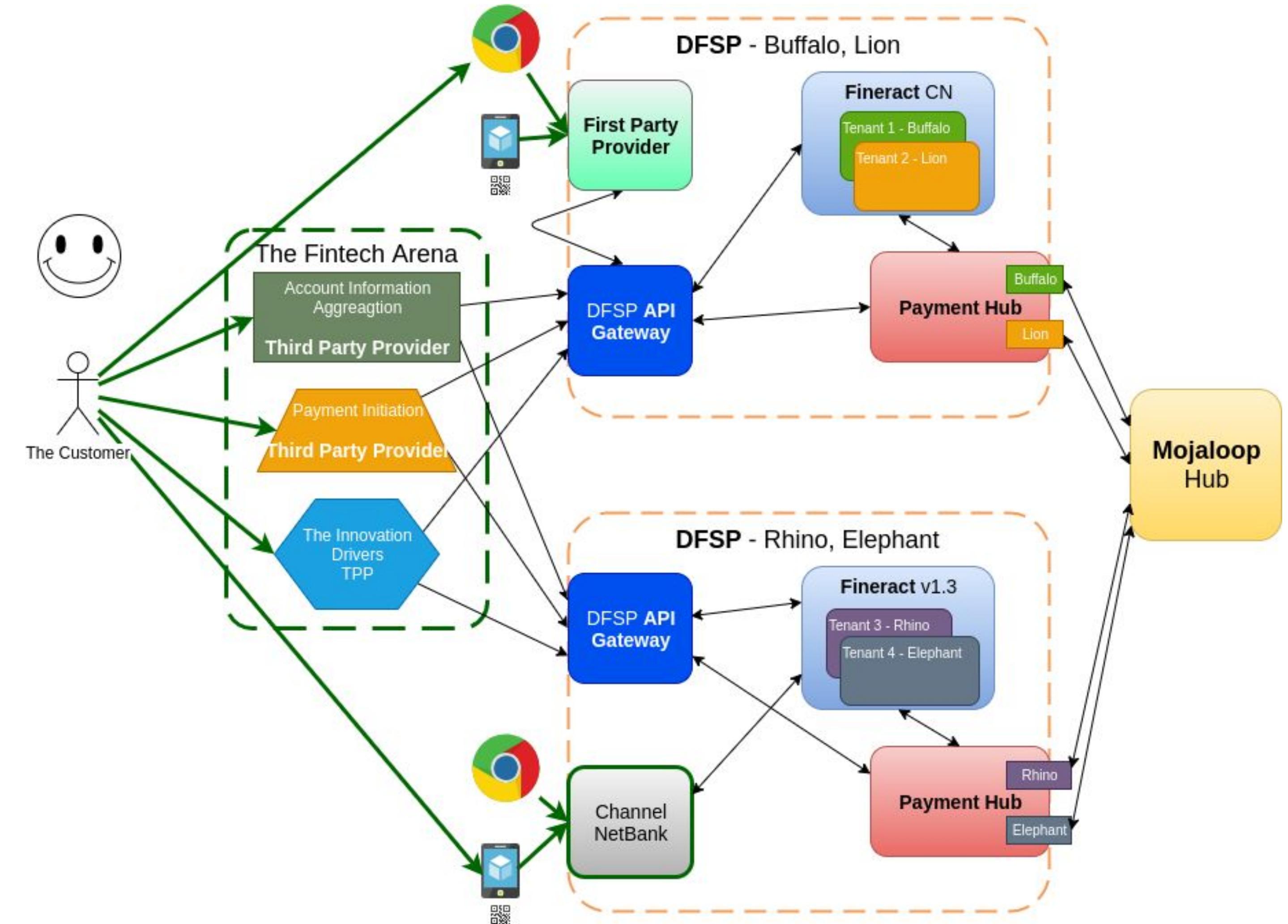
Interactive & Immersive Lab Environment



Open Banking App

Mobile Banking

Mobile Wallet



Demonstrate Value of Mojaloop

- Hands-On & Visual
- Actionable & Interactive
- Reference OS Toolset

Long Tail of Adoption and Innovation

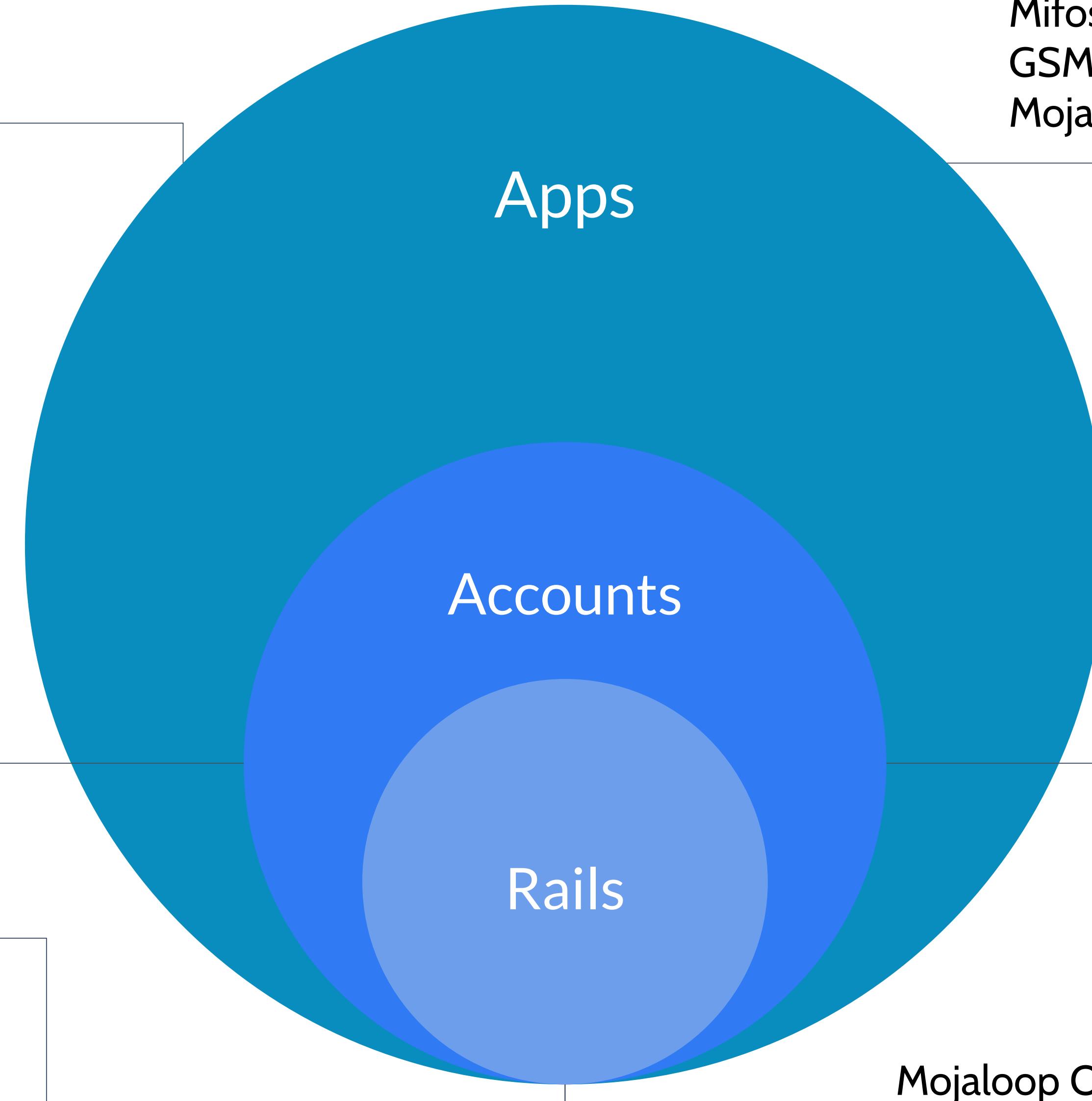


Fintechs/PISPs

DFSPs

Switch Operator

mojaloop
foundation



Mifos/Fineract
OS Core Banking API

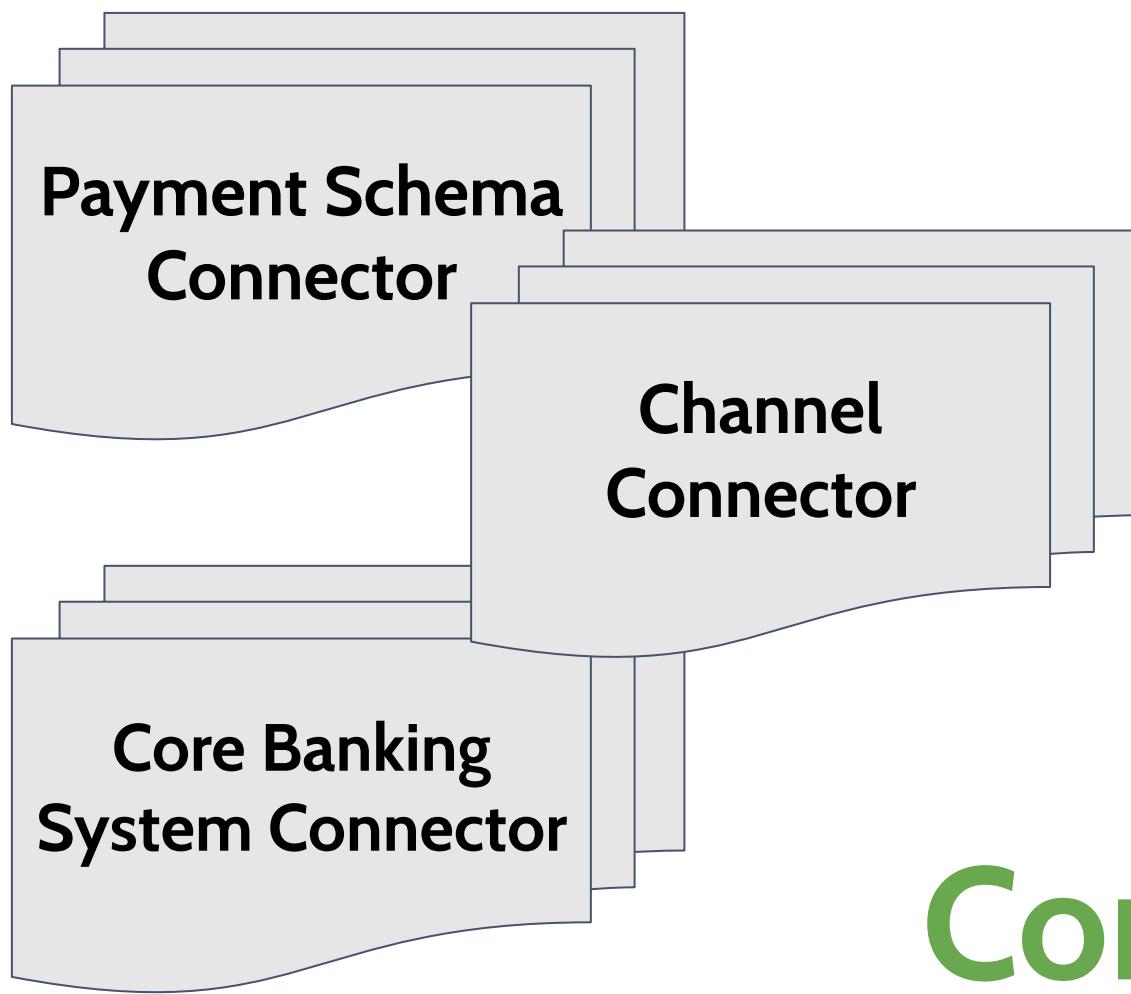


m

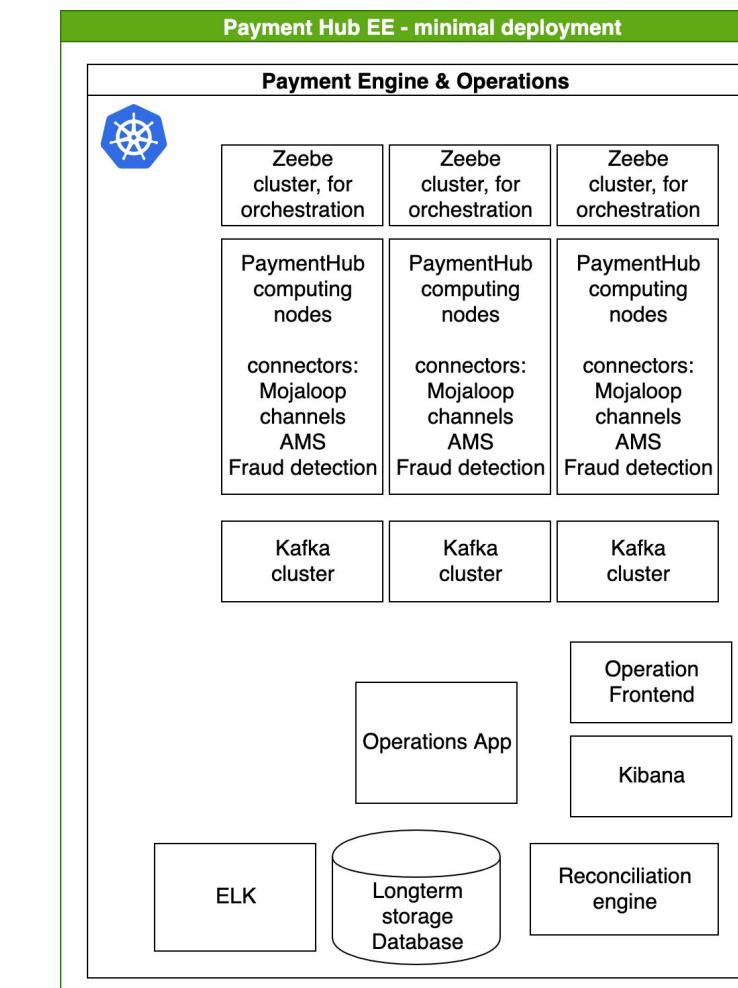
Mojaloop Open API

Benefits of Payment Hub EE

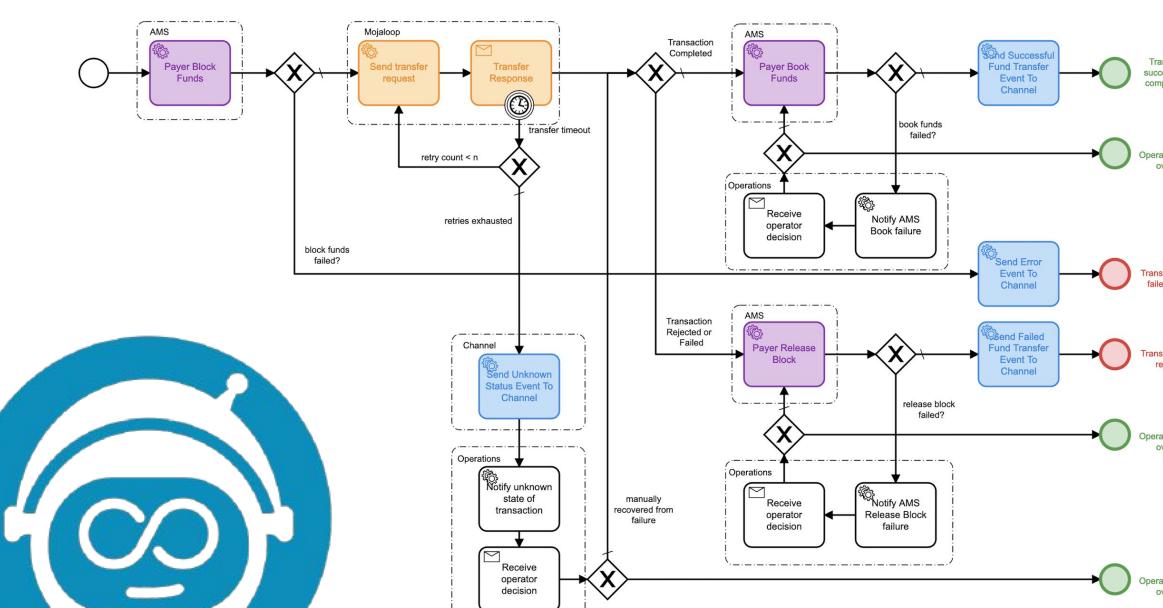
Extensible



Scalable



Configurable

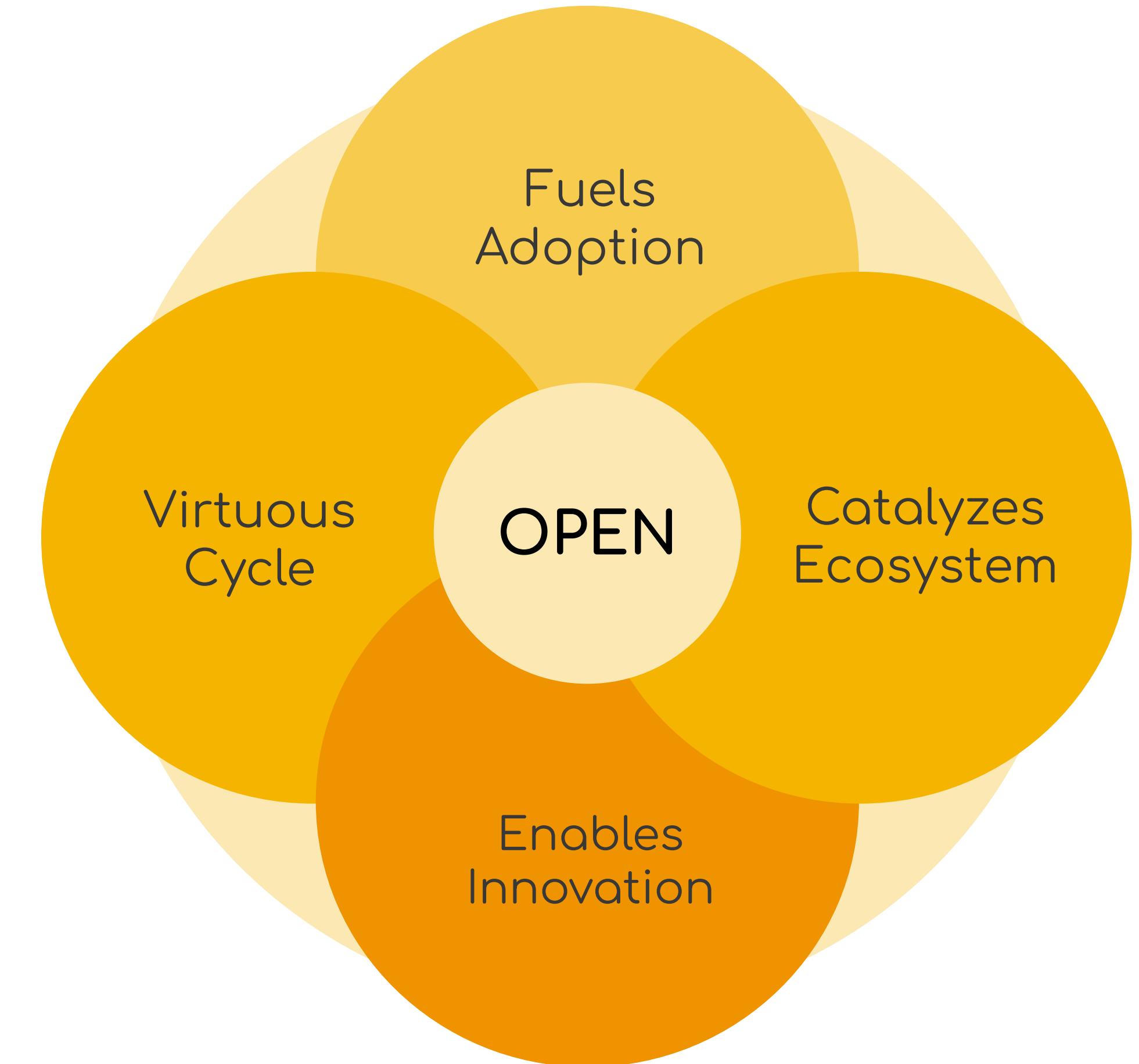


Practical

A screenshot of a web application interface for 'Payment Hub EE'. The top navigation bar includes 'Home', 'Payment Hub EE', 'Admin', and a search bar. The main content area shows a transaction details page for ID 2251799814249533. It displays information for the Payer (MSISDN 27710306999, DFSP id in03tn06, Gorilla Bank) and Payee (MSISDN 27710101999, DFSP id ir01tn01, Buffalo Bank). The transaction details include Transfer Code 58f4aea5-8cc4-404f-98ee-191ef1fc02b9, Transfer Amount 215 Tzs, Transfer Currency Tzs, Transfer Completed 2020-07-22 10:54:15, and Transfer Status COMPLETED. A 'Refund' button and a 'BPBM Diagram' link are visible in the top right.

Why DFSP Business Ops should be Open

- ❑ Lower Total Cost of Ownership
- ❑ Accelerated Development
- ❑ Focus On Differentiation & Value-Add
- ❑ Freedom from Vendor Lock-In
- ❑ Sustainable Business Models



Tier 3 and 4 Institutions

Turnkey solution for digitization & digital transformation of MFIs & SACCOs

Challenge

MFIs lack processes & systems to participate in switch

MFIs lack systems with 24x7 capabilities

MFIs have wide variety of core banking systems and channels

MFIs learn in visual manner

Difficulty connecting dozens or hundreds of MFIs

MFIs need support and training in getting regulated & connected

Solution

Cloud solution for digitizing and automating core banking operations

Payment Hub EE can provide stand-in processing

Extensible by simply building additional core banking system or channel connectors.

Hands-on lab environment with actionable tools

Multi-tenancy of Mifos and Payment Hub EE enabling economies of scale for shared service providers

Mifos network of local on-the-ground integrators & support partners with deployment & domain expertise.



Tier 1 DFSPs

Robust Performance
Deployment Flexibility
Ease of Connecting
Operational Controls
Seamless Channel Integration



- Core Banking System Agnostic
- Highly configurable workflow engine
- Extensible via connectors
- Ready to customize and extend
- Vendor-agnostic
- Operational Control Center

Fintechs

Ease of Connection
Rapid Innovation
Sandbox
API Standardization
Use Case Prototyping



- OS Reference Apps
- Interactive Lab Environment
- End to End OS Stack for DFS
- Open Banking API Layer
- GSMA Mobile Money API support

Regulators

Transparency & Visibility
More input into Fraud Detection
Velocity of Cash



- DFSP-level fraud monitoring
- Visibility into on-us transactions
- Generate regulatory reports
- Consistency of data
- User Access for regulators

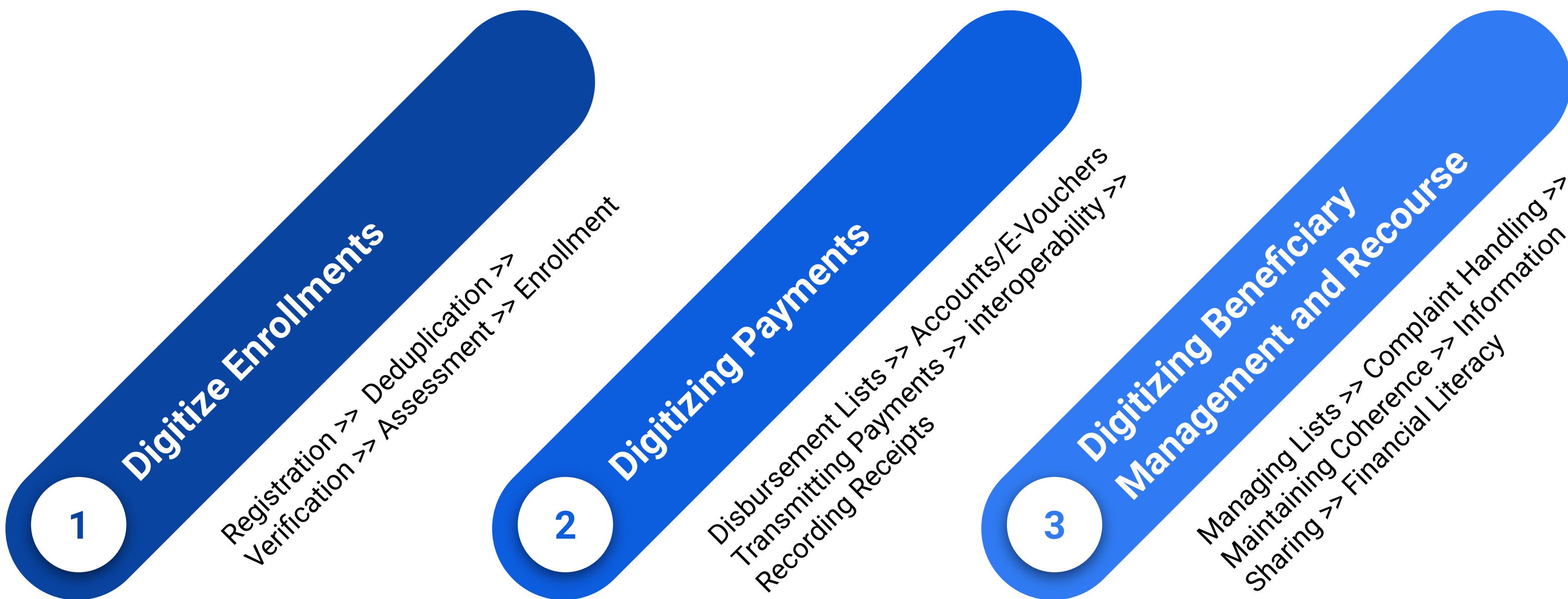


G2P & Bulk Payments

Open G2P

Digital public good digitizing large scale cash transfers with open source building blocks

- ❑ **COVID-19 Opportunity:** Accelerating cash transfer is the single most important response to getting assistance in the hands of people who need it most in a timely and transparent manner.
- ❑ **Origin:** Originated from iDT Labs and Government of Sierra Leone when they saved more than \$10M digitizing payments to 30,000 health workers during Ebola crisis
- ❑ Addresses **common challenges** in effectively digitizing government to persons & large-scale social protection transfers.



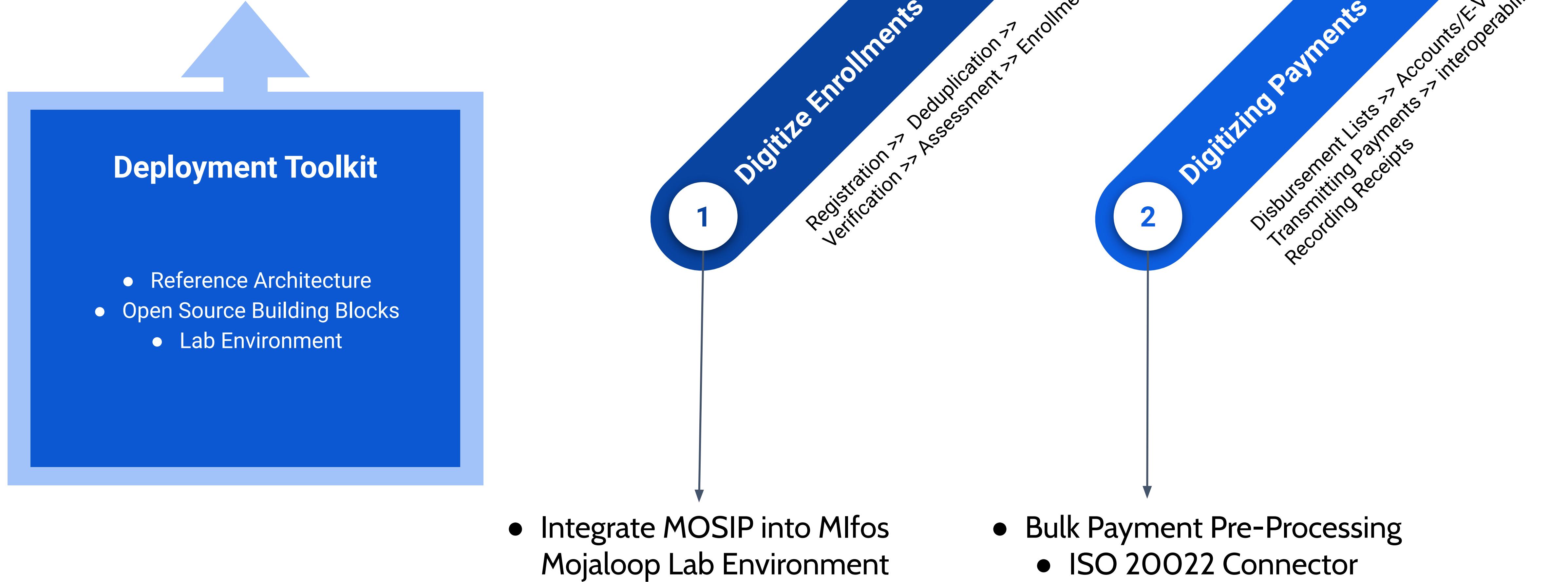
Open G2P

What is it?



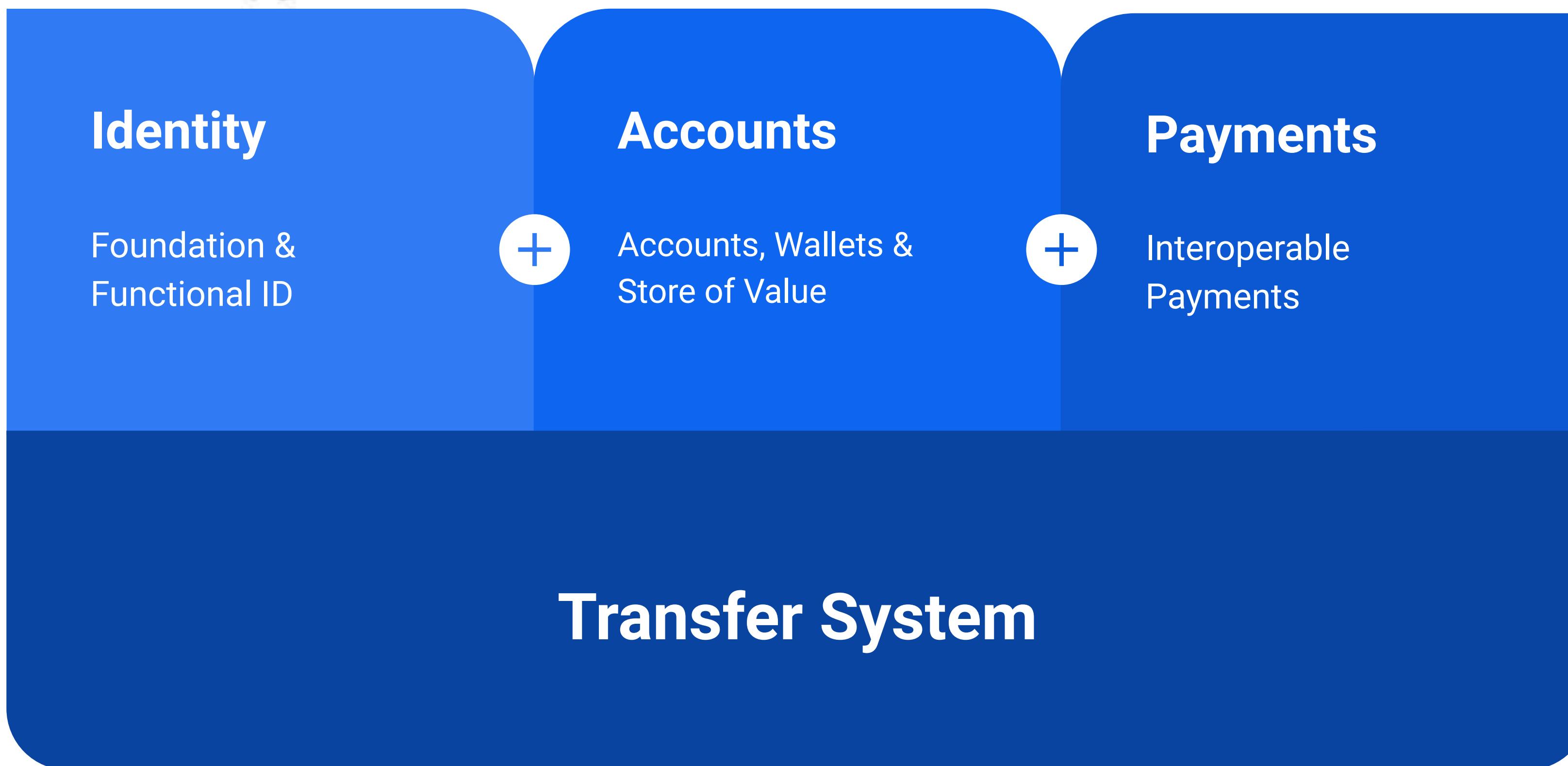
Open G2P

Roadmap alignment





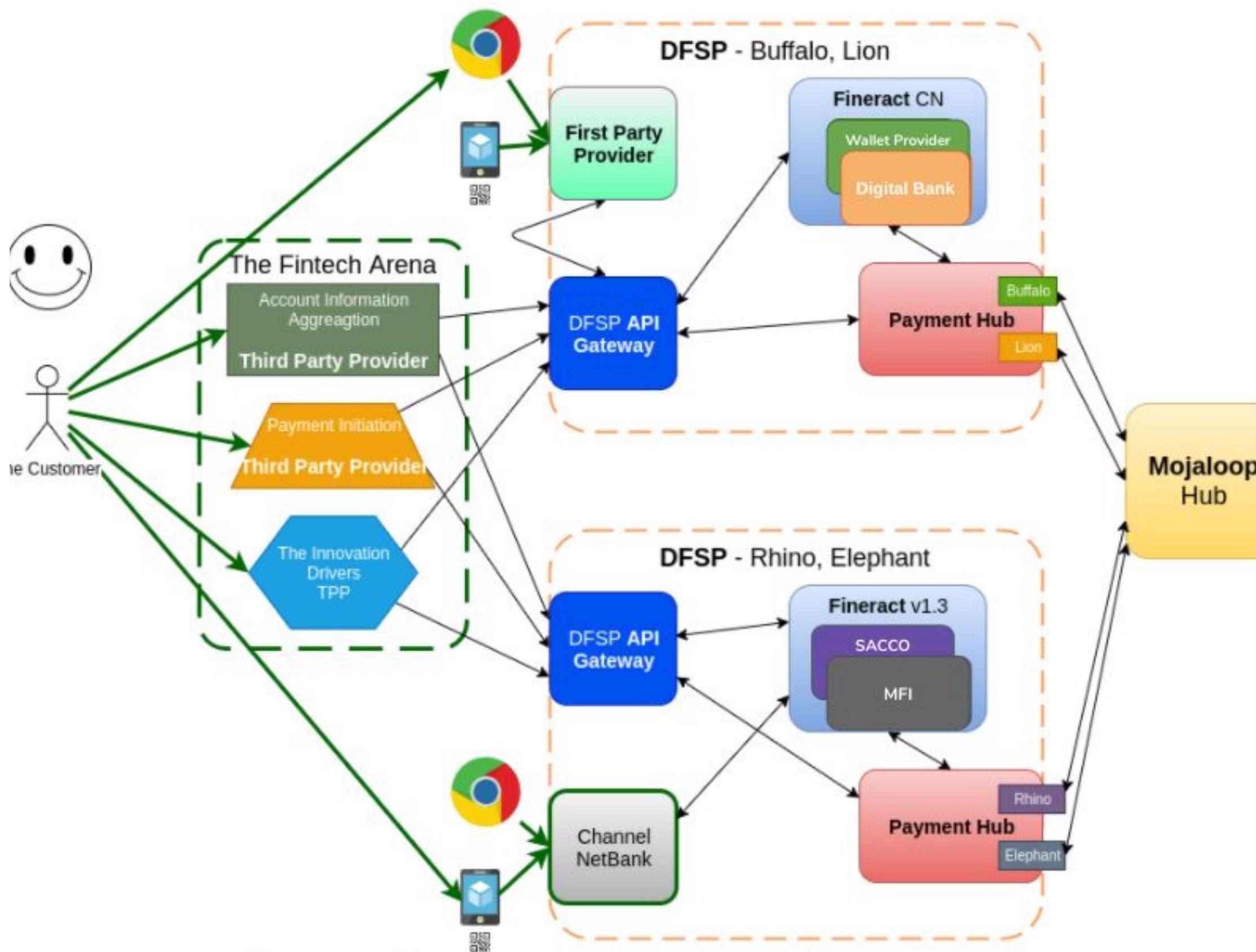
mojaloop



Fintechs Exploring Mojaloop in our Lab Environment



Sellpay Liberia



mojaloop
foundation



Circle Cash Sudan

Audience	API Layer	OS Toolset
Fintechs	<ul style="list-style-type: none"> - Open Banking API Layer - GSMA Mobile Money API (in progress) 	<ul style="list-style-type: none"> ➤ Mobile Wallet App ➤ Mobile Banking App ➤ Online Banking App ➤ Open Banking Fintech App
	Mifos/Fineract API Layer <ul style="list-style-type: none"> - Identity & KYC - Wallet/Account Management - Loan & Savings Management - Accounting & Ledger - Reporting 	<ul style="list-style-type: none"> ➤ API-Driven Open Source Core Banking Platform ➤ Angular Web UI for staff ➤ Android Mobile UI for staff
	Mojaloop API Layer <ul style="list-style-type: none"> - Peer to Peer - Merchant Proximity Payment - Merchant Request to Pay - Bulk Payment/Transfer (upcoming) 	<ul style="list-style-type: none"> ➤ Payment Hub EE <ul style="list-style-type: none"> ○ Operations UI ○ BPMN Workflow Engine

Payment Hub EE Traction

Where's it Being Deployed in Mifos Ecosystem

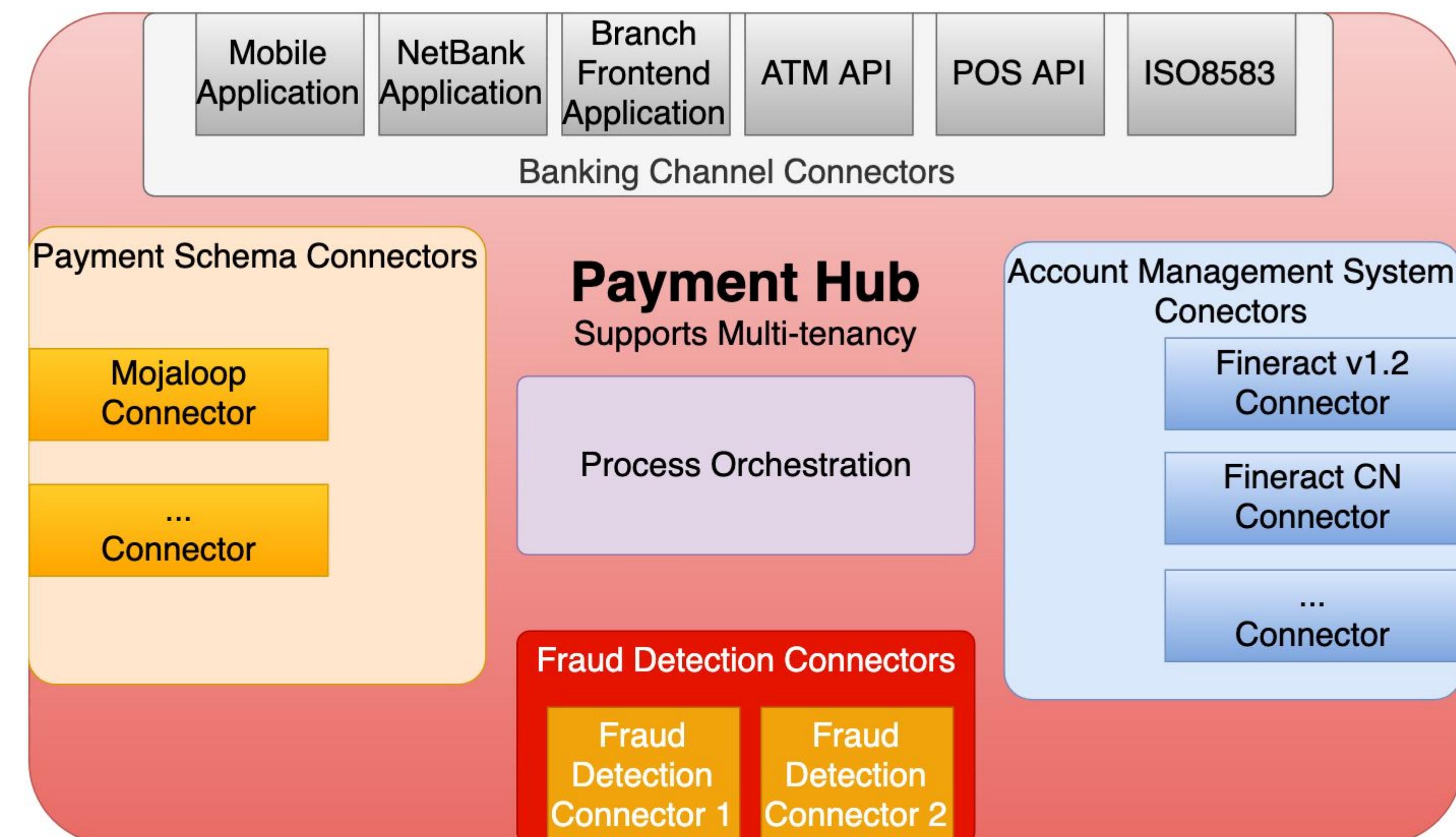
- ❑ Cross-border Payments Network between USA & India
- ❑ Bulk Disbursements for Banco del Bienestar in Mexico
- ❑ Stokvel Digitization in South Africa
- ❑ SACCO Digitization in Kenya
- ❑ Mobile Wallet in Cote D'Ivoire
- ❑ Rural Bank Digitization in Ghana
- ❑ Neobanks in SE Asia and Latin America

Additional Connectors

- ❑ Payment Schema
 - ❑ In-Progress
 - ❑ M-Pesa (Kenya)
 - ❑ SPEI (Mexico)
 - ❑ Under Consideration
 - ❑ Interac (Canada)
 - ❑ MFS Africa
- ❑ Channels
 - ❑ GSMA Mobile Money
 - ❑ Nextgen SIM Overlay

Payment Hub as an Open Source Asset for the Community

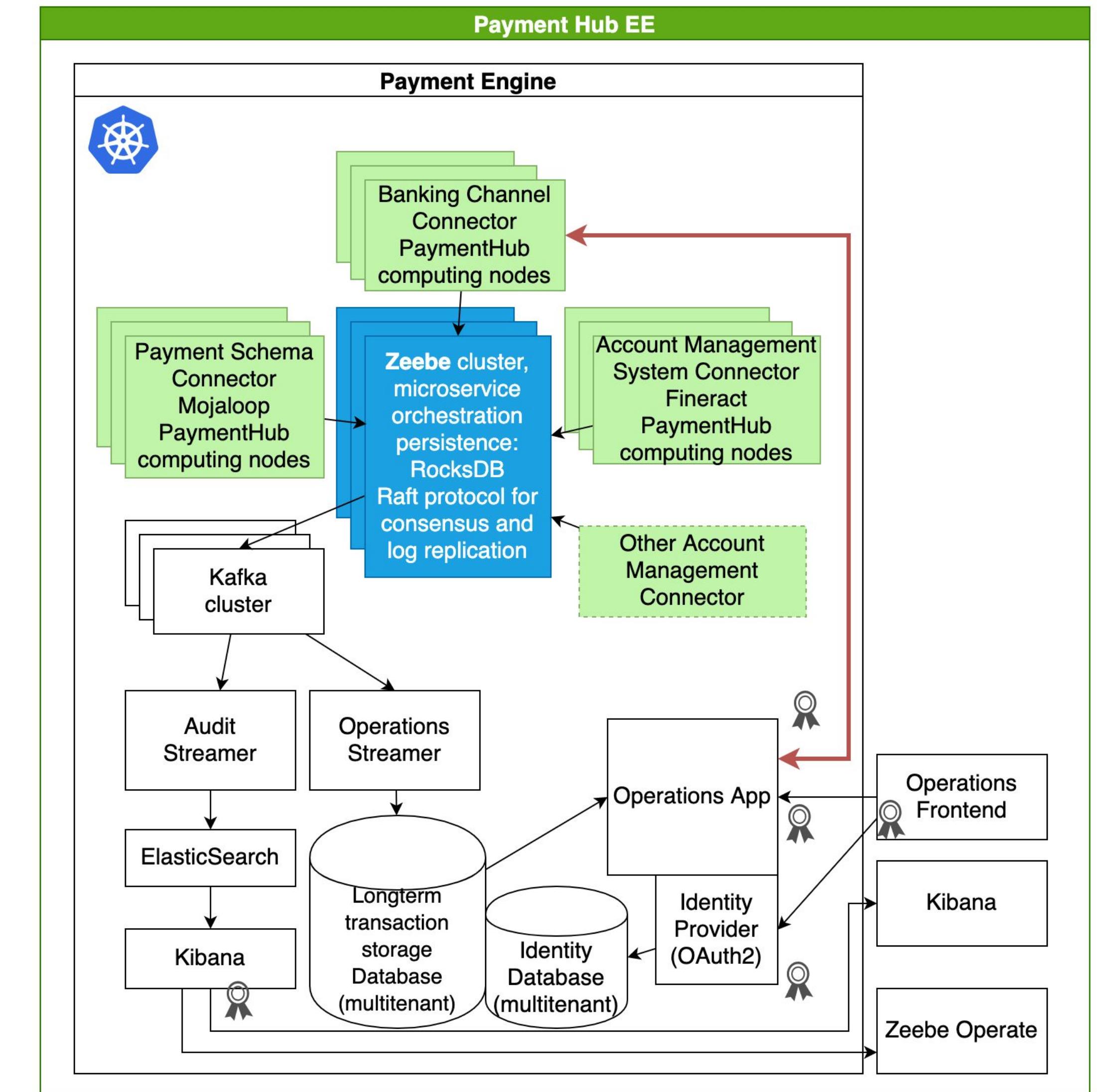
- The role of a payment hub to connect:
 - Financial Institution channels (Mobile, Internet, Branch, Callcenter, ATM, POS, API Gateways)
 - Account Management Systems (AMS / Core banking platform), optionally fraud monitoring tools
 - Payment Schemes, such as Mojaloop
- Need
 - Consistent Way to Connect to Mojaloop
 - Effective Operational Participation
- Additional Capabilities
 - DFSP-level fraud monitoring
 - Bulk Transfer Campaign Management
 - Operational Monitoring
 - Manages the identifier – account relation
 - Trigger notifications
- Built on proven open-source technology:
 - Java, SpringBoot, Kafka, Elasticsearch
 - Apache Camel, Camunda Zeebe
 - Kubernetes



Payment Hub Components

Microservices based architecture to support easy customization for integrators

- microservices orchestration
- microservices for the connectors
- running in a Kubernetes cluster
- operations application for DFSP command center



Focus on system integrators

- BPMN let implementation team **focus on the business flow changes inside the given DFSP**, making adoption and customization much easier
- **Multi programming language support for component implementation**, let implementation team to create the necessary connector components in any of the supported languages, the engine is able to orchestrate the steps. (e.g. multiple AMS integration is required), using gRPC protocol for efficient communication
- **Connector components are stateless, easy and simple implementation**
- No additional components required in the realtime engines, no databases to maintain, all active process states are stored in the embedded RocksDB data store. Auditlogs, and other events are sent to a Kafka cluster before storing them for long term access
- Scalability - using partitioning (sharding), the system scales horizontally to create thousands of workflow instances per second
- Fault tolerance enables, that system recovers from a failure without data loss

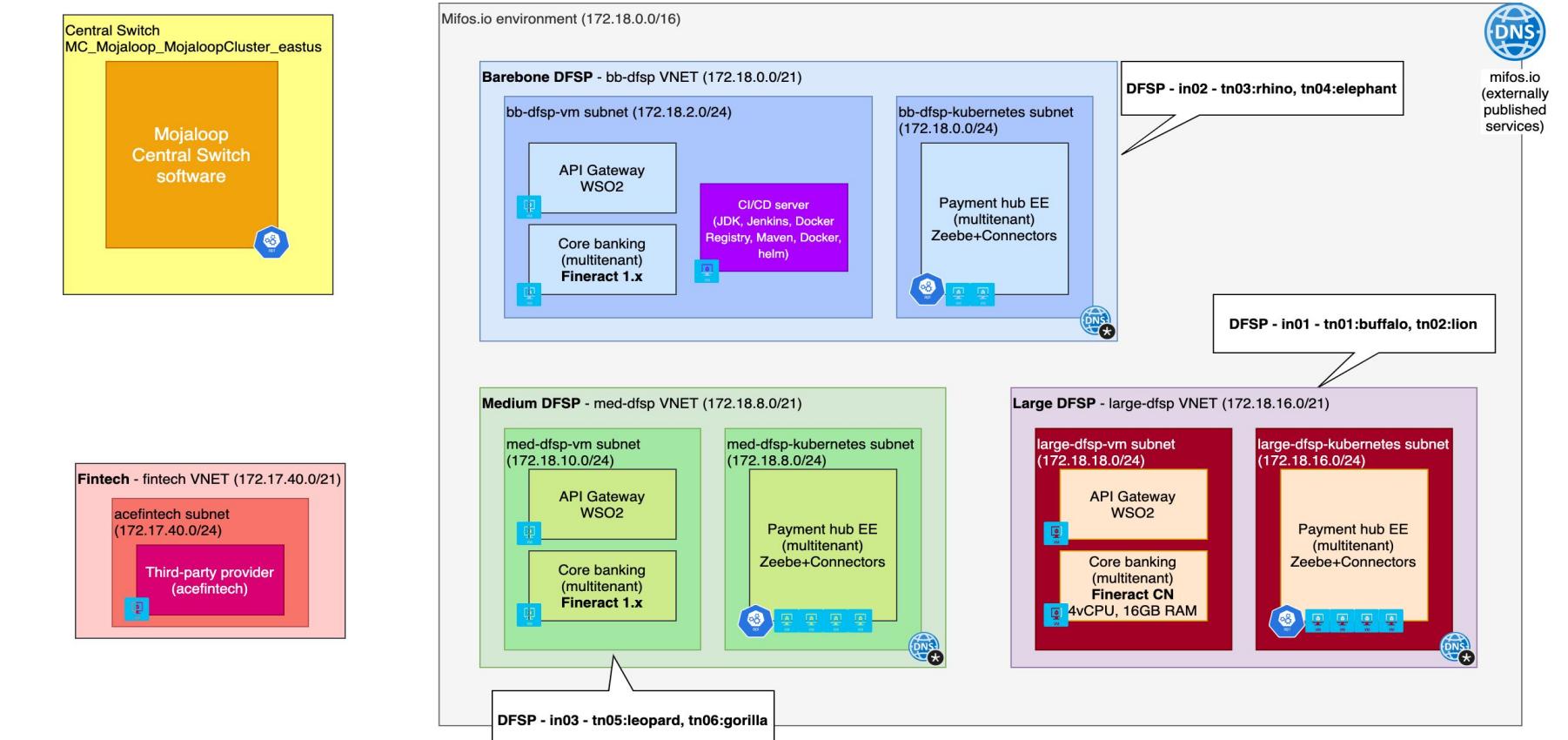
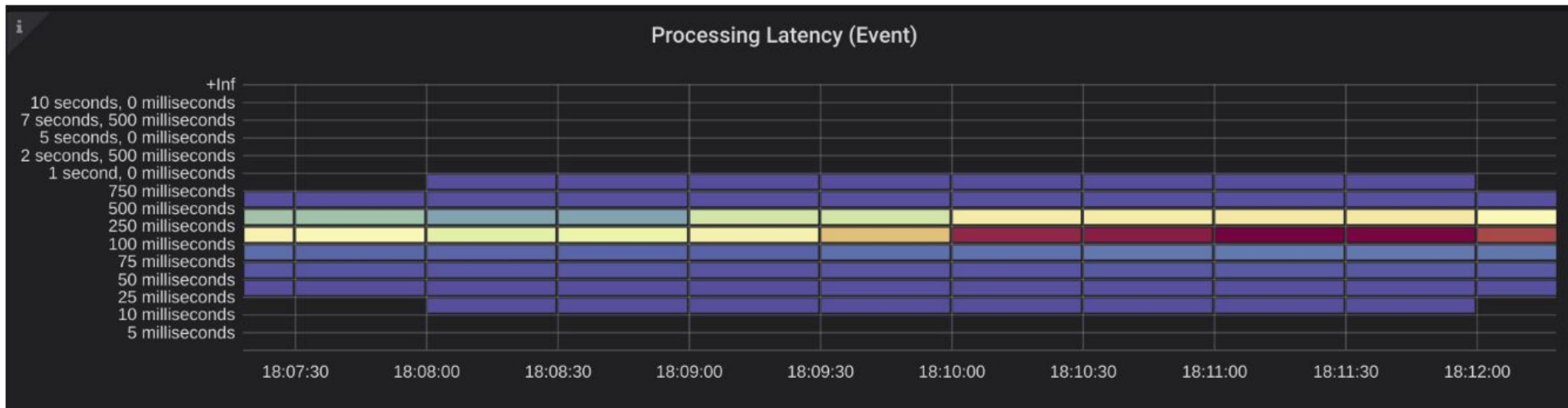
Performance measurements

Payment Hub EE Large Lab Environment

- 4 nodes Kubernetes cluster on Azure (3 × 8 cpu core, 56Gb ram, 1 × 2 cpu core, 2Gb ram)

Test benchmark

- 10000 complete payment transaction flows
- number of parallelly running workflow instances peaked over 5200 workflows
- all transactions finished within 3:15, hitting the **average of 51 cfTPS**
- single event process latency primarily within 100-250ms
- executing payment transfer operations with over 300 fTPS



Tuning

Performance tuning documentation in the PHEE Handbook:

<https://mifos.gitbook.io/docs/payment-hub-ee/overview/tuning-and-performance>

Primary tuning considerations

- Zeebe orchestration engine parameters (partitions count, replication factor, CPU / IO thread counts, JVM garbage collector options)
- replace built-in Elasticsearch exporter with PHEE's improved asynchronous Elasticsearch exporter
- tune backpressure options / disable for throughput benchmarks
- enable Prometheus metrics to get the necessary insights
- tune the number of connector pods & worker thread counts in the clusters to match the target load
- running on AMD cpus seems to have a serious impact on performance (under validation)

Load testing and measurements

Screenshot of the JBoss Seam Test Plan interface showing a configuration for an HTTP Request.

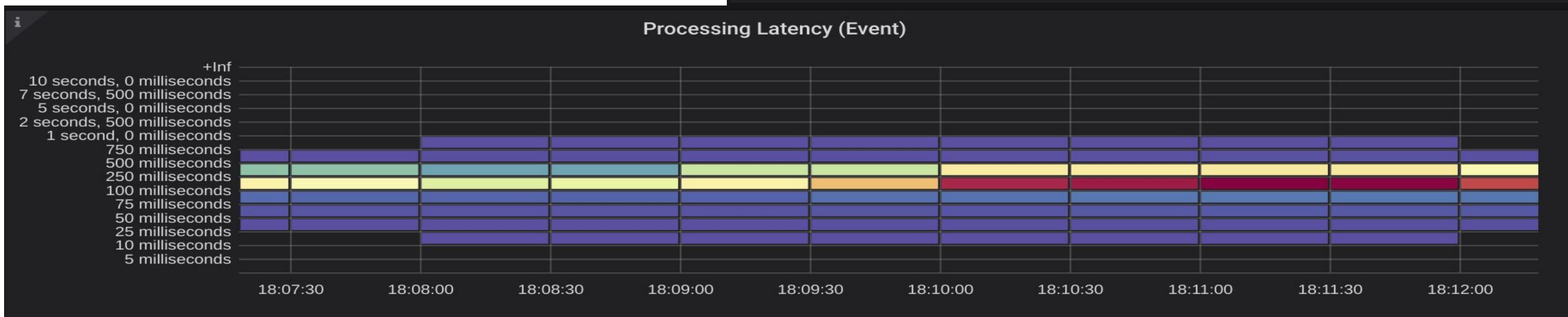
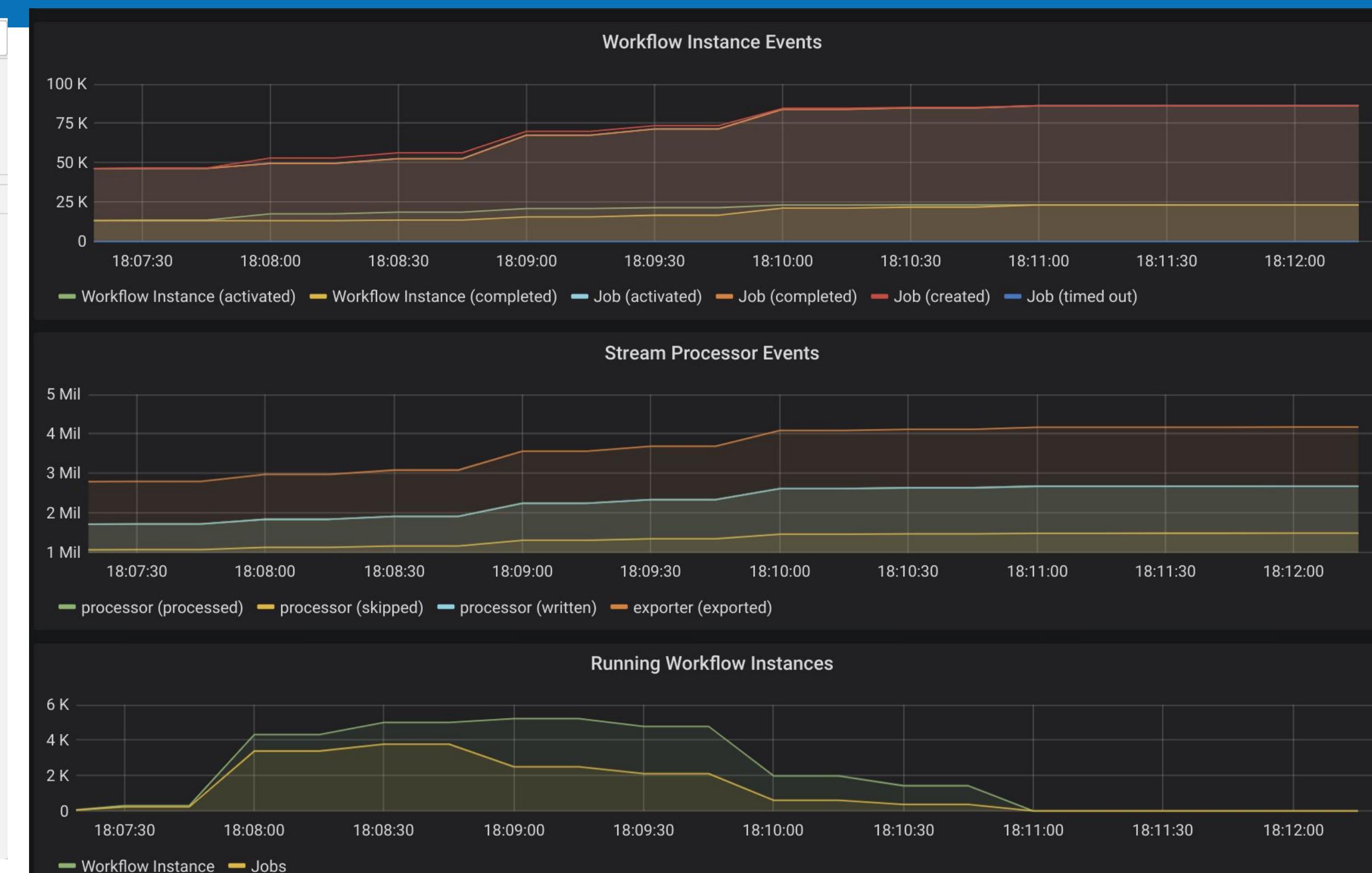
Test Plan (Tree View):

- HTTP Request Defaults
- HTTP Header Manager
- Counter
- bzm - Arrivals Thread Group
 - HTTP Request
- Thread Group
- View Results Tree
- Graph Results

HTTP Request Configuration:

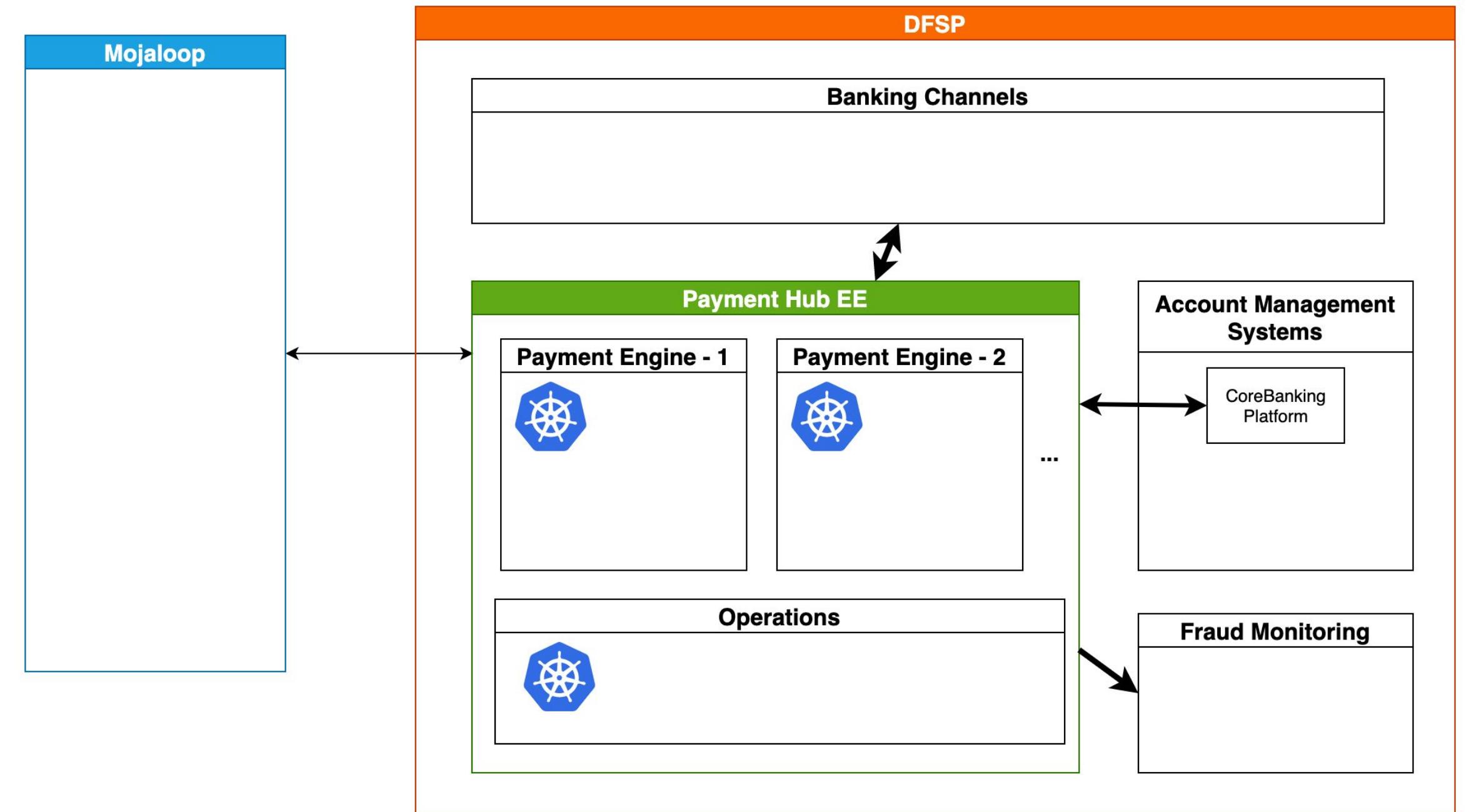
- Name:** HTTP Request
- Comments:** [Empty]
- Basic** tab selected
- Web Server:** Protocol [http], Server Name or IP: [Empty], Port Number: [Empty]
- HTTP Request:** Method: POST, Path: /channel/transfer, Content encoding: [Empty]
- Advanced Options:** Redirect Automatically (unchecked), Follow Redirects (checked), Use KeepAlive (checked), Use multipart/form-data (unchecked), Browser-compatible headers (unchecked)
- Parameters:** JSON payload (Body Data tab) shown below:

```
1: {
2:   "payer": {
3:     "partyIdInfo": {
4:       "partyIdType": "MSISDN",
5:       "partyIdentifier": "666627710101999",
6:       "fspId": "in01tn01"
7:     }
8:   },
9:   "payee": {
10:    "partyIdInfo": {
11:      "partyIdType": "MSISDN",
12:      "partyIdentifier": "27710305999"
13:    }
14: },
15:   "amount": {
16:     "amount": "${count}",
17:     "currency": "Tzs"
18:   },
19:   "transactionType": {
20:     "scenario": "WITHDRAWAL",
21:     "initiator": "PAYER",
22:     "initiatorType": "CONSUMER"
23:   }
24: }
```



DFSP Level Fraud Detection - Regulatory Fraud Detection

- The payment process could support the DFSP level Fraud monitoring tools
 - to stop transactions or alert operators
- There are requirements to keep the on-us transactions inside the DFSP
- PHEE can be the component to provide information to regulators with the required amount of granularity of on-us transaction details (considering GDPR and similar regulations)
 - send all transactional information as individual reporting messages
 - aggregate on-us traffic data for a required period (minutes to days)



Multitenancy

Single deployment serving multiple DFSPs

Isolated audit logs for the different DFSPs served with separate user databases for the DFSP operators

Benefits:

- . Single operator could provide services for multiple individual DFSPs while sharing operational costs
- . In case using a multi-tenant core banking platform (e.g. Mifos) a single Payment Hub EE deployment could serve all tenants (DFSPs)

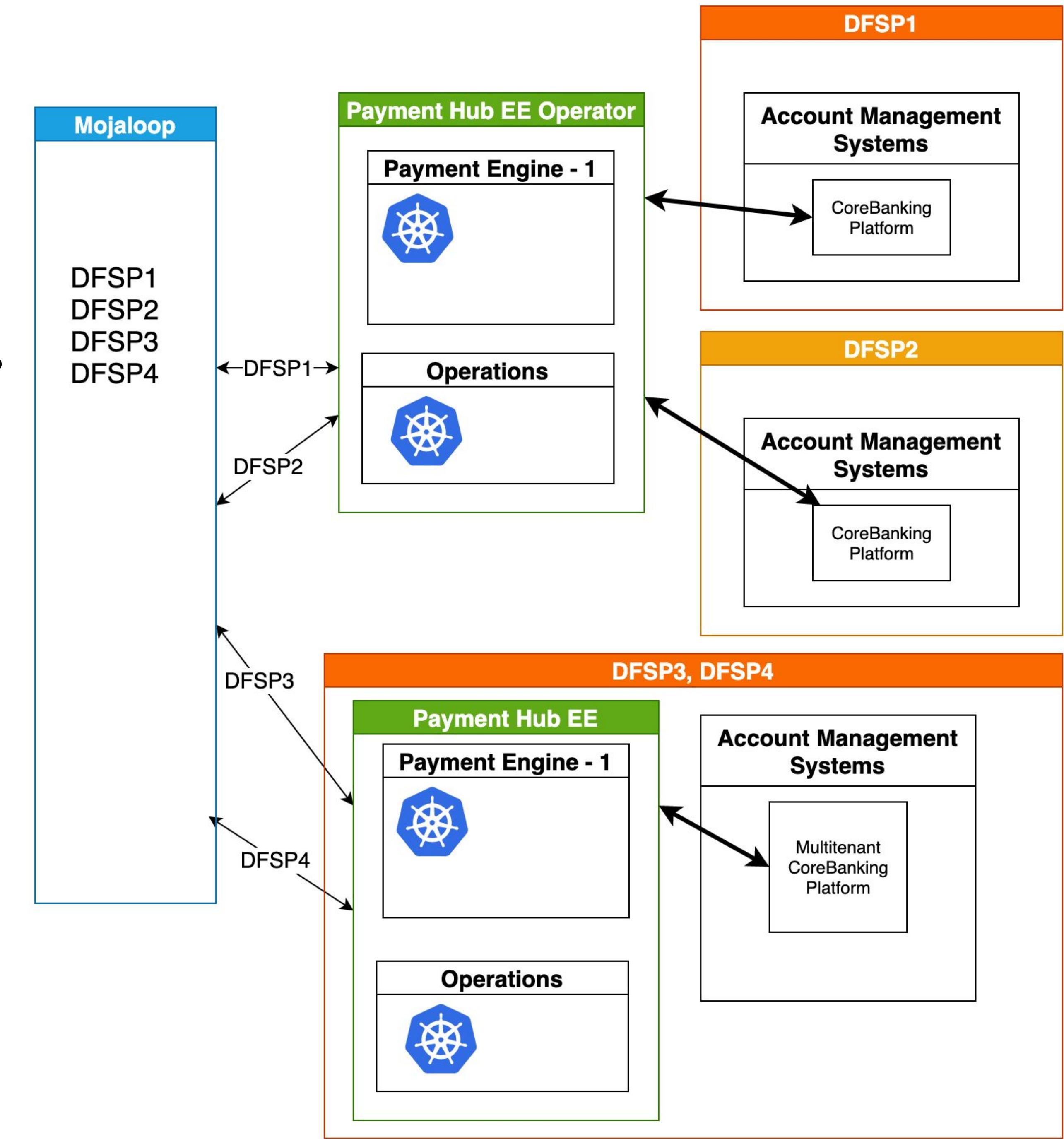
Multitenancy

Helping adoption for smaller MFIs

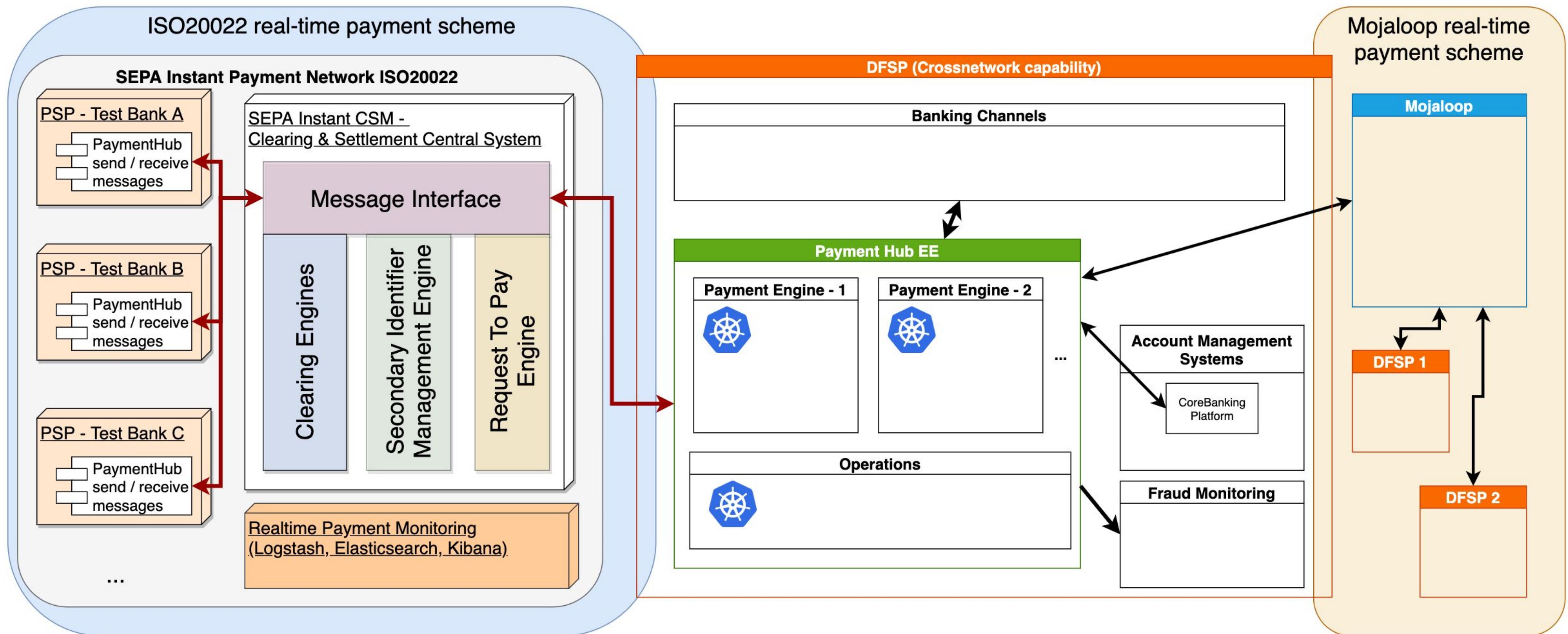
- sharing operational cost
- simplifying management
- reduce implementation effort

Aggregator model for multiple smaller DFSPs

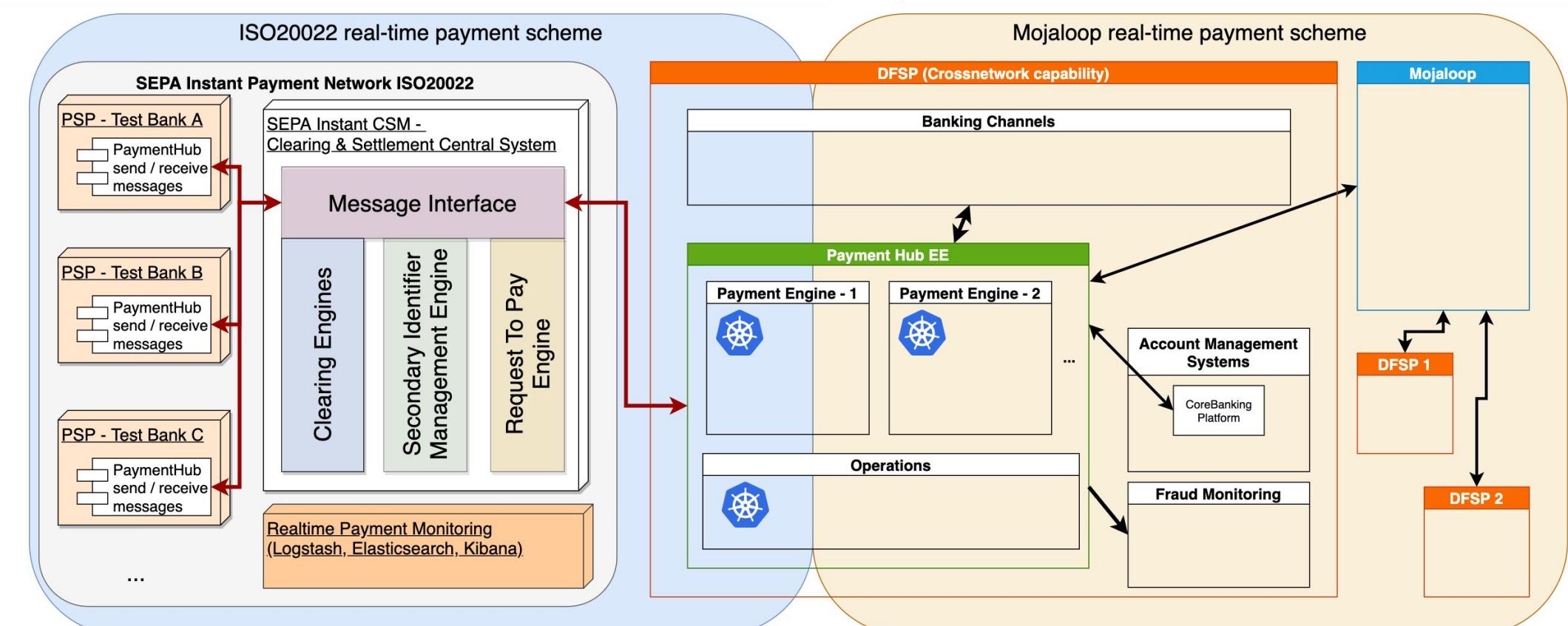
Multitenant model for already aggregated DFSPs served from a single multi-tenant core banking



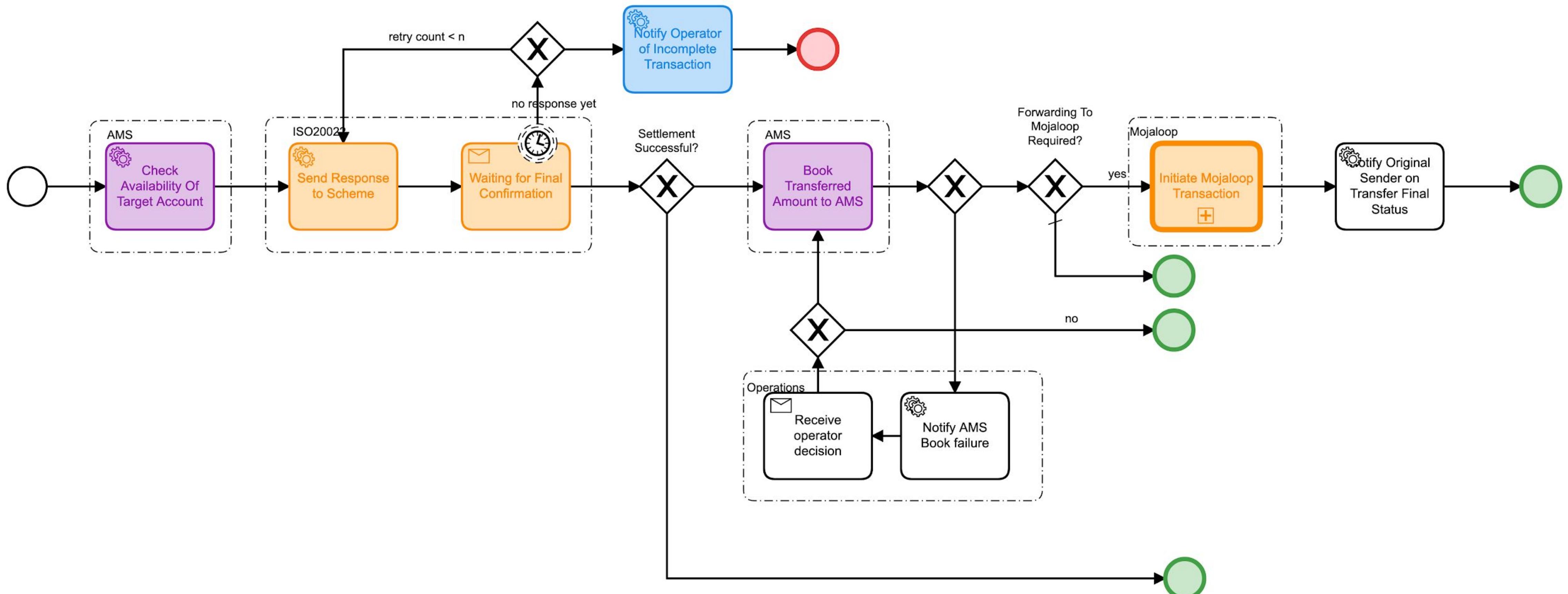
Crossnetwork - ISO20022 (Payments Clearing and Settlement - pacs.008, pacs.002) - Mojaloop



Crossnetwork - ISO20022 - Testing



Crossnetwork BPMN flow



Correspondent bank concept

- How to address the ultimate payer in a structured way in the Mojaloop transfer message structure?
- ISO20022 has terms Creditor and UltimateCreditor to handle this situation

Roadmap for Payment Hub EE

Continue working on bulk payments

Continue performance testing and high availability testing

Provide a platform for Cross Network Payments

- . ISO20022 (Payments Clearing and Settlement - pacs.008, pacs.002) realtime payment network integration
- . Other national networks and Mojaloop network

Providing stand-in capability

Reconciliation

Realtime transactions, Mojaloop reports, Corebanking ledger

Bulk Payment Support

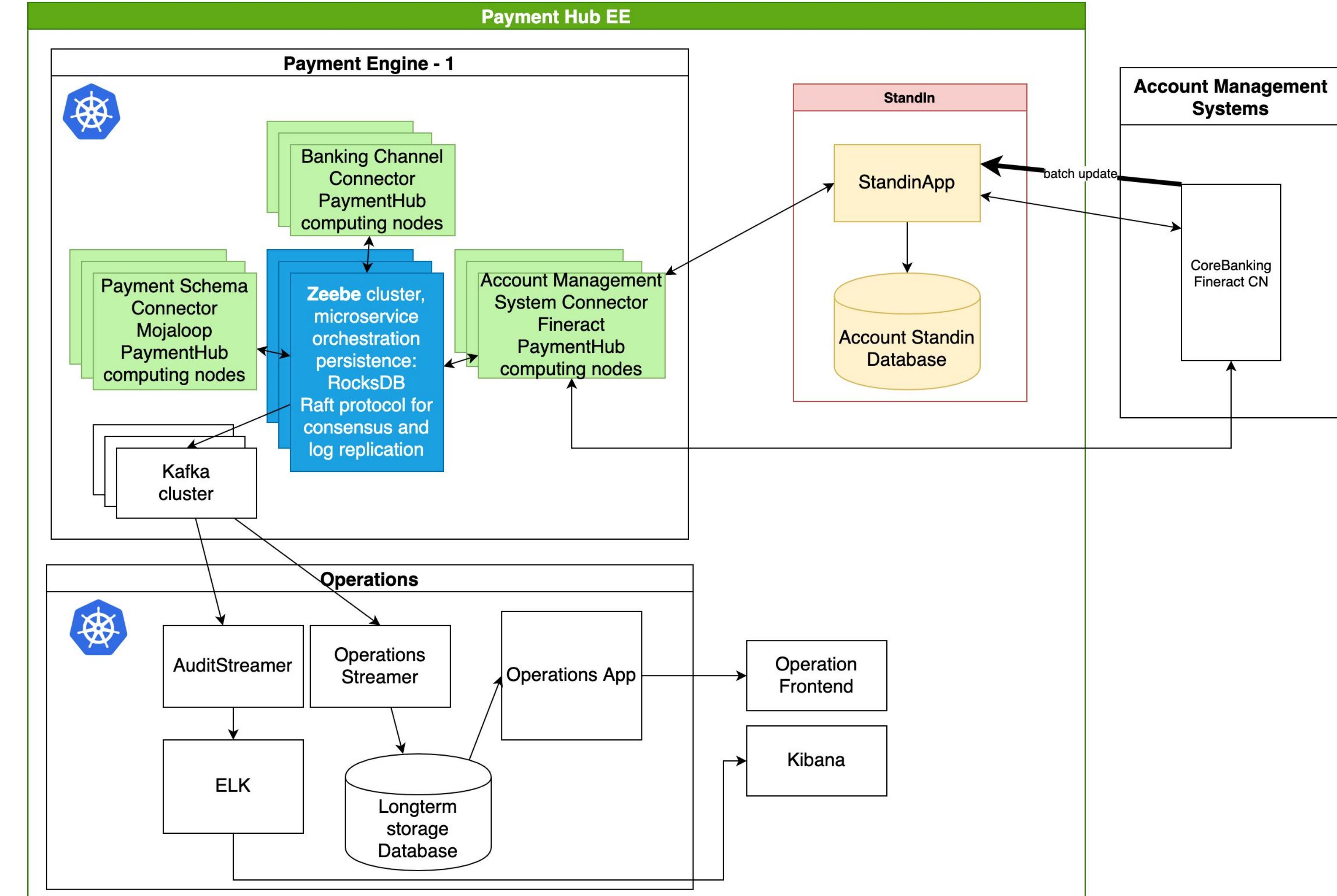
- . Preprocessing bulk payments
 - Lookup Payee's DFSP ID for the transactions to determine target DFSP
 - Splitting the incoming bulk into smaller batches per target DFSP
 - On-us transactions can be handled differently
 - Manage communication with Mojaloop for the batches
 - Aggregate incoming results to provide response to the bank's channel
 - In case transferring from a single account (pension, aid), booking can be individual, aggregated by target DFSP or single grand total

Stand-in

Integrated into the Payment Hub EE solution a stand-in system could provide functionality in case the Account Management System is not available.

MFI's systems are often not 24x7.

- only incoming transactions - simple solution, requires only synchronizing the valid account and party ids
- both incoming and outgoing transactions - requires more complex solution to minimize risk of overdraft



Documentation

Business Overview

- Vision
- Benefits
- Target Users

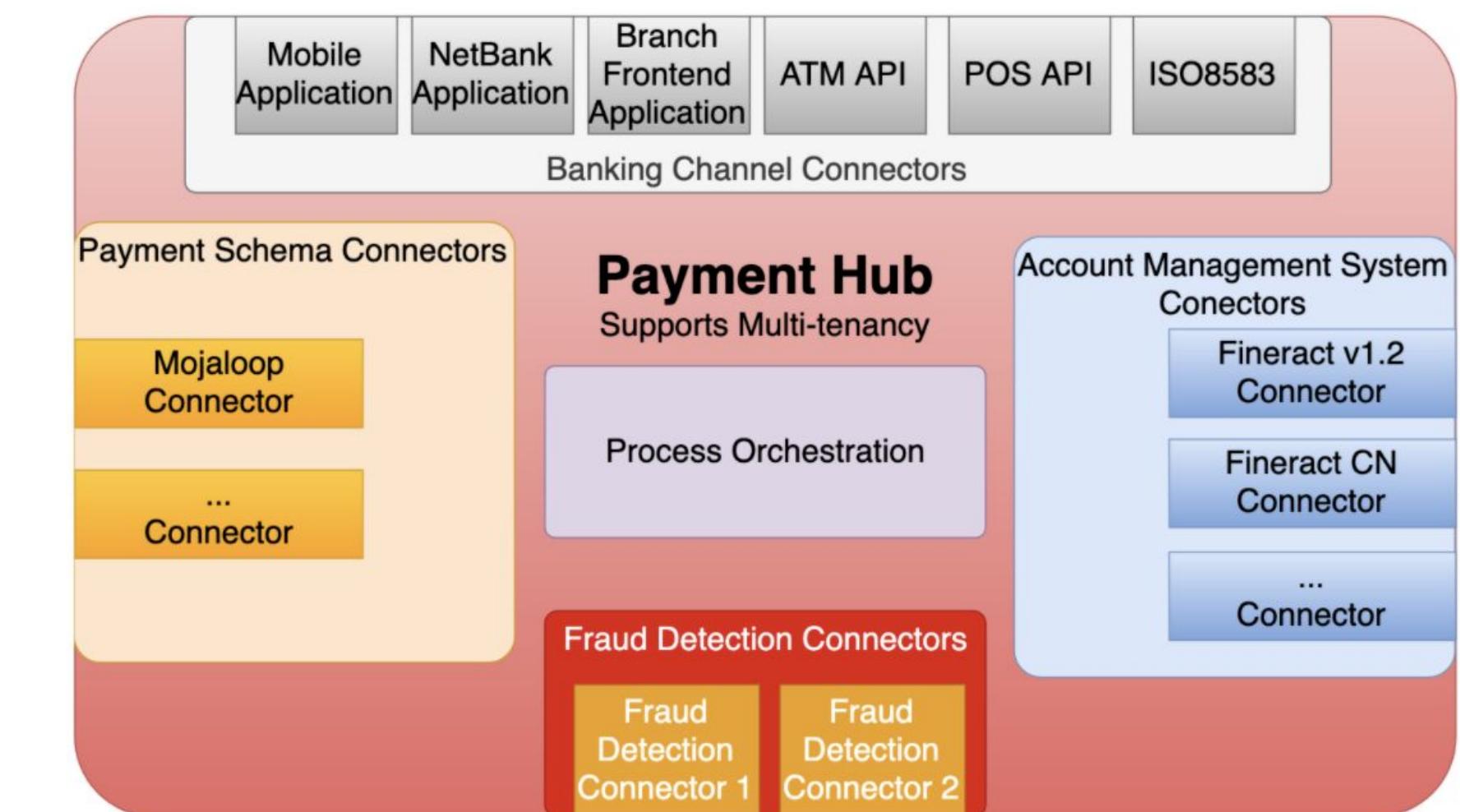
Technical Overview

- Deployment models
- Lab Environment
- Source code
- Installation instructions
- Tuning and performance

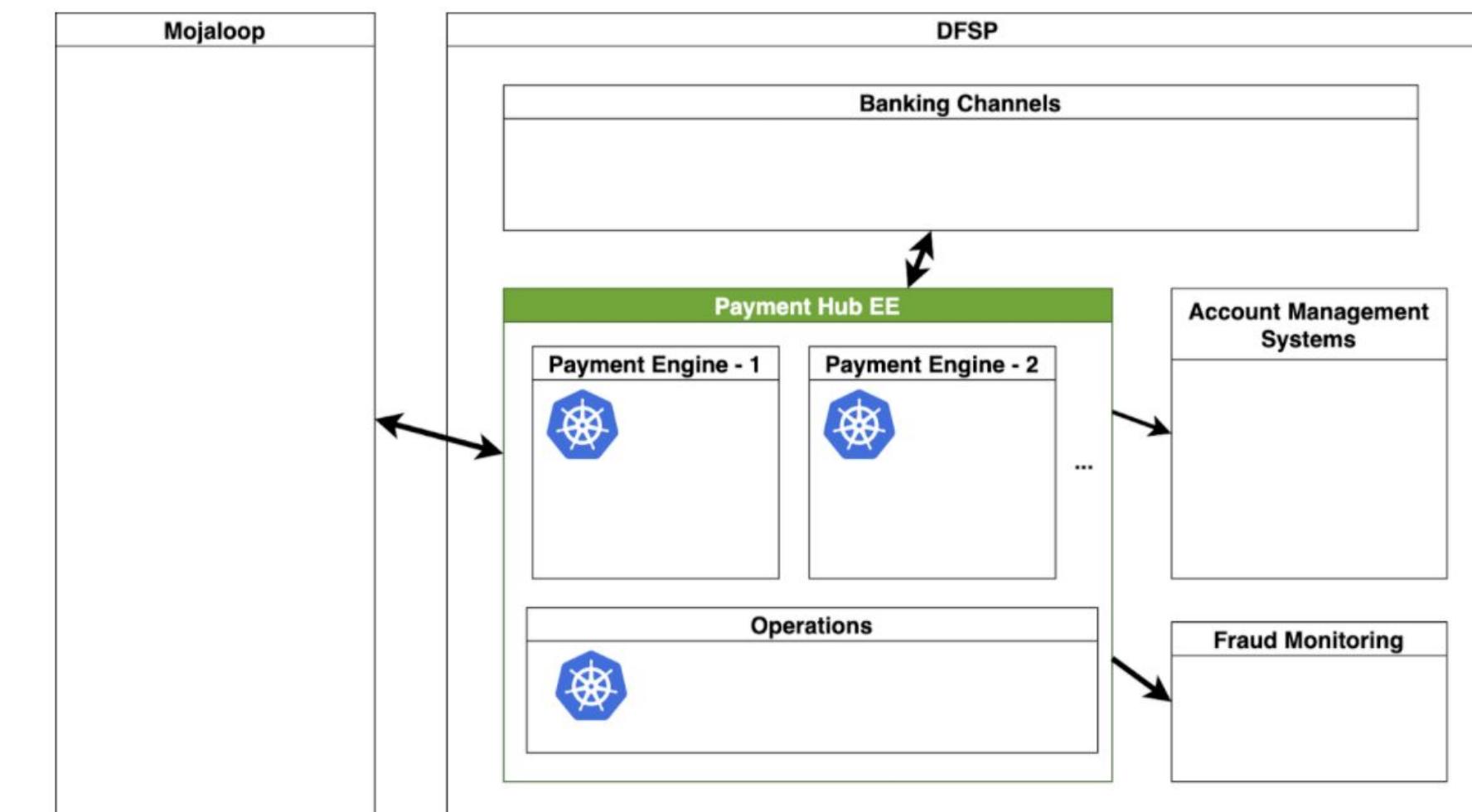
mojaloop
foundation

The screenshot shows a digital documentation interface for a 'LionFintech FPP demo application'. The left sidebar includes navigation links for Edits, Activities, Share, Design, Teams, Integrations, Analytics, and Advanced. The main content area displays the 'Business Overview' section under 'PAYMENT HUB EE', which includes sub-links for Vision, Benefits of Payment Hub EE, Target Users of Payment Hub EE, Using Payment Hub EE, and Use cases. A 'Technical Overview' section is also present. At the bottom, there are links for New and Files.

Design



Logical Model of the Payment Hub



The role of the Payment Hub EE at a DFSP

Thank You

- Miller Abel, Kim Walters & Ariel Delaney
- Core OSS Team

Edward Cable

edcable@mifos.org

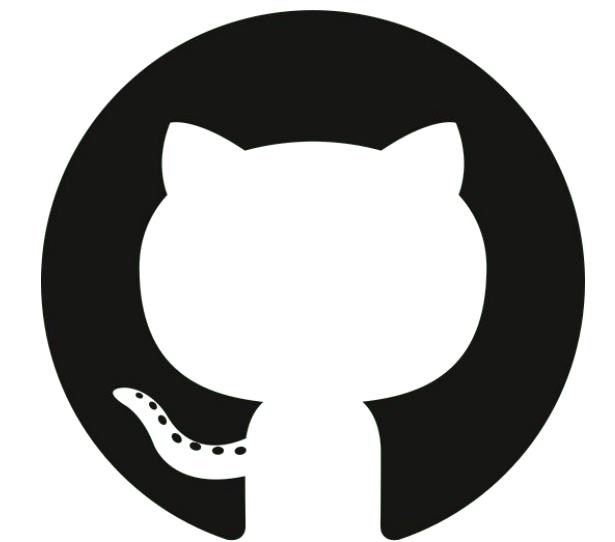
<https://mifos.org>

Istvan Molnar

istvan.molnar@dpc.hu

<https://dpc.hu>

github.com/openMF
github.com/apache/fineract
<https://fineract.apache.org>



[Browse the Docs:](#)

<https://mifos.gitbook.io/docs/payment-hub-ee>

[Explore the Code:](#)

<https://github.com/openMF?q=ph-ee>

[Discuss on Slack:](#) <https://bit.ly/3eMoVS1>

[Request Access to the Lab:](#)

<https://mifos.gitbook.io/docs/payment-hub-ee/overview/lab-environment>

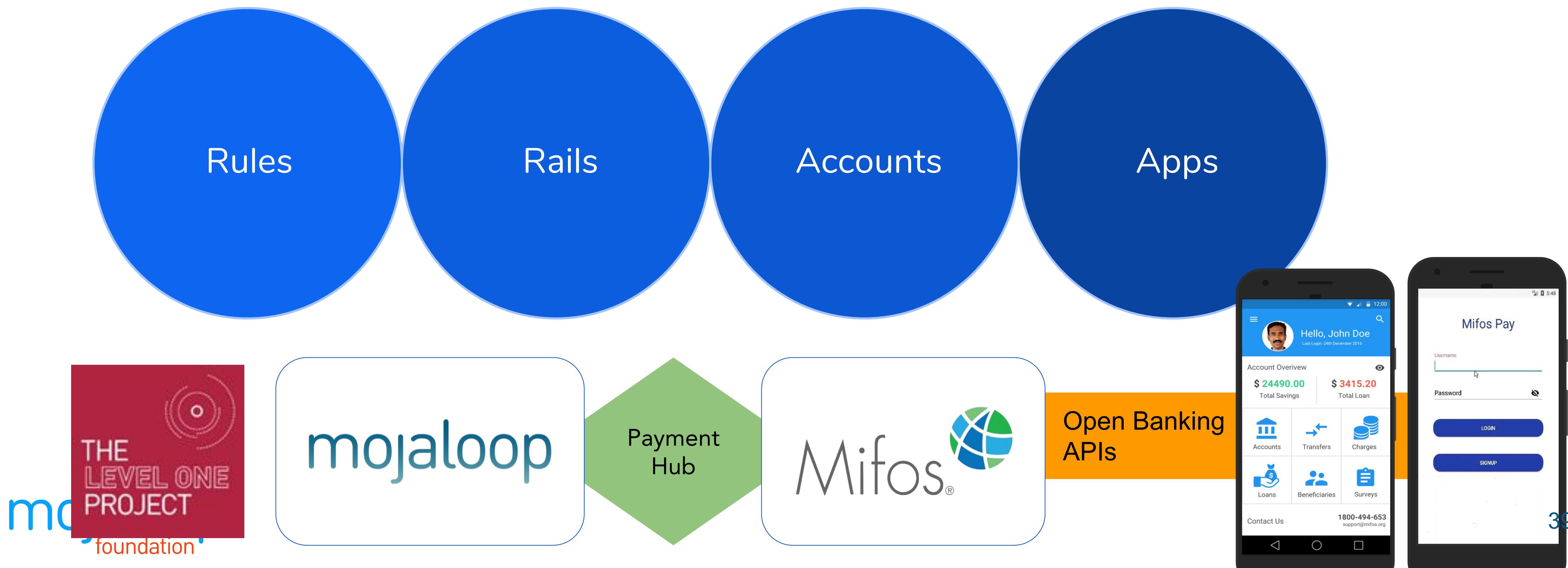


Our Vision

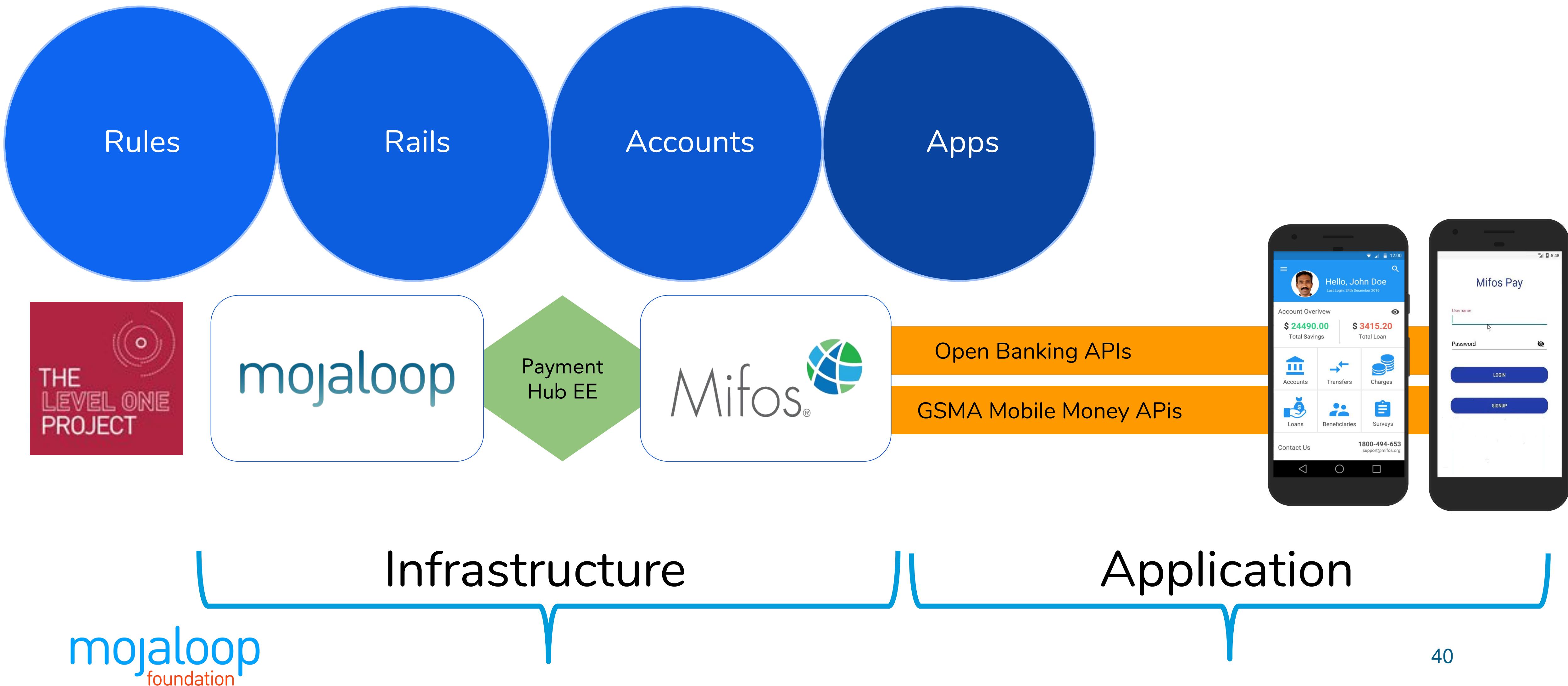
End to End Open Source Stack for Digital Financial Services

. Open Stack

- OS L1-Aligned Payment Switch - Mojaloop
- OS Bridge - Payment Hub
- OS Account Management System - Mifos/Fineract
- OS Reference Mobile Apps - Mobile Banking, Mobile Wallet



Four Layers of APIs at 2 Different Levels



Enabling Access & Meaningful Usage of DFS

MFIs can digitize and digitally transform.

Payment Hub allows simple and low-cost participation in scheme.

Mifos X provides flexible, open production-ready system to digitize all providers, formal & informal.

- Directly offer digital financial services
- Use Open Banking API to partner with fintechs

Rules

Rails

Accounts

Apps

Payment Hub facilitates easy connection to Mojaloop

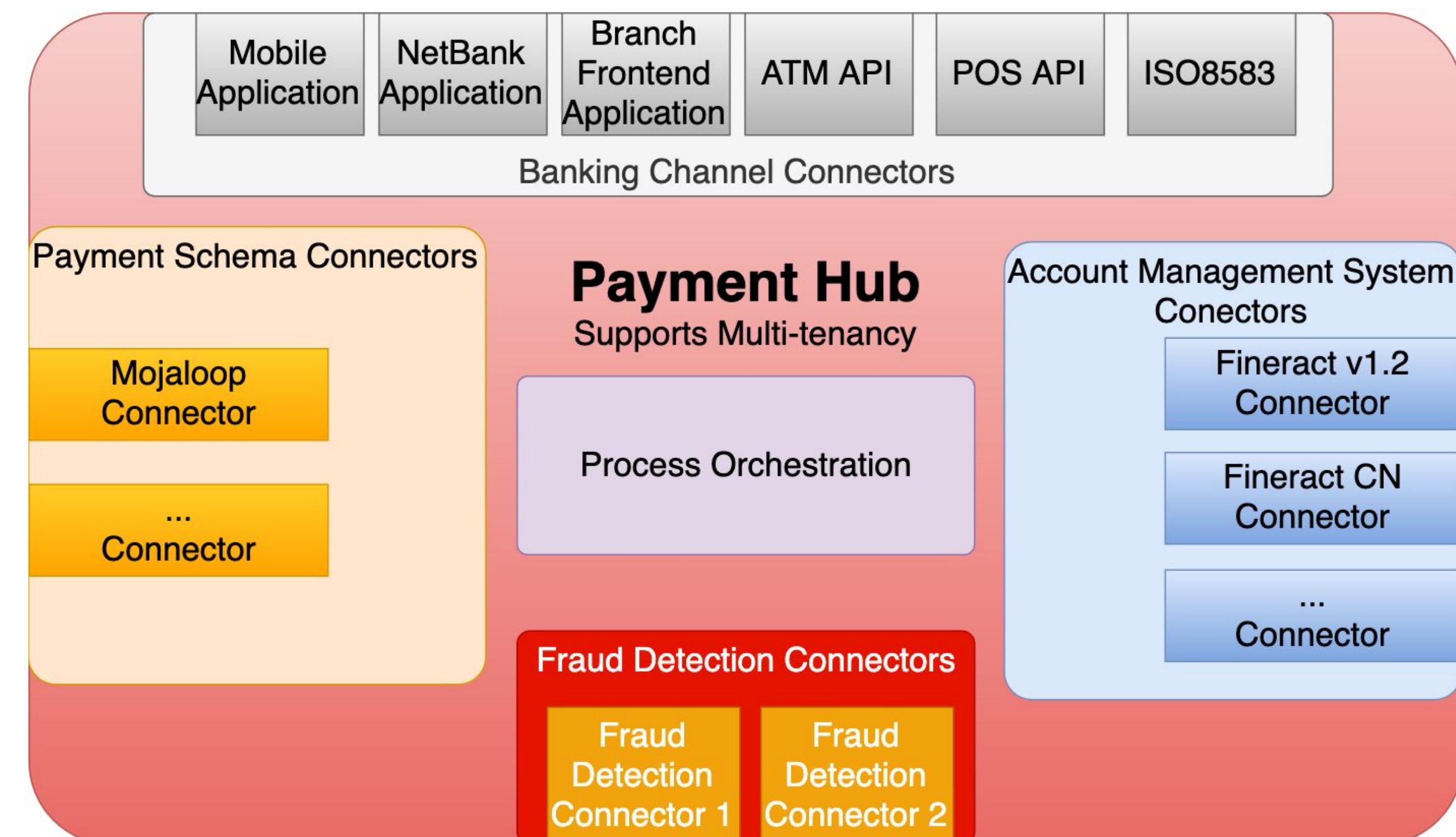
Mobile Wallet Management System with Core Banking Built-in

MMOs can easily roll out adjacent loan & savings services

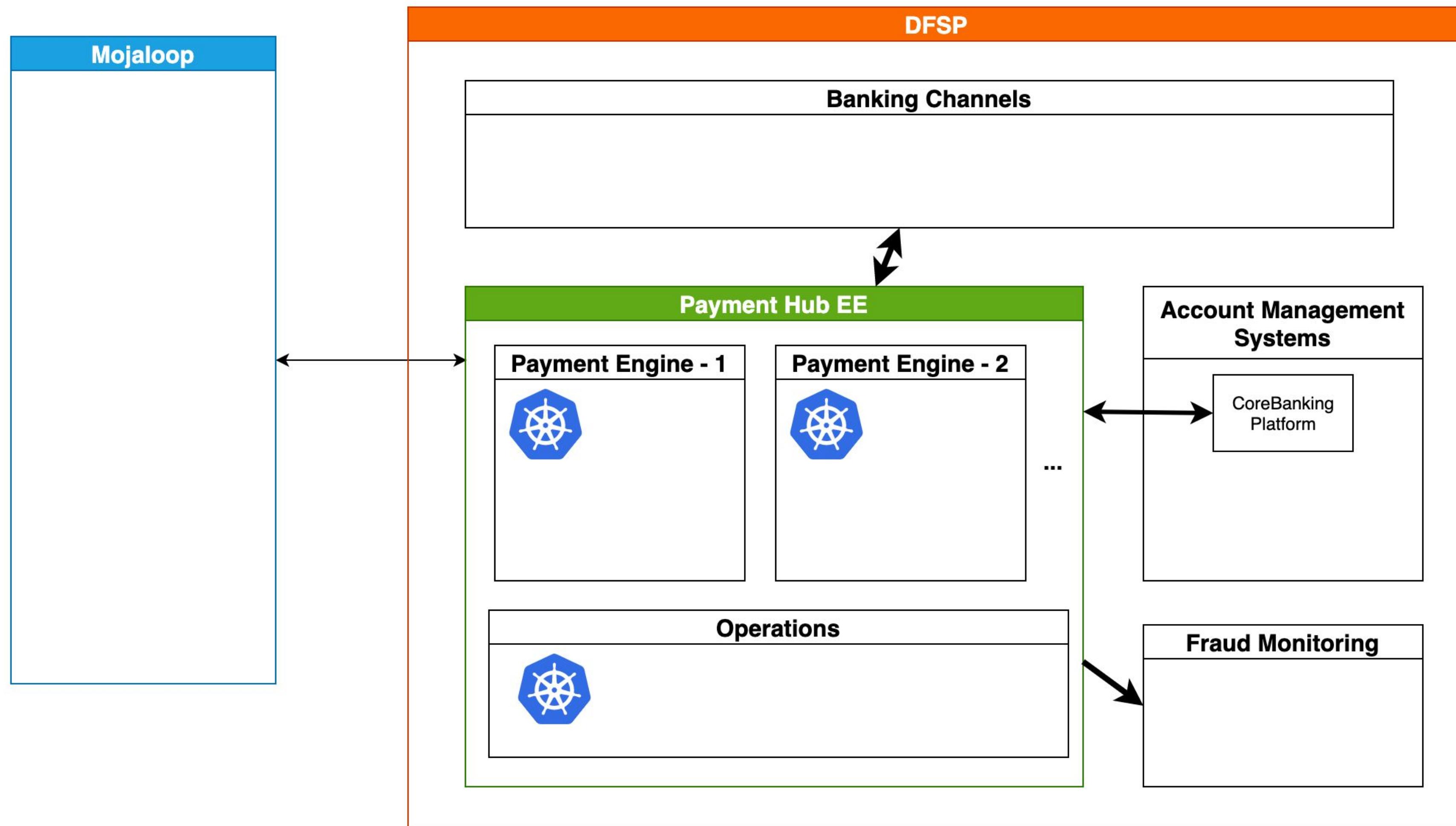
MMOs can evolve to become payment & service platforms

Payment Hub as an Open Source Asset for the Community

- The role of a payment hub to connect:
 - Financial Institution channels (Mobile, Internet, Branch, Callcenter, ATM, POS, API Gateways)
 - Account Management Systems (AMS / Core banking platform), optionally fraud monitoring tools
 - Payment Schemes, such as Mojaloop
- Need
 - Consistent Way to Connect to Mojaloop
 - Effective Operational Participation
- Additional Capabilities
 - DFSP-level fraud monitoring
 - Bulk Transfer Campaign Management
 - Operational Monitoring
 - Manages the identifier – account relation
 - Trigger notifications
- Built on proven open-source technology:
 - Java, SpringBoot, Kafka, Elasticsearch
 - Apache Camel, Camunda Zeebe
 - Kubernetes



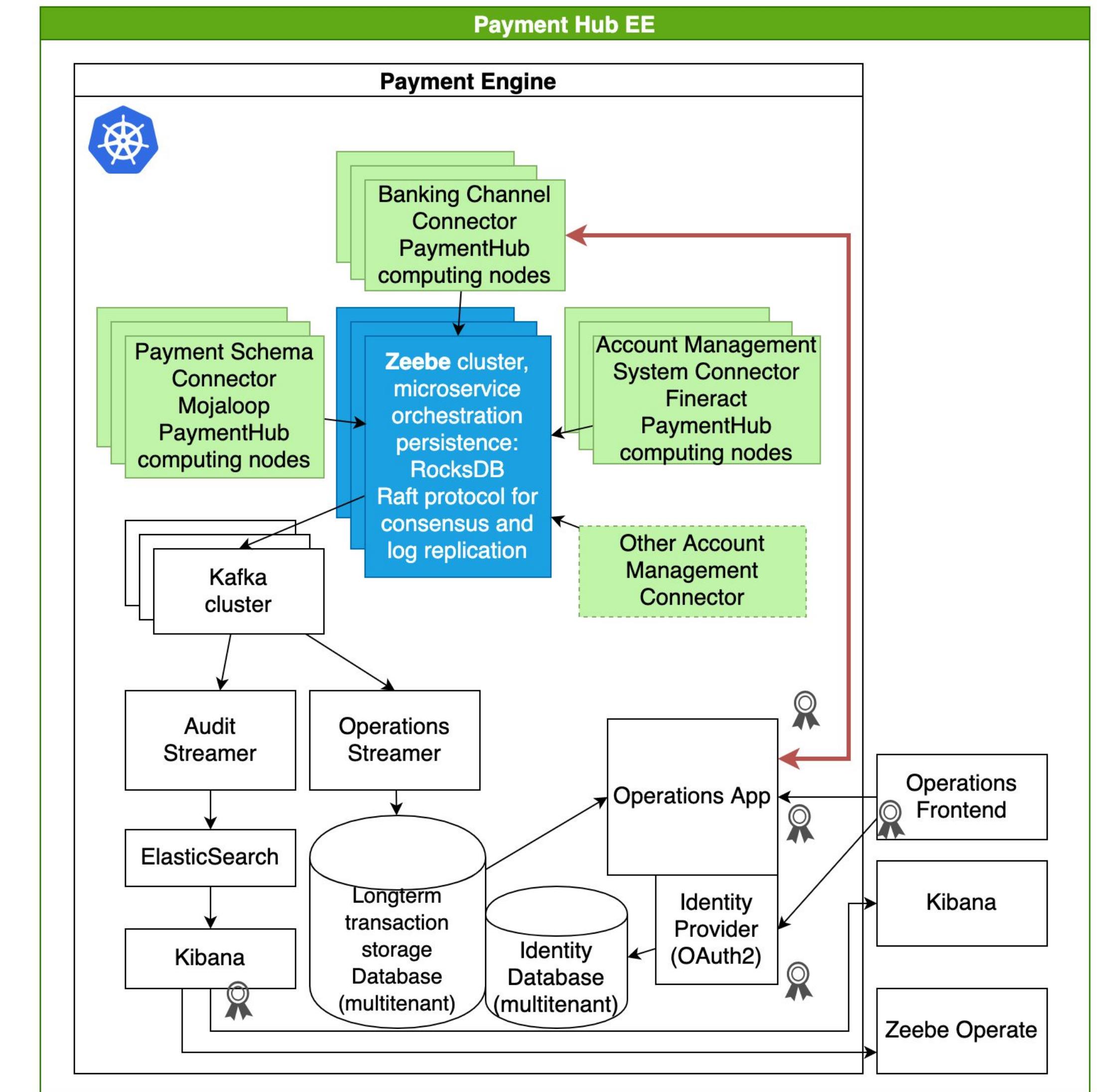
Payment Hub EE in the Payment Context



Payment Hub Components

Microservices based architecture to support easy customization for integrators

- microservices orchestration
- microservices for the connectors
- running in a Kubernetes cluster
- operations application for DFSP command center



Accomplishments for PI-10

Process flow enhancements

- . Account Registration Process - Party Identifier Registration
- . Payer-Initiated Transaction (P2P)
- . Payee-Initiated Transaction (Request To Pay transactions)
 - Automatic and payer approvals according to the corresponding scenario

Multitenancy

Operational Control Center for DFSP actions

Integrated LAB environment with Digital Channels and Fintechs with Openbanking API

Documentation



Moving into Production

An extensible, enterprise-grade integration with Mojaloop

Payment Hub EE is Production Ready

- ❑ Lab Environment is Available for Access
- ❑ Payment Hub is Production Ready
 - ❑ Being deployed as bridge for other use cases in Mifos Ecosystem
 - ❑ Being tested out for Mojaloop by fintechs like Kanzu Code
 - ❑ Eager to have it used for connection in live Mojaloop deployments.

Tier 1 and 2 Institutions

Scalable, extensible, enterprise-grade Operation Control Center for DFSPs

- Mifos and Core Banking System agnostic → simply build another connector.
- Not just to abstract out the complexity of API integration
- Operational Control Center for a DFSP - monitor, analyze, and respond
- Scalable and Enterprise-grade, deployable in multiple topologies.
- Extensible - powerful bridge to connect to other payment systems.
- Zeebe Workflow Engine for Microservices Orchestration - Orchestrate any end-to-end workflow across your payments, systems, and channels
- Open Source and extensible - ready to be commercialized by integrators or enhanced and extended by in-house IT
- Stand-in System capabilities



Deployment models

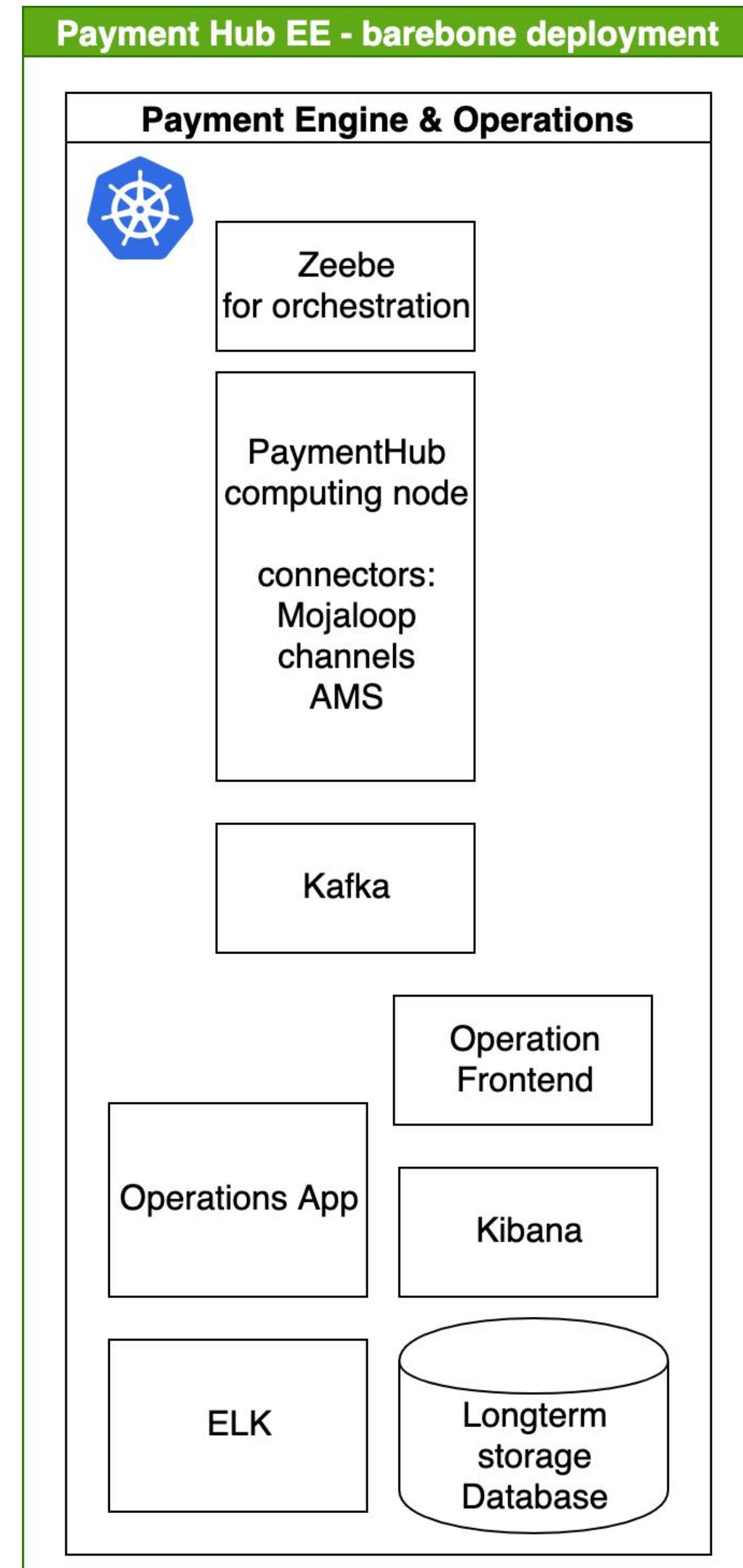
On-premises and any of the cloud providers

Shared service serving multiple DFSPs run by an aggregator (multiple SACCOs, credit unions on a multitenant setup) or dedicated setup

Depending on the DFSP requirements it could be deployed as

- **barebone** - single instance of components, minimized resource usage, no loss of functionality
 - Might not run on a feature phone, but we will get there.
- **medium** - single realtime engine
- **fully scaled** - multiple realtime engines

The difference is in availability, fault tolerance and the volume of transactions, which can be handled.



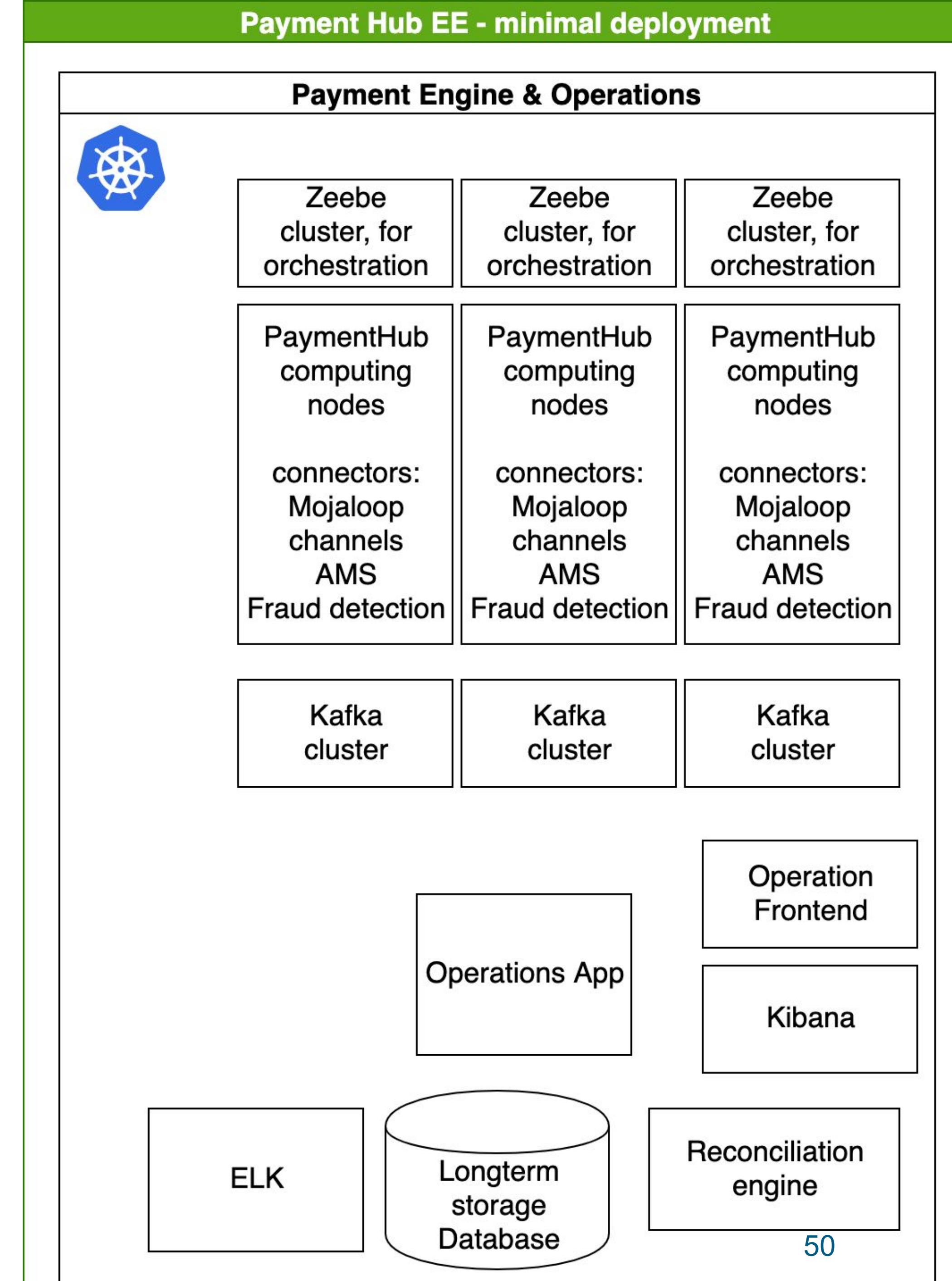
Deployment model - medium

Medium deployment

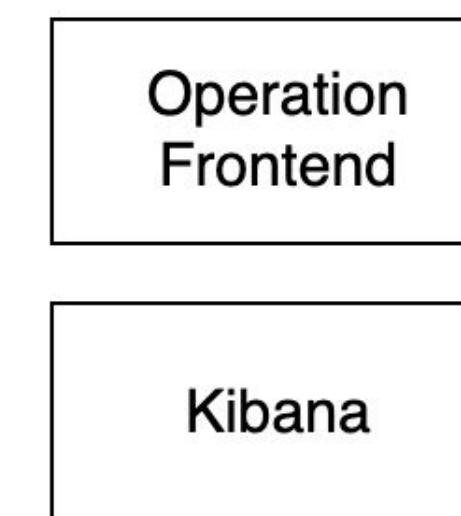
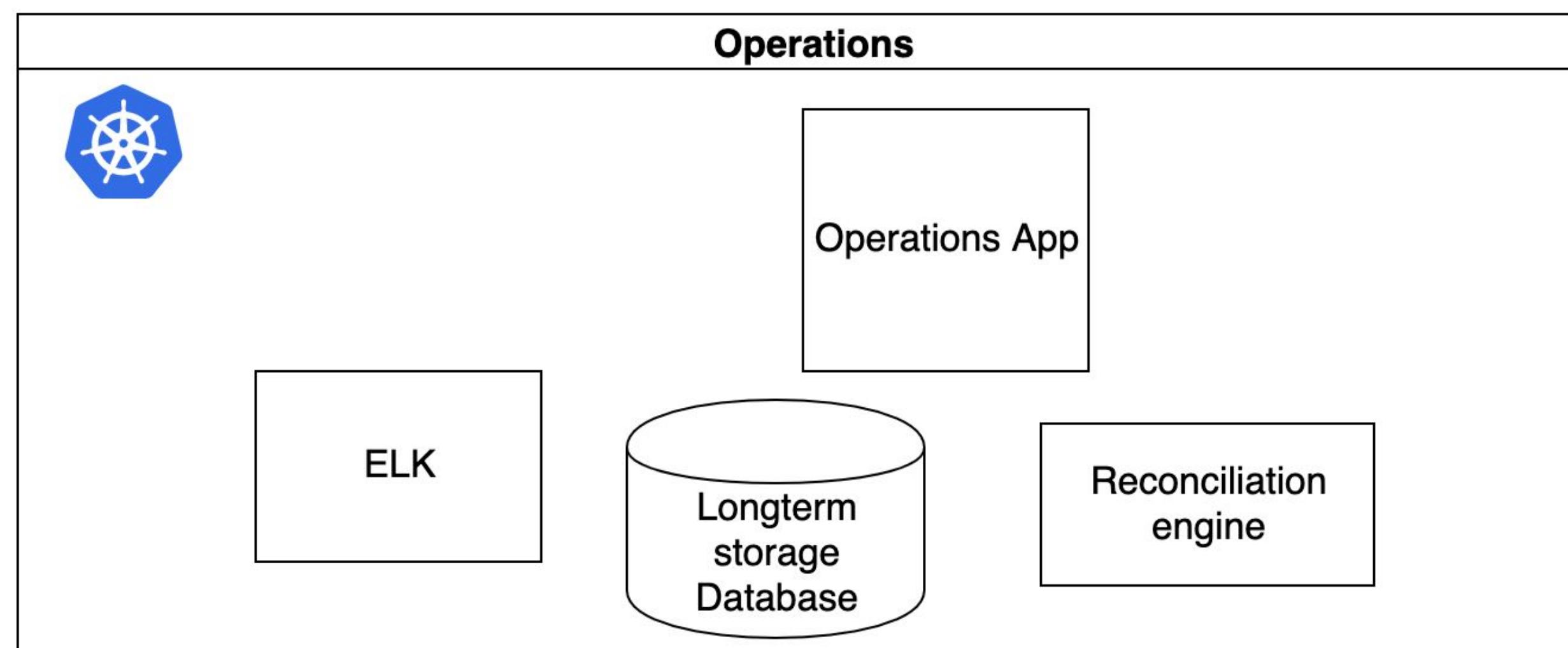
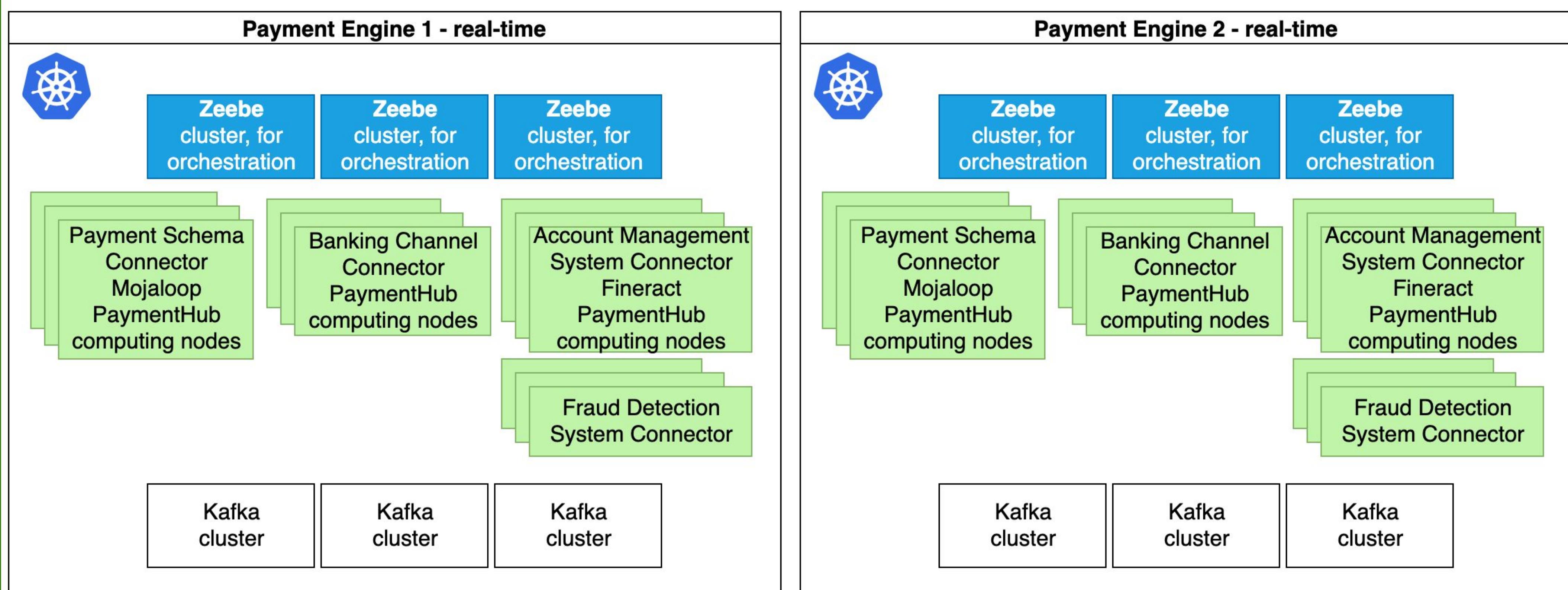
- Single kubernetes cluster to contain all the necessary components
- Fault tolerance provided by the clustered components
- Stretched installation across data centers possible, but not ideal

Full scale deployment

- Multiple independent payment engines (a single engine is collocated for performance), enabling complete version upgrades without service interruptions
- Running in different data centers on independent network connections (high availability, fault tolerant even in case of disaster scenarios)
- Partitioning the load across the engines



Payment Hub EE



Accomplishments for PI-10

Process flow enhancements

- . Account Registration Process - Party Identifier Registration
- . Payer-Initiated Transaction (P2P)
- . Payee-Initiated Transaction (Request To Pay transactions)
 - Automatic and payer approvals according to the corresponding scenario

Multitenancy

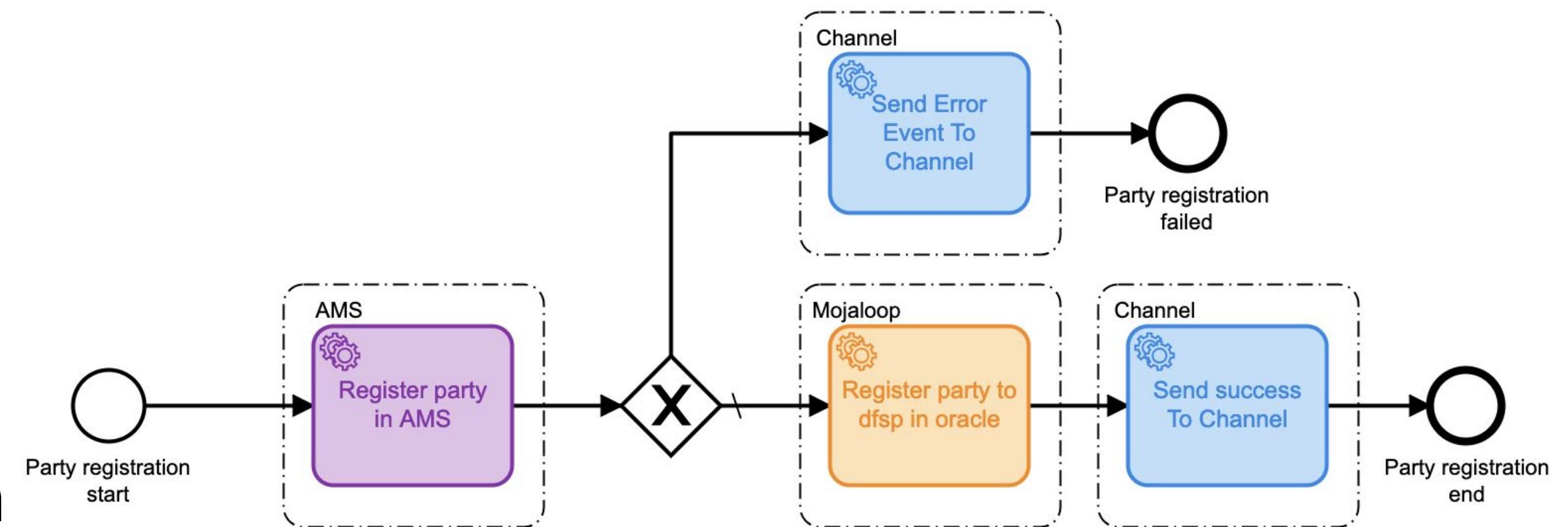
Operational Control Center for DFSP actions

Integrated LAB environment with Digital Channels and Fintechs with Openbanking API

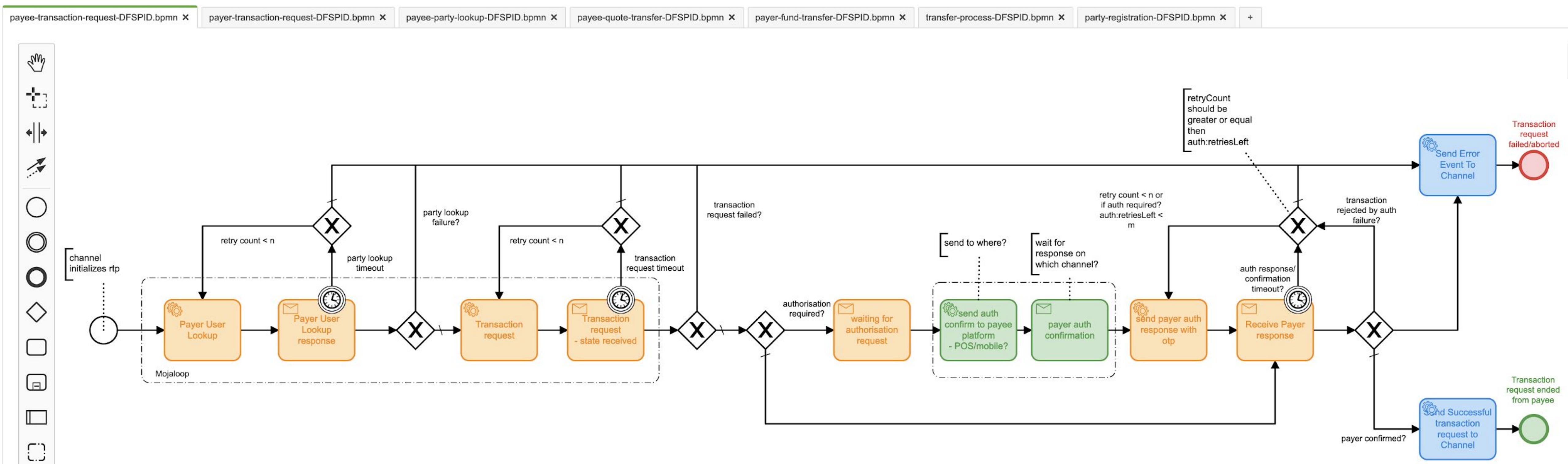
Documentation

Party Identifier Registration

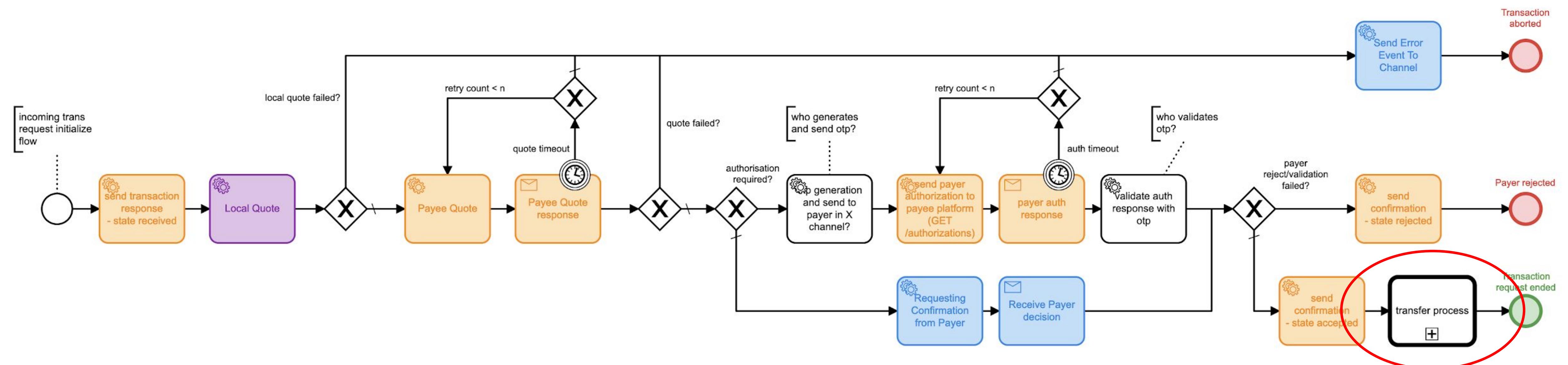
- Initiate the association of a party identifier (MSISDN) with an account at DFSP
- Register identifier in the DFSP systems (in the Account Management System in our environment)
- Manage the registration process with error handling at the “Oracle”



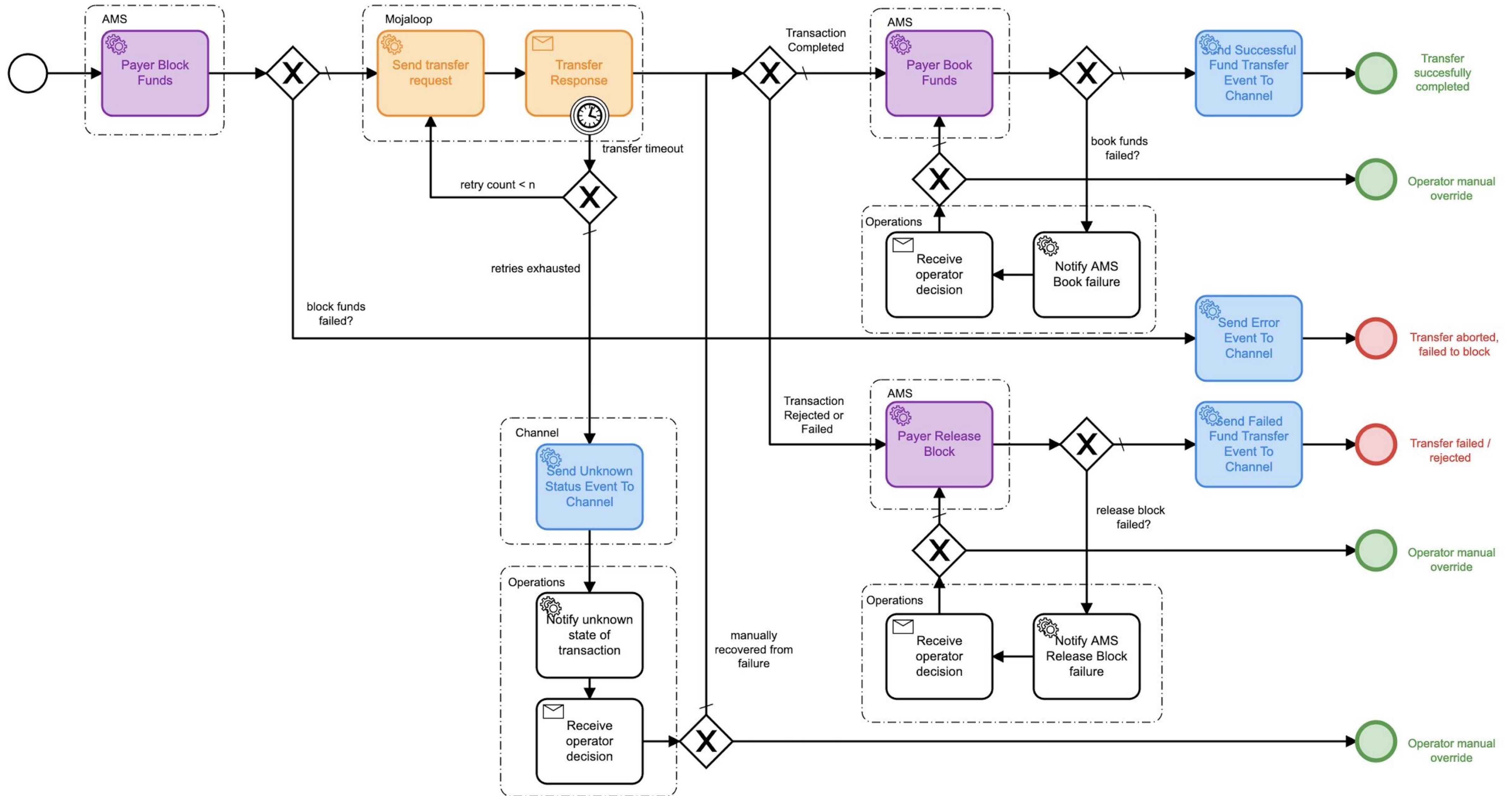
Payee Initiated flows - Request To Pay



Accepting request to pay at Payer



Payer Fund Transfer



Accomplishments for PI-10

Process flow enhancements

- . Account Registration Process - Party Identifier Registration
- . Payer-Initiated Transaction (P2P)
- . Payee-Initiated Transaction (Request To Pay transactions)
 - Automatic and payer approvals according to the corresponding scenario

Multitenancy

Operational Control Center for DFSP actions

Integrated LAB environment with Digital Channels and Fintechs with Openbanking API

Documentation

Operational Control Center for DFSP actions

Authentication and authorization with privileges for different actions

Complete segregation of tenants, separate databases and users

Search and detailed view of transactions

Refund capability for privileged operators on successful incoming transfers

After multiple automated attempts, transactions can be handed over to operations to retry or resolve manually

Search capability for the different flows

The screenshot shows the Mojaloop Payment Hub EE interface. At the top, there is a blue header bar with the following elements from left to right: a green and blue square icon, a back arrow, a 'Payment Hub EE' logo, an 'Admin' icon, a search icon, a language dropdown set to 'en-US', a water drop icon, a bell icon, and a user profile icon.

The main content area has a white background. On the left, there is a sidebar with a house icon. The main area displays the following search options:

- Search Incoming Transactions**
Advanced search option for incoming transactions
- Search Outgoing Transactions**
Advanced search option for outgoing transactions
- Search Incoming Request to Pay**
Advanced search option for incoming request to pays
- Search Outgoing Request to Pay**
Advanced search option for outgoing request to pays

Incoming Payment at Payee

Payment Hub EE Admin Language en-US 🔍 ⚡ 📲

Incoming Transactions | Home / Payment Hub EE / Incoming Transactions

Payer Id	Payer DFSP Id	Payer DFSP name	Payee Id
Transaction ID	Status	Amount	Currency
Transaction Date From	Transaction Date To		

Start Time (UTC)	Completed Time (UTC)	Transaction ID	Payer Id	Payee Id	Payer DFSP Id	Payer DFSP Name	Amount	Currency	Status
2020-07-22 10:54:07	2020-07-22 10:54:15	58f4aea5-8cc4...	27710306999	27710101999	in03tn06	Gorilla Bank	215	TZS	COMPLETED
2020-07-22 08:32:59	2020-07-22 08:33:07	0f9bd223-d91f...	27710306999	27710101999	in03tn06	Gorilla Bank	117	TZS	COMPLETED

Incoming transfers with Refund capability

The screenshot shows the Payment Hub EE interface for an incoming transaction. The top navigation bar includes a logo, 'Payment Hub EE', 'Admin' status, a search icon, language selection ('en-US'), and user icons.

The URL in the address bar is [2251799814249533](#). The page title is 'Home / Payment Hub EE / Incoming Transactions / 2251799814249533'.

Key components of the interface include:

- Payer:** MSISDN 27710306999, DFSP Id in03tn06, DFSP Name Gorilla Bank.
- Payee:** MSISDN 27710101999, DFSP Id in01tn01, DFSP Name Buffalo Bank.
- Transfer:** Transfer Code 58f4aea5-8cc4-404f-98ee-191ef1fc02b9, Amount 215 TZS, Completed 2020-07-22 10:54:15, Status COMPLETED.
- Fees:** Payer quote code, Payer fee, Payee quote code d2f05505-beb0-4a7f-91df-92fb2e99368b, Payee fee 0 TZS.
- Action Buttons:** A red circle highlights the 'Refund' button, and other buttons include 'BPMN Diagram'.

Outgoing Transfer - The Refund

The screenshot shows the Mojaloop Payment Hub EE interface. At the top, there is a navigation bar with a logo, a back arrow, the text "Payment Hub EE", an "Admin" icon, a search icon, a language dropdown set to "en-US", and icons for notifications and user profile.

The main title "Outgoing Transactions" is displayed above the search bar. The search bar includes fields for "Payer Id", "Payee Id", "Payee DFSP Id", "Payee DFSP name", "Transaction ID", "Status" (with a dropdown arrow), "Amount", and "Currency". Below the search bar are filters for "Transaction Date From" and "Transaction Date To".

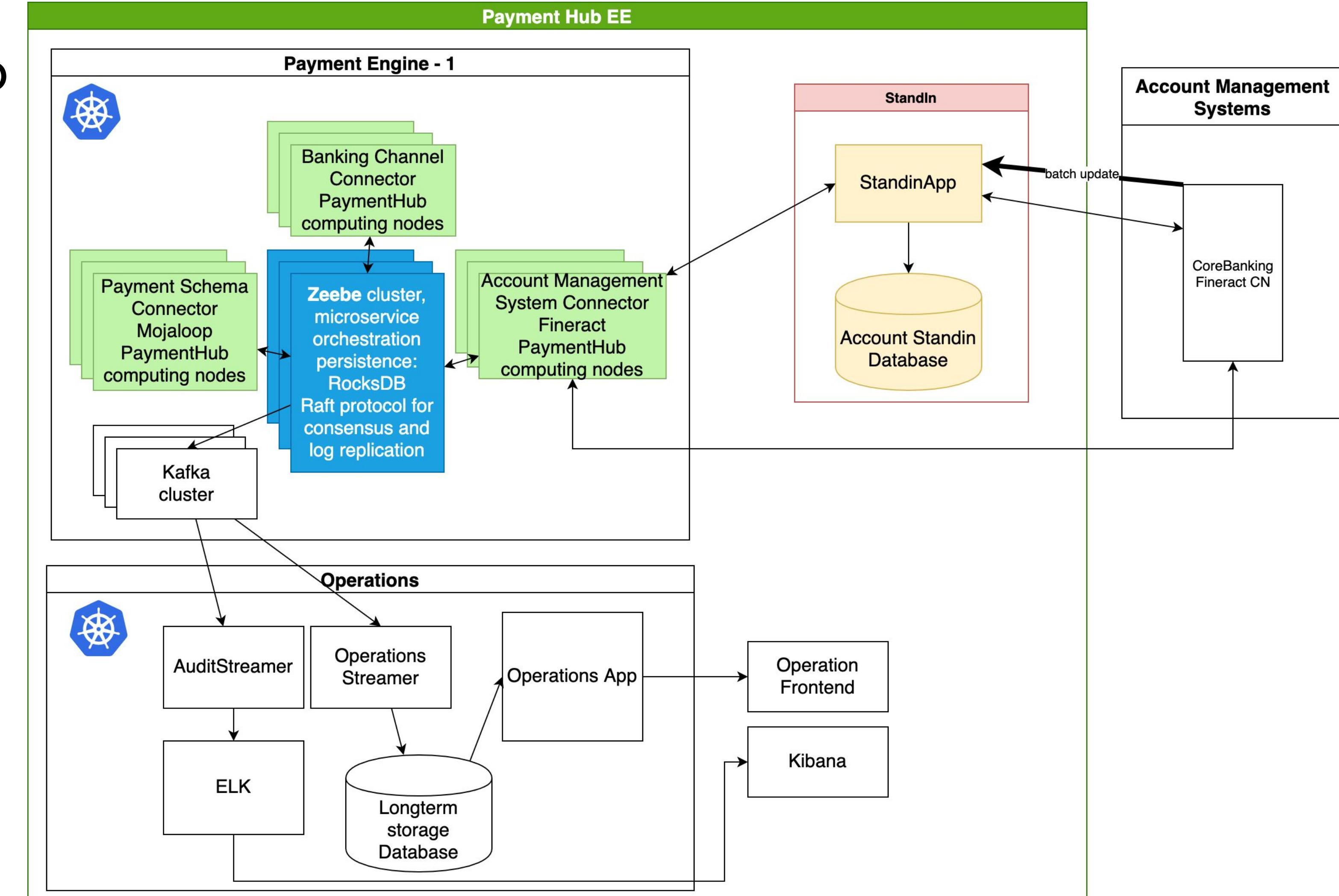
A table below the filters displays transaction details:

Start Time (UTC)	Completed Time (UTC)	Transaction ID ↑	Payer Id	Payee Id	Payee DFSP Id	Payee DFSP Name	Amount	Currency	Status
2020-07-22 11:01:19	2020-07-22 11:01:32	c9b34ebd-dc5c...	27710101999	27710306999	in03tn06	Gorilla Bank	215	TZS	COMPLETED
2020-07-22 08:24:05	2020-07-22 08:24:23	3c158a60-9689...	27710101999	27710306999	in03tn06	Gorilla Bank	199	TZS	COMPLETED

Stand-in

Integrated into the Payment Hub EE solution a stand-in system could provide functionality in case the Account Management System is not available.

- only incoming transactions - simple solution, requires only synchronizing the valid account and party ids
- both incoming and outgoing transactions - requires more complex solution to minimize risk of overdraft



Accomplishments for PI-10

Process flow enhancements

- . Account Registration Process - Party Identifier Registration
- . Payer-Initiated Transaction (P2P)
- . Payee-Initiated Transaction (Request To Pay transactions)
 - Automatic and payer approvals according to the corresponding scenario

Multitenancy

Operational Control Center for DFSP actions

Integrated LAB environment with Digital Channels and Fintechs with Openbanking API

Documentation

Multitenancy

Single deployment serving multiple DFSPs

Isolated audit logs for the different DFSPs served with separate user databases for the DFSP operators

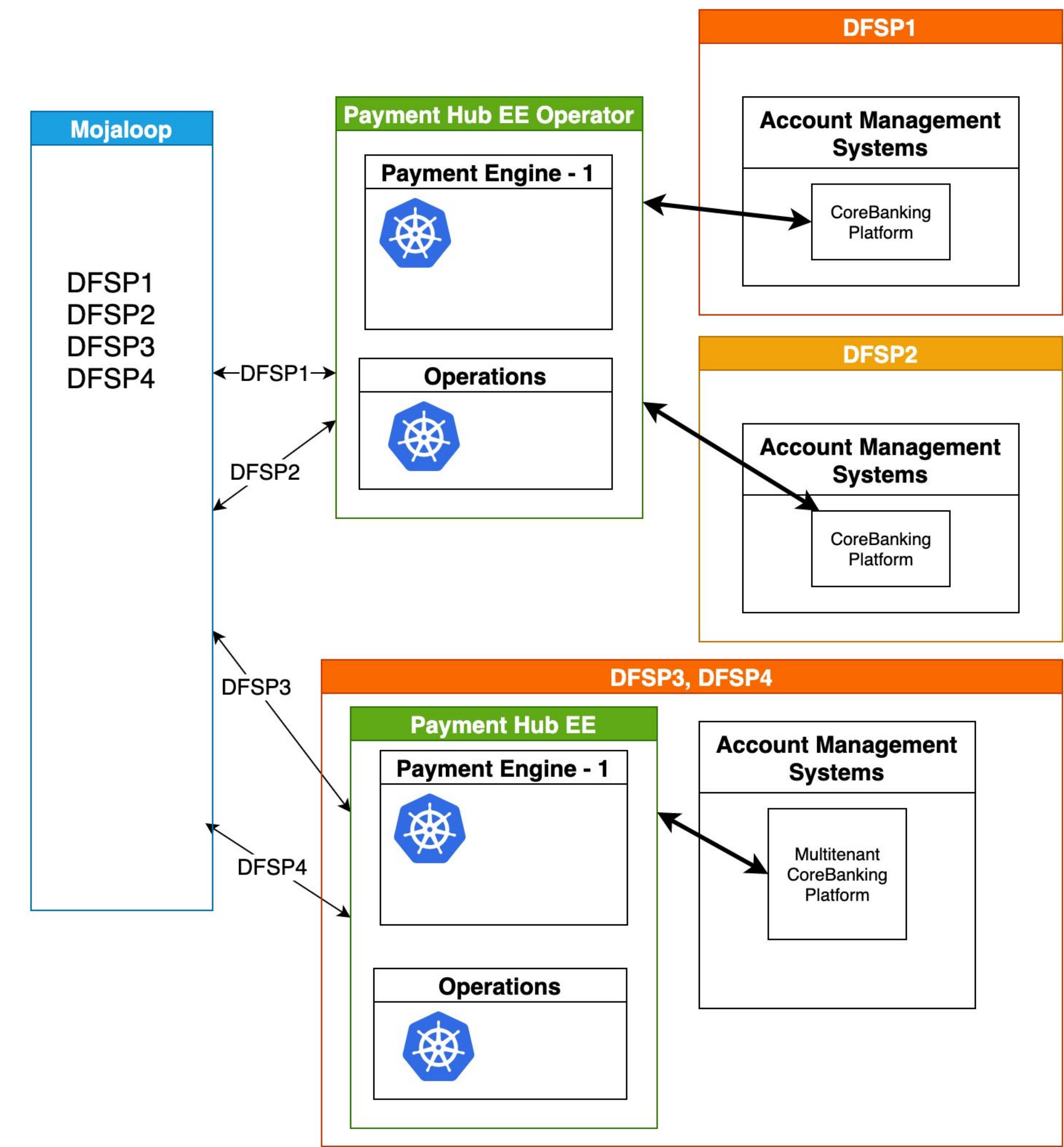
Benefits:

- . Single operator could provide services for multiple individual DFSPs while sharing operational costs
- . In case using a multi-tenant core banking platform (e.g. Mifos) a single Payment Hub EE deployment could serve all tenants (DFSPs)

Multitenancy

Aggregator model for multiple smaller DFSPs

Multi-tenant model for already aggregated DFSPs served from a single multi-tenant core banking

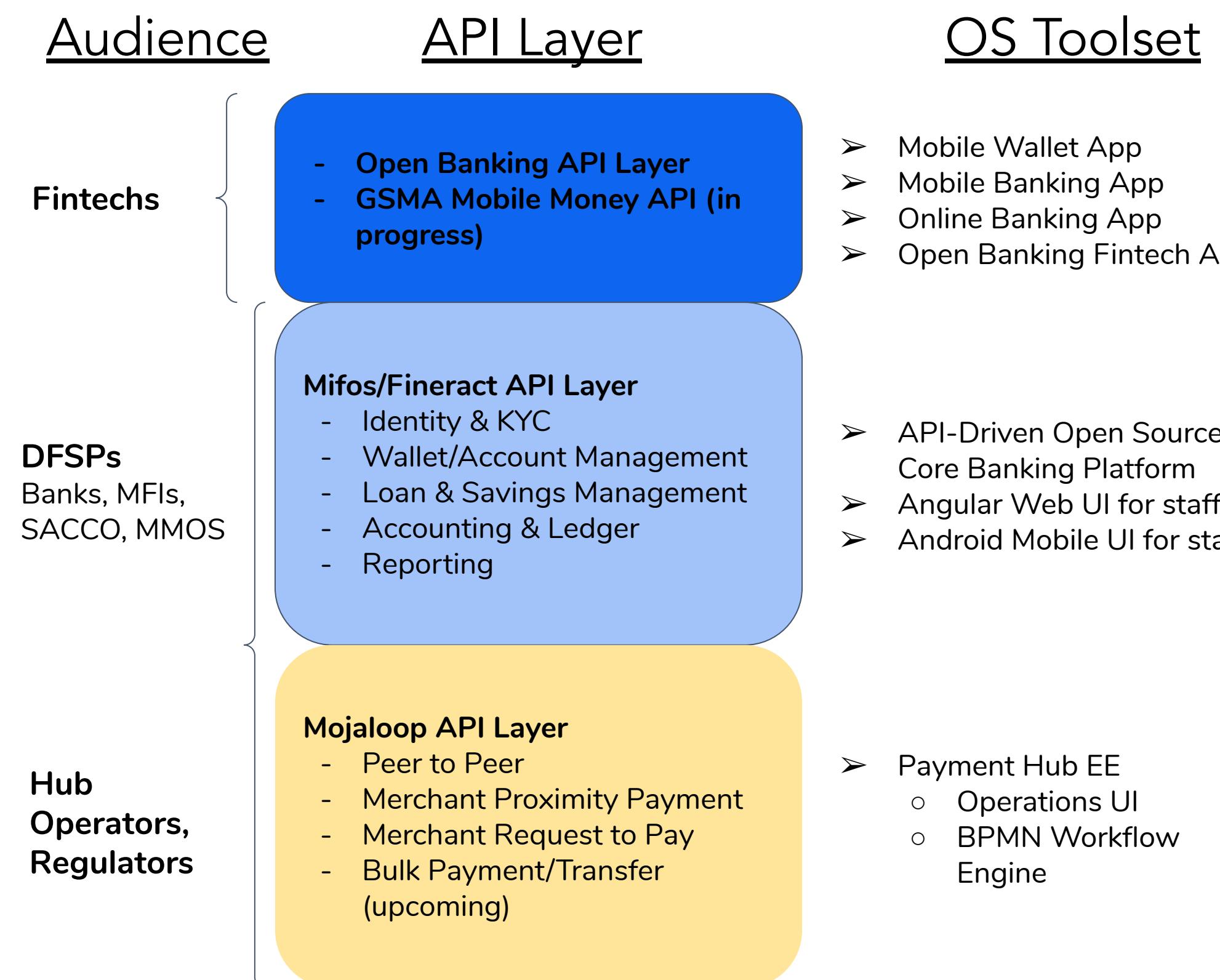
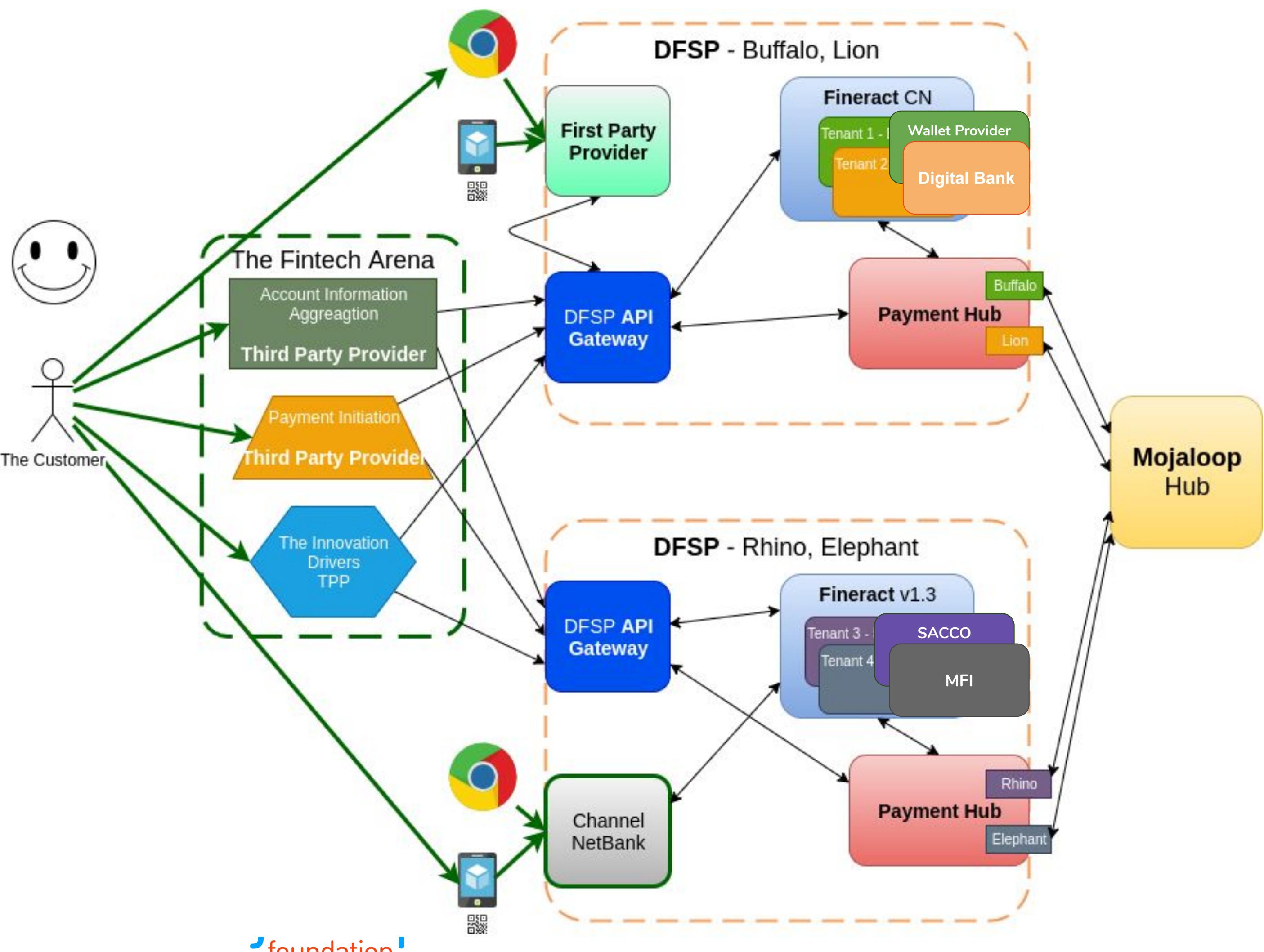




An Immersive User Experience

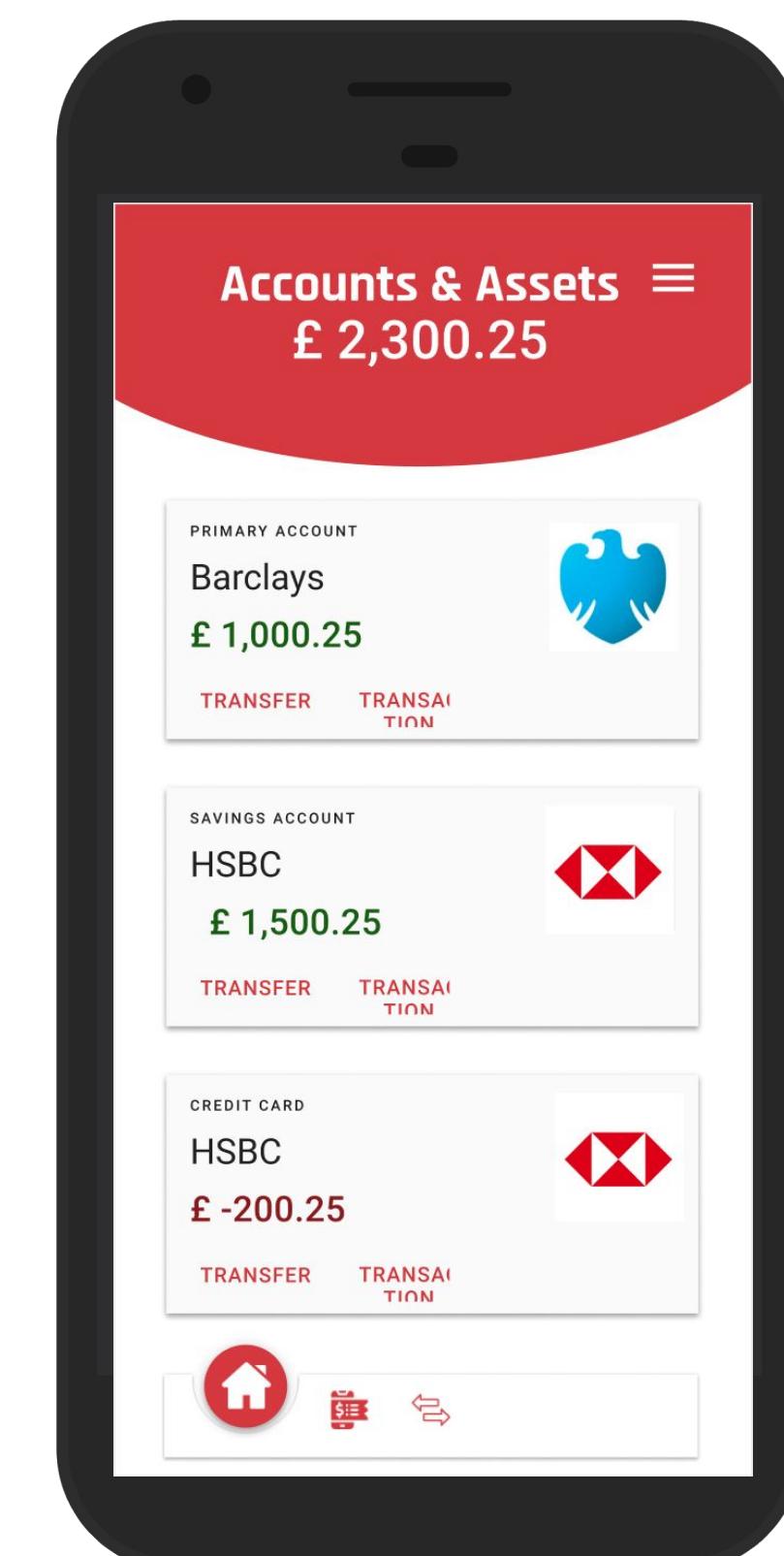
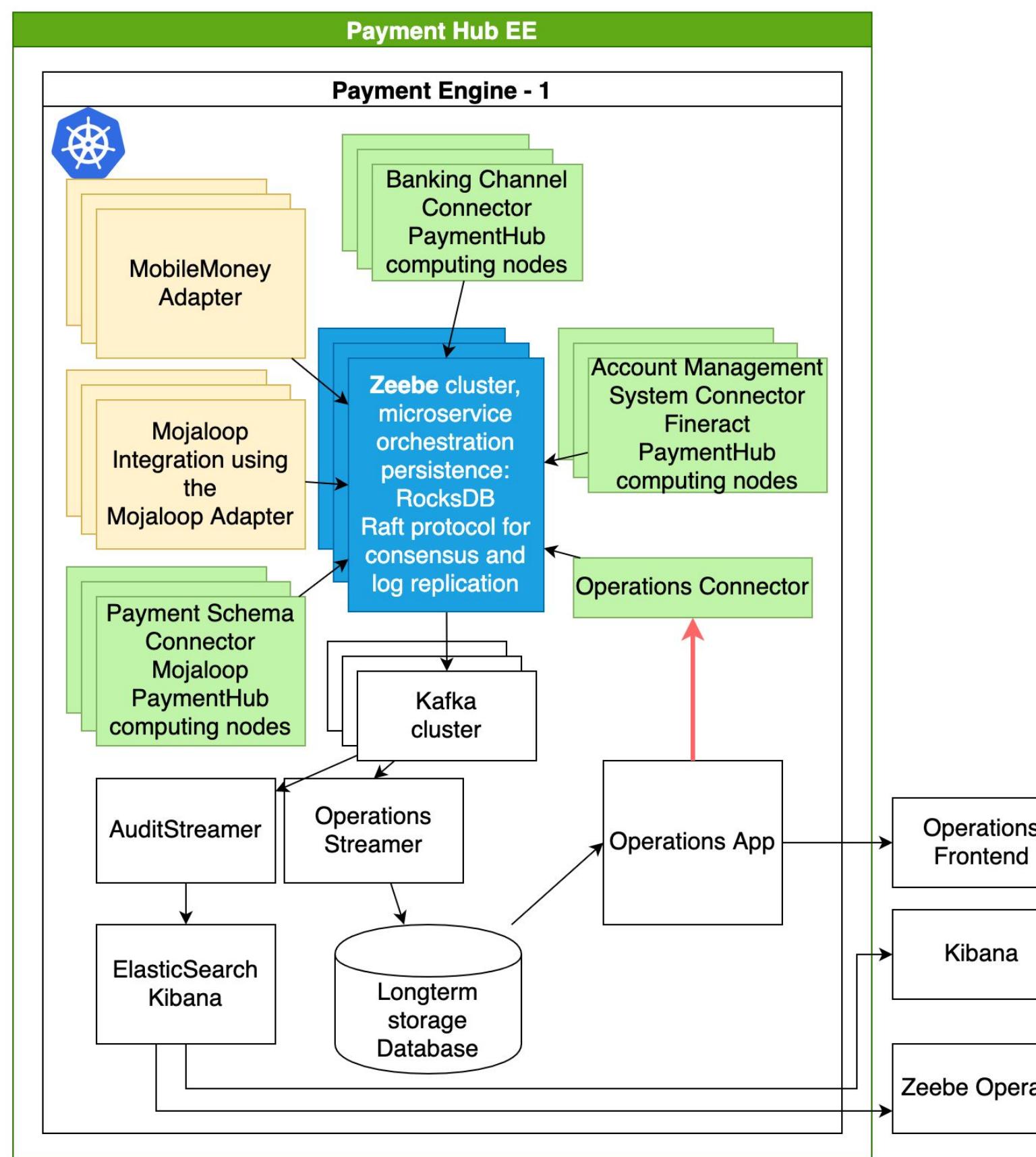
Robust & Accessible Lab Environment

Sandbox Environment for Hackathons

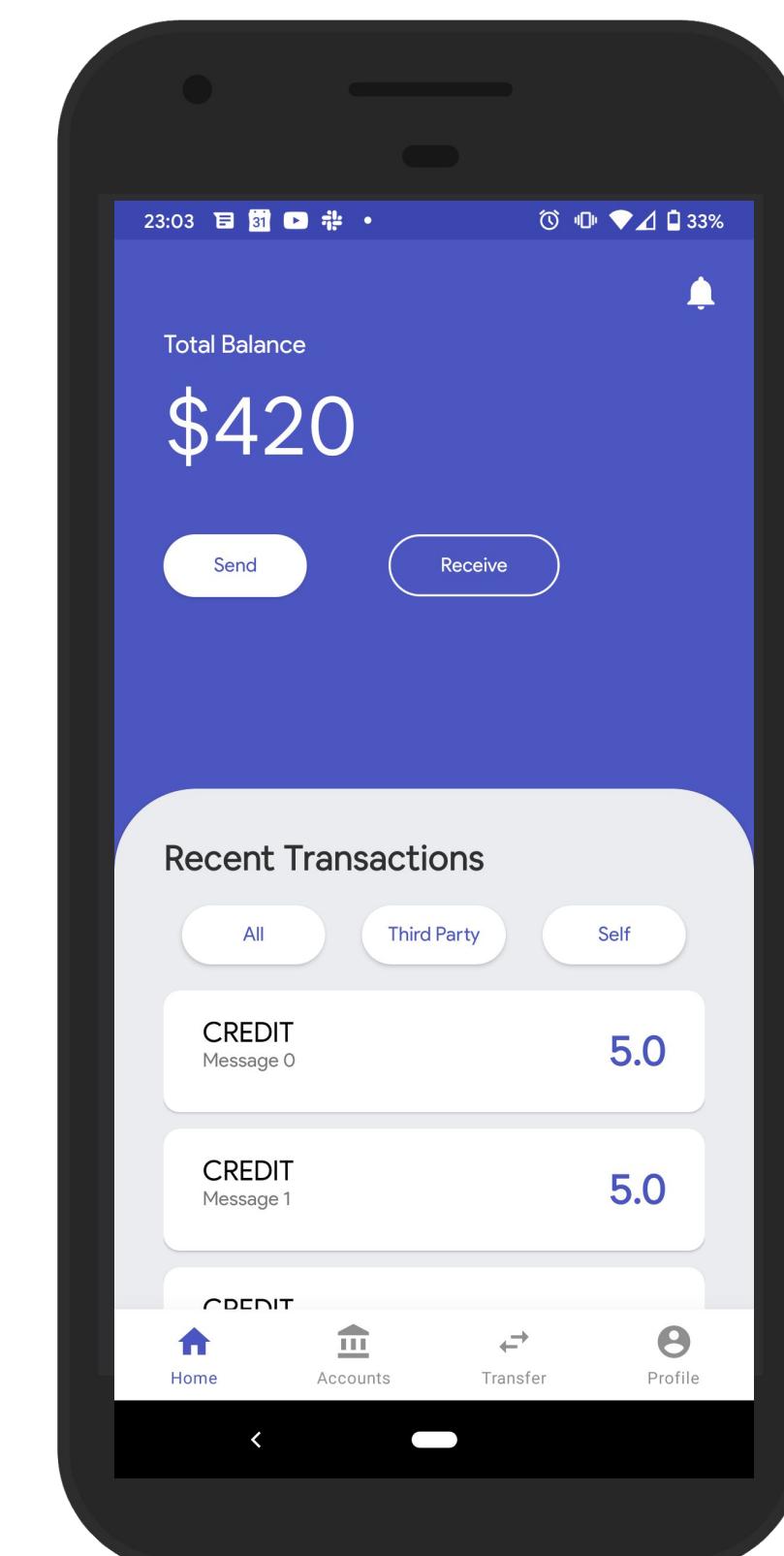


Lab Environment Updates

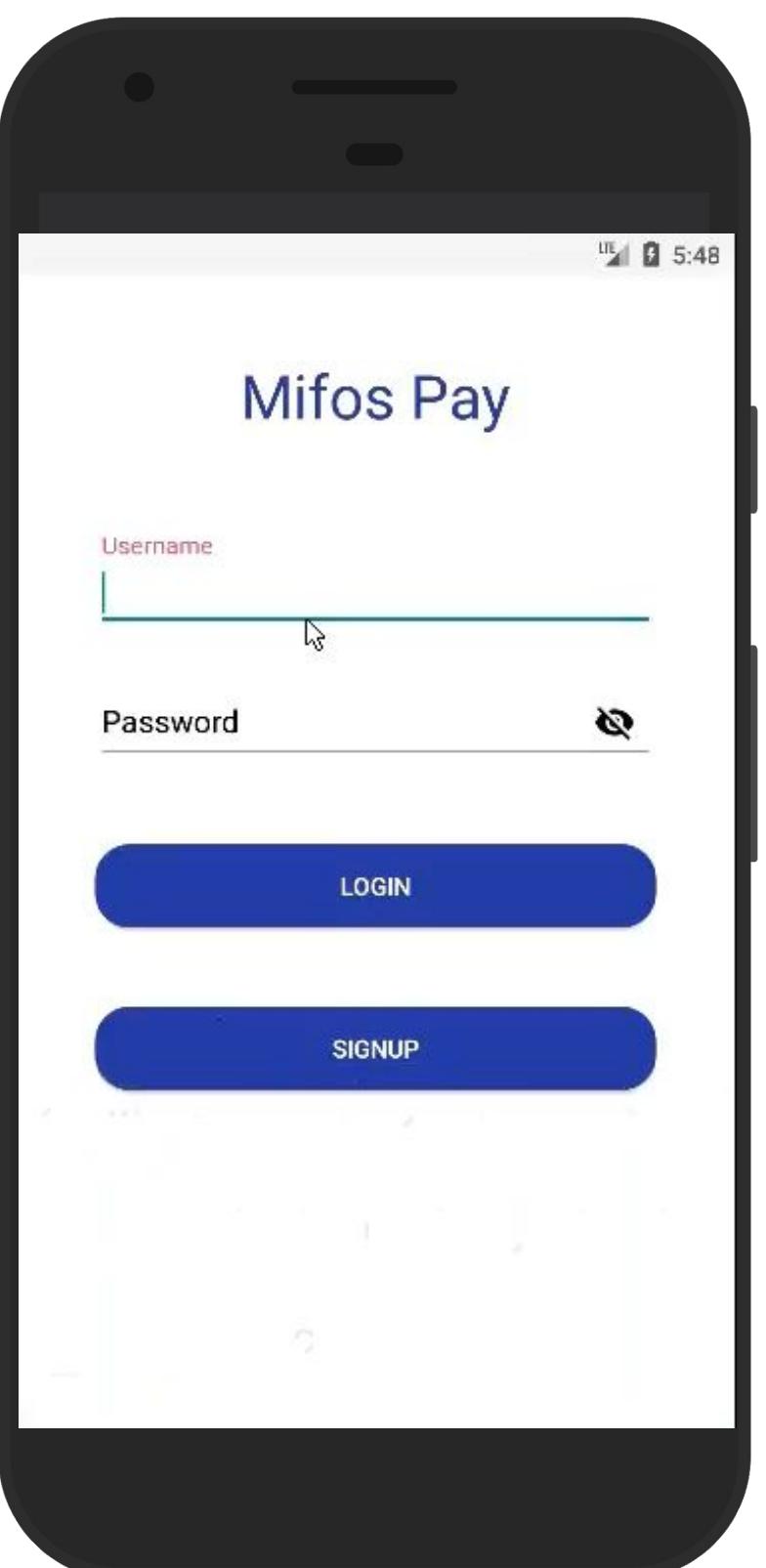
❑ GSMA Mobile Money API Connector



Open Banking App



Mobile Banking



Mobile Wallet

Accomplishments for PI-10

Process flow enhancements

- . Account Registration Process - Party Identifier Registration
- . Payer-Initiated Transaction (P2P)
- . Payee-Initiated Transaction (Request To Pay transactions)
 - Automatic and payer approvals according to the corresponding scenario

Multitenancy

Operational Control Center for DFSP actions

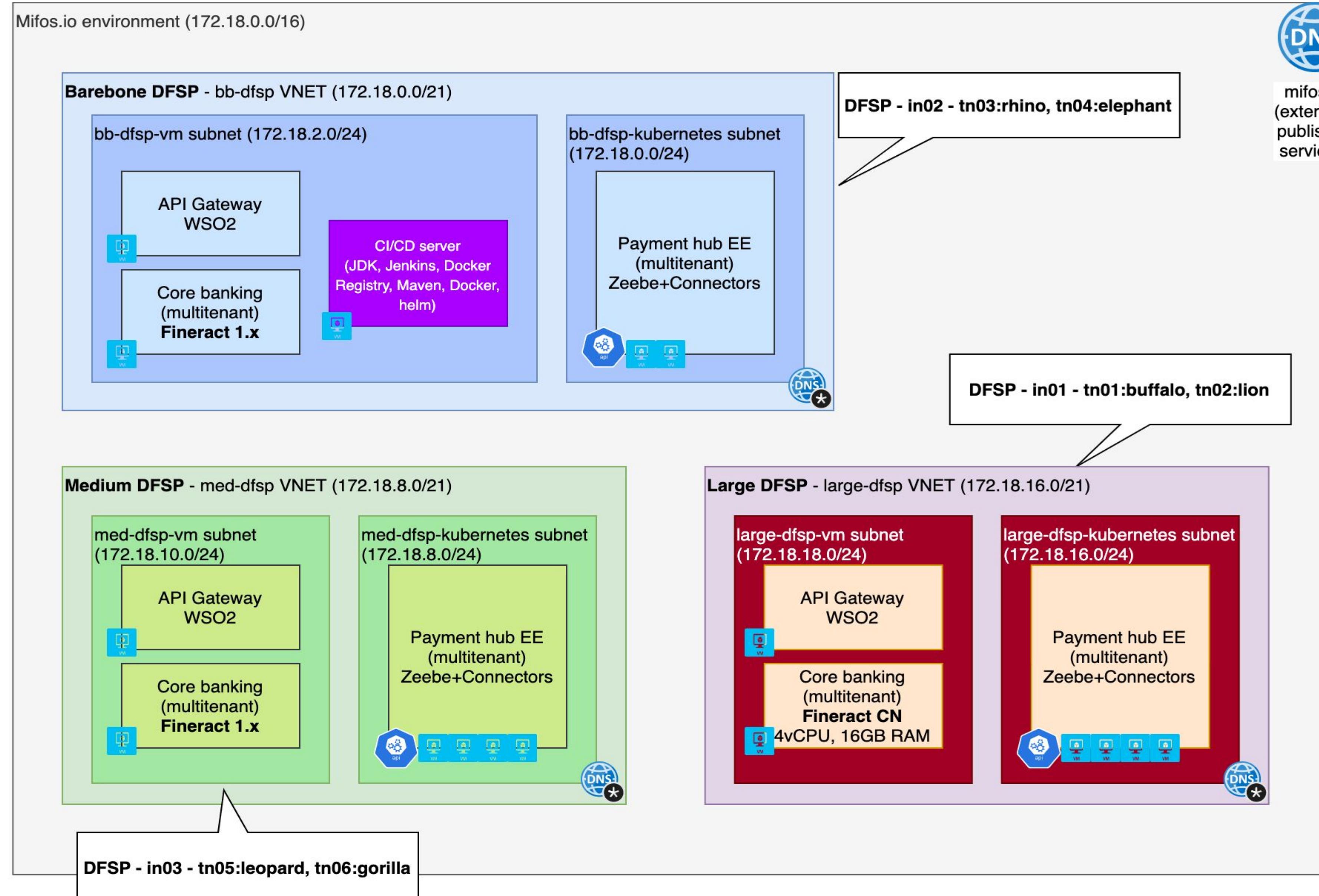
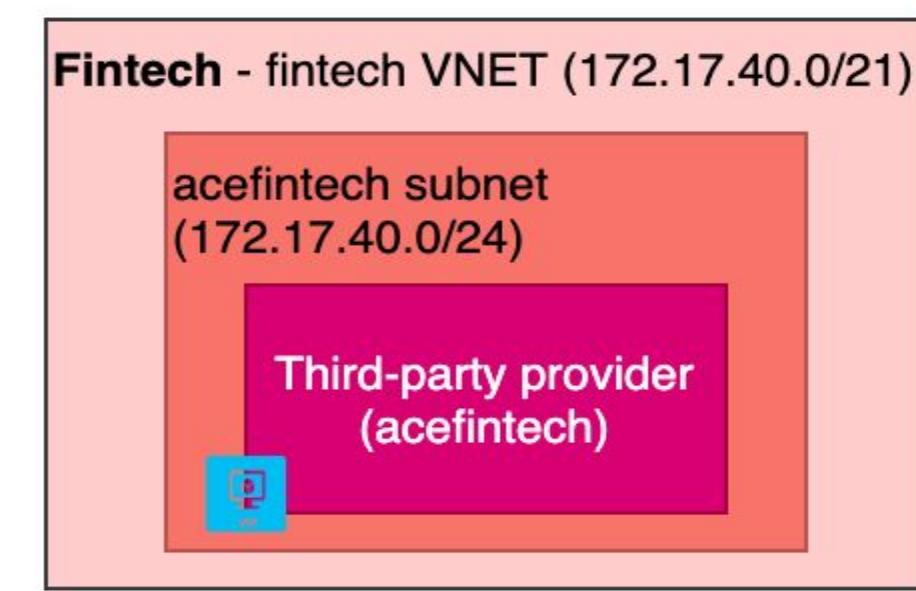
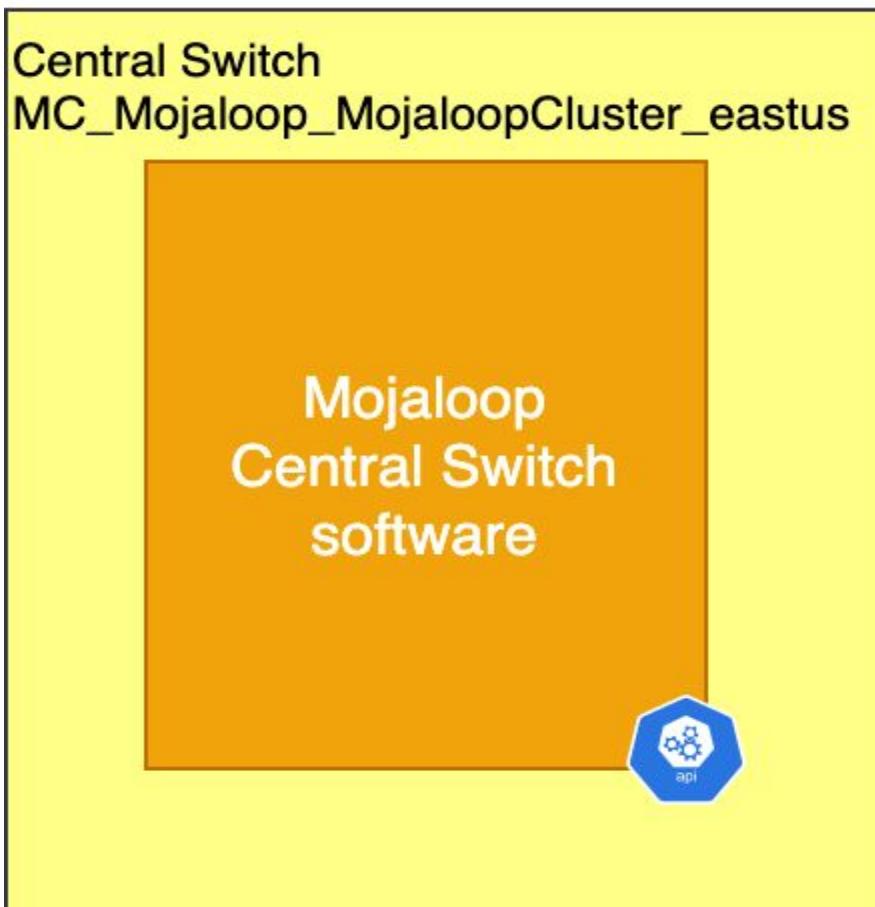
Integrated LAB environment with Digital Channels and Fintechs with Openbanking API

Documentation

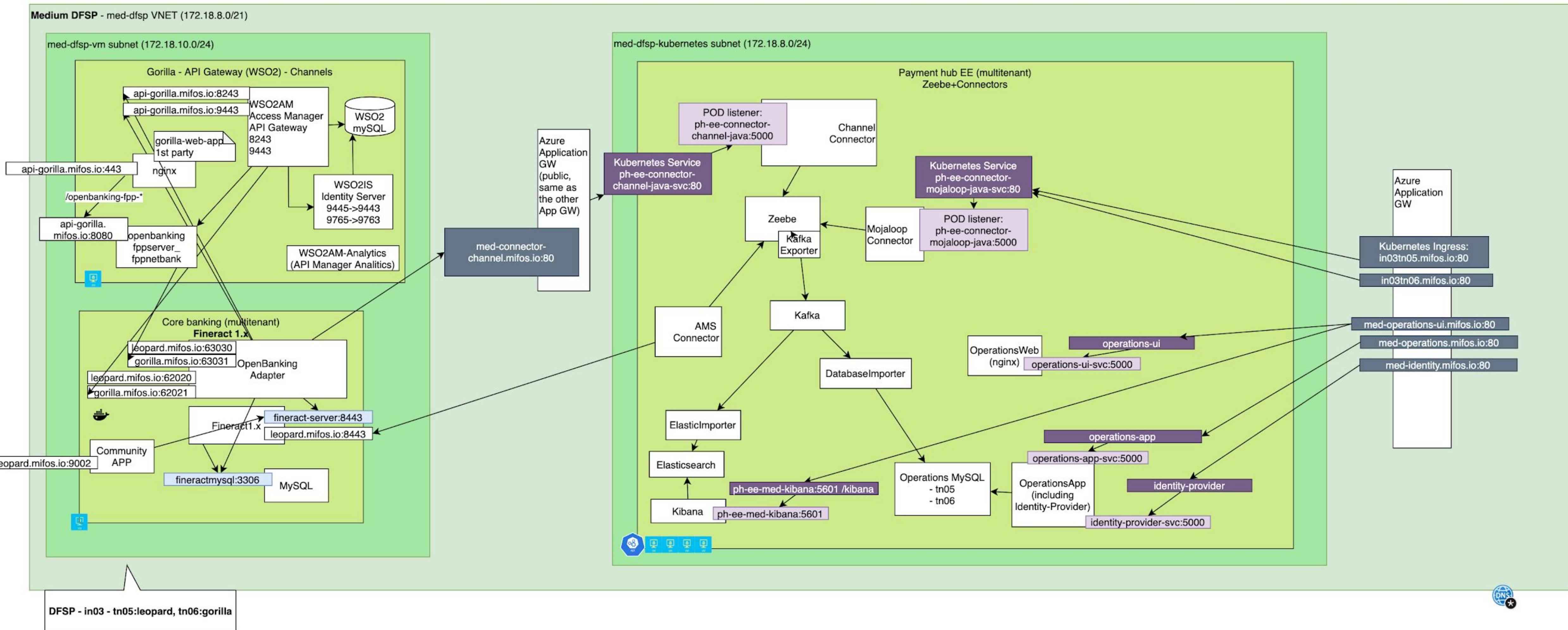
Integrated LAB Environment

- 3 independent deployments
- 2 Mifos 1.x and 1 Mifos CN multitenant core banking deployments
- 2 DFSPs in each deployment unit utilising the multitenant capability of Fineract (one tenant corresponds to a DFSP)
- 1 Mojaloop instance to enable instant payments across the 6 DFSPs
- 1 Fintech application to utilise the OpenBanking APIs provided by the DFSPs. This enables to demonstrate an account information aggregation and payment initiation third party
- 1 CI/CD server to be able to build and deploy the various microservices of the payment hub
- Join the Mifos Slack to get access to the environment!

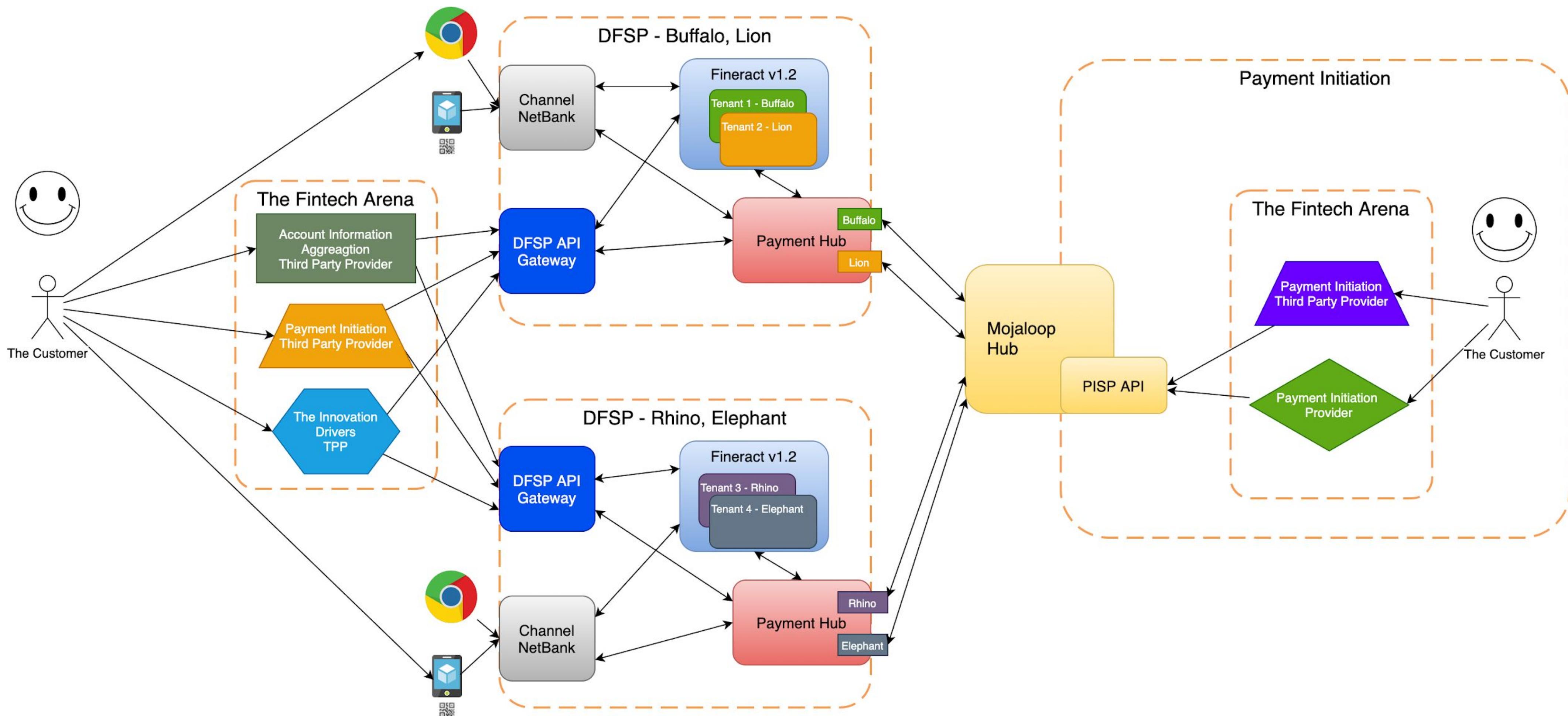
Mifos Lab - Full featured lab environment



Internals of a DFSP



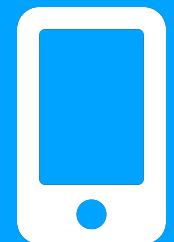
Supporting the PISP APIs



- Support the new APIs, so PISPs could execute the transactions

OpenG2P Reuses & Augments Existing Systems

Building blocks approach means programs deploy only components addressing gaps without discarding what works or start from scratch!



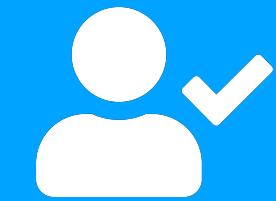
Mobile Tools:

Mobile-based solutions to enrolling, complaint handling, & beneficiary management in resource-challenged rural & perimeters



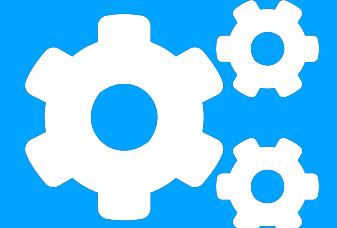
Deduplication Engine:

Extensible entity resolution & biometric framework for deduplicating & finding/matching beneficiaries usually lacking unique identities



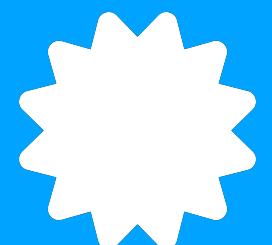
Verification Service:

Abstraction layer and tools for connecting to identity sources, e.g. civil registry, & verifying beneficiary identity against



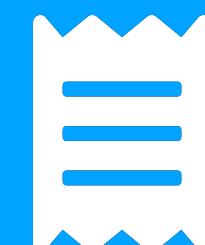
Disbursement Engine:

Abstraction layer and tooling for integrating with the financial system through existing payment rails and payment initiators



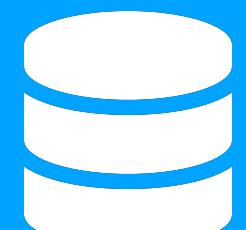
E-Voucher Engine:

Solution for serving beneficiaries outside the reach of the formal financial sector or running conditional cash transfers



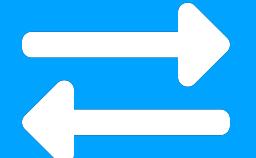
Proof of Receipt Service:

Solutions to irrefutable proof of receipt, asserting recipient's uniqueness, & running "non-preassembled list" transfers



OpenG2P ERP

A ERP, built on the Odoo ERP, for managing programs, enrolment, beneficiary data, disbursements, complaints, and more



Discovery Specification:

Open specification for information sharing among independent programs serving similar demographic



Eligibility Rules Engine:

Being iterated off continued Inputs from Mifos, DIAL, the Government of Sierra Leone, & other thought leaders

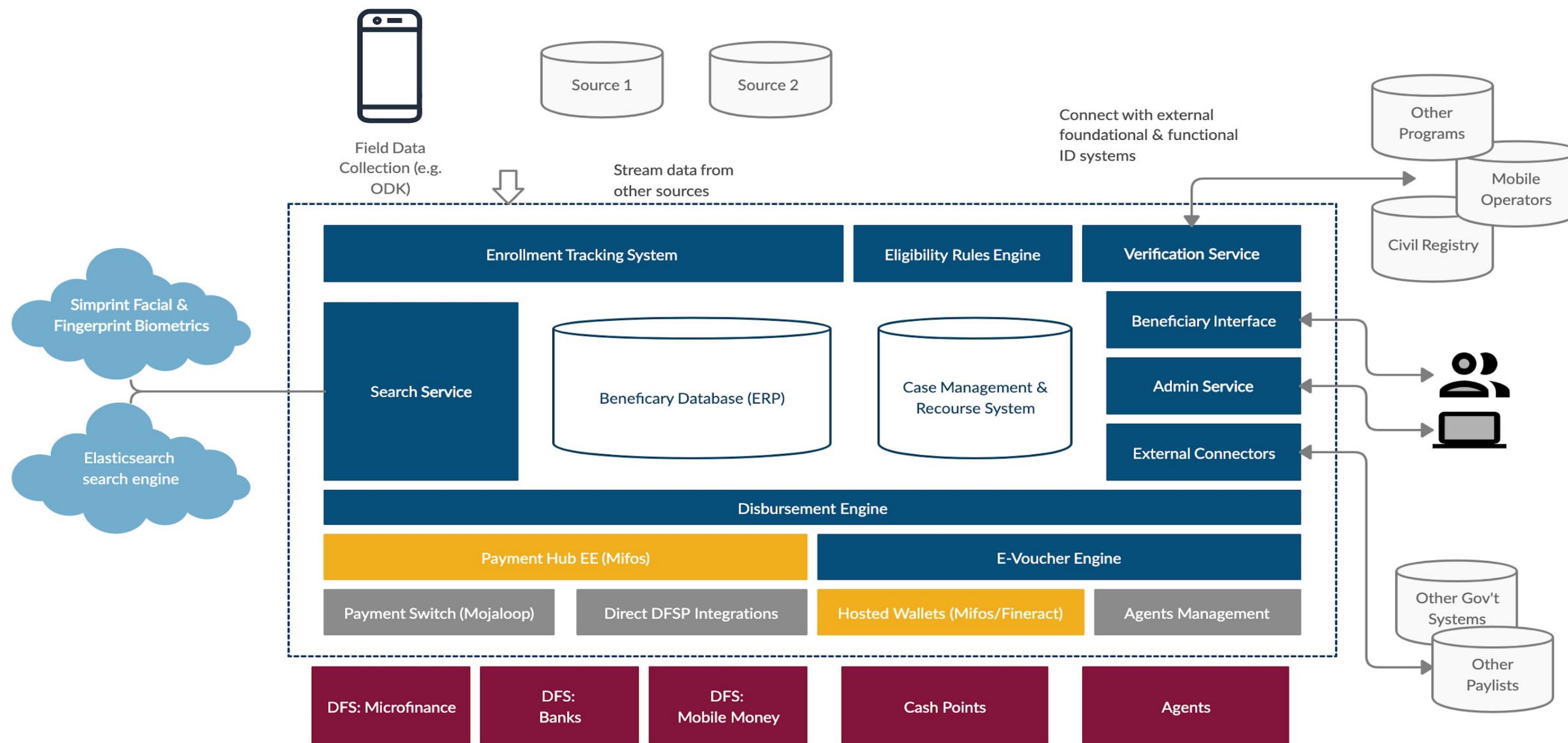
Our Guiding Ideals

It's a work in progress, but these principles guide us

- **Identification** - implement solutions consistent with ID4D principles but provide options for functional identification as long as protective of consumer privacy, user rights and provide sufficient functionality for deduplication and verifying people as unique recipients
- **Open Source First** - always include an open source version of the components.
- **Agnostic** - provide the “glue” between different core components and remain agnostic and “vendor neutral”.
- **Sustainability** - be low cost, open source for reference, and additive to existing efforts.
- **Gender Intentional** - promote women’s financial empowerment in the development of the solutions and in execution.
- **Community Engagement** - enable a toolkit or lab approach such that others can evaluate the solutions and to encourage agile and iterative software development, and take on feedback and ideas
- **Prioritize Urgent Needs** - design with the urgency of this moment for both COVID-19 physical distancing and economic crisis in many countries,

OpenG2P High Level Architecture

Each component is easily switched out

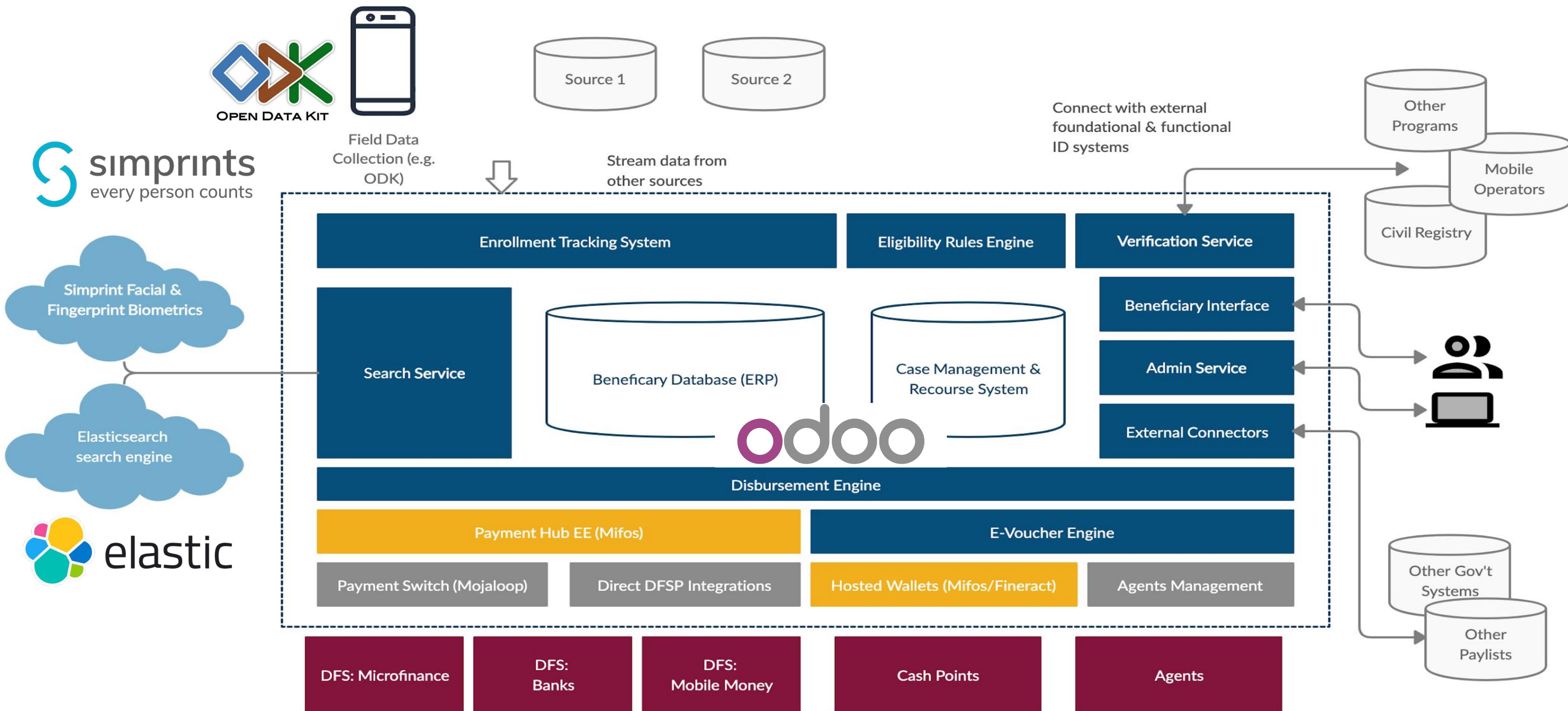


OpenG2P High Level Architecture

Deployment in Sierra Leone

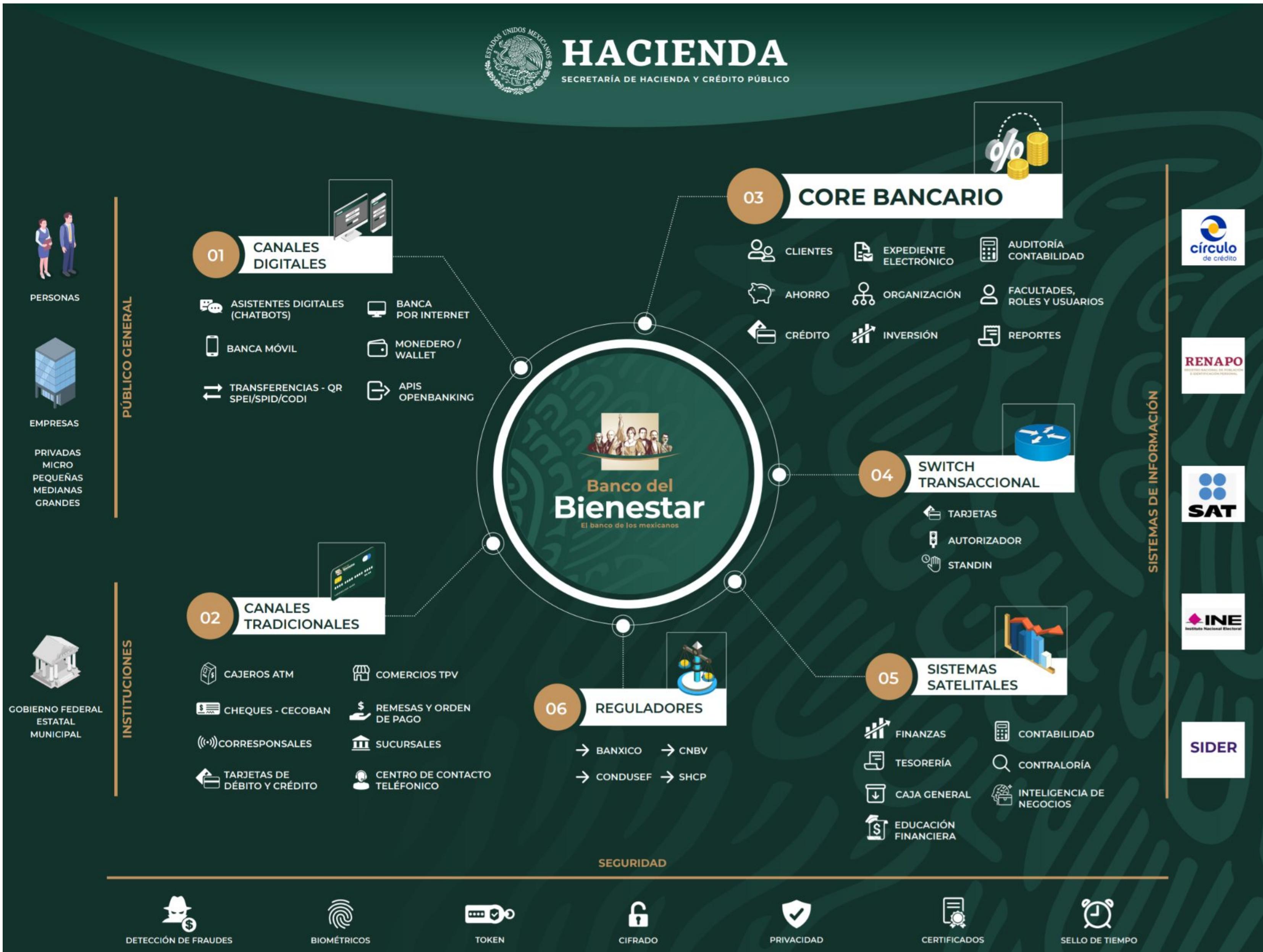


Government
of Sierra Leone



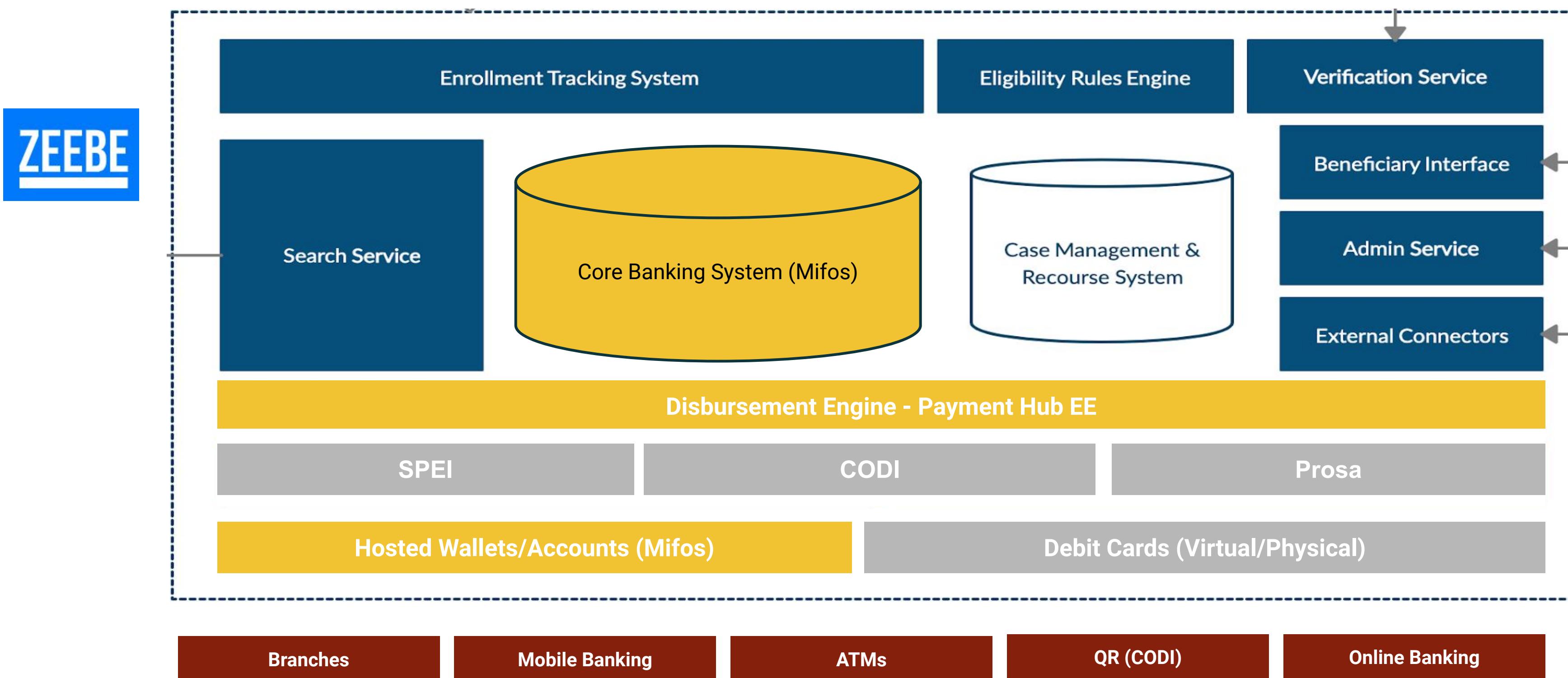
OpenG2P High Level Architecture

Deployment in Mexico



OpenG2P High Level Architecture

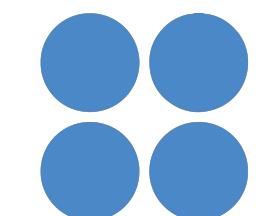
Deployment in Mexico



RENAPO

REGISTRO NACIONAL DE POBLACIÓN
E IDENTIFICACIÓN PERSONAL

INE
Instituto Nacional Electoral

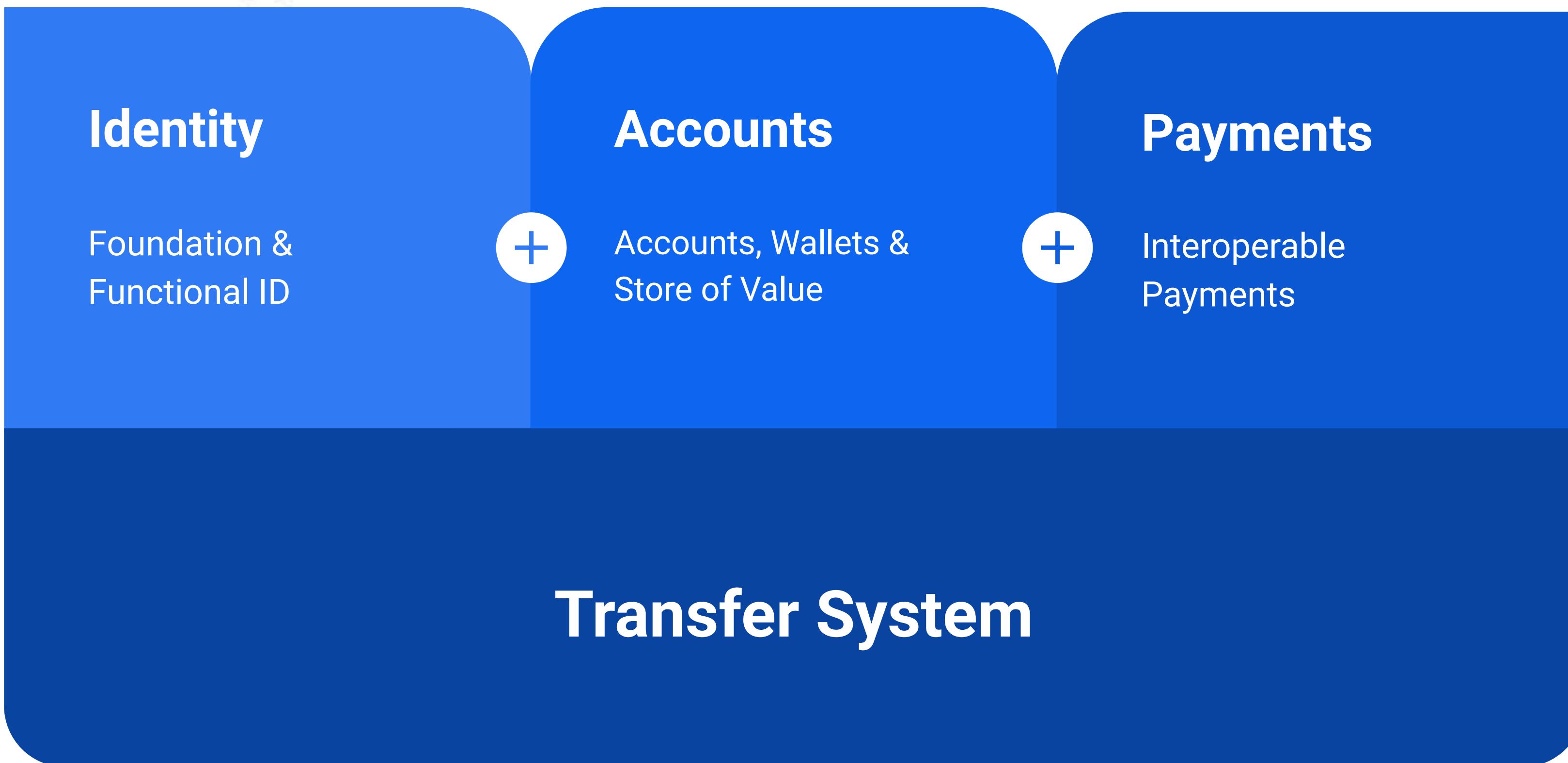


SAT

ESTADOS UNIDOS MEXICANOS
FGR
FISCALÍA GENERAL
DE LA REPÚBLICA



mojaloop



Documentation

Business Overview

- Vision
- Benefits
- Target Users

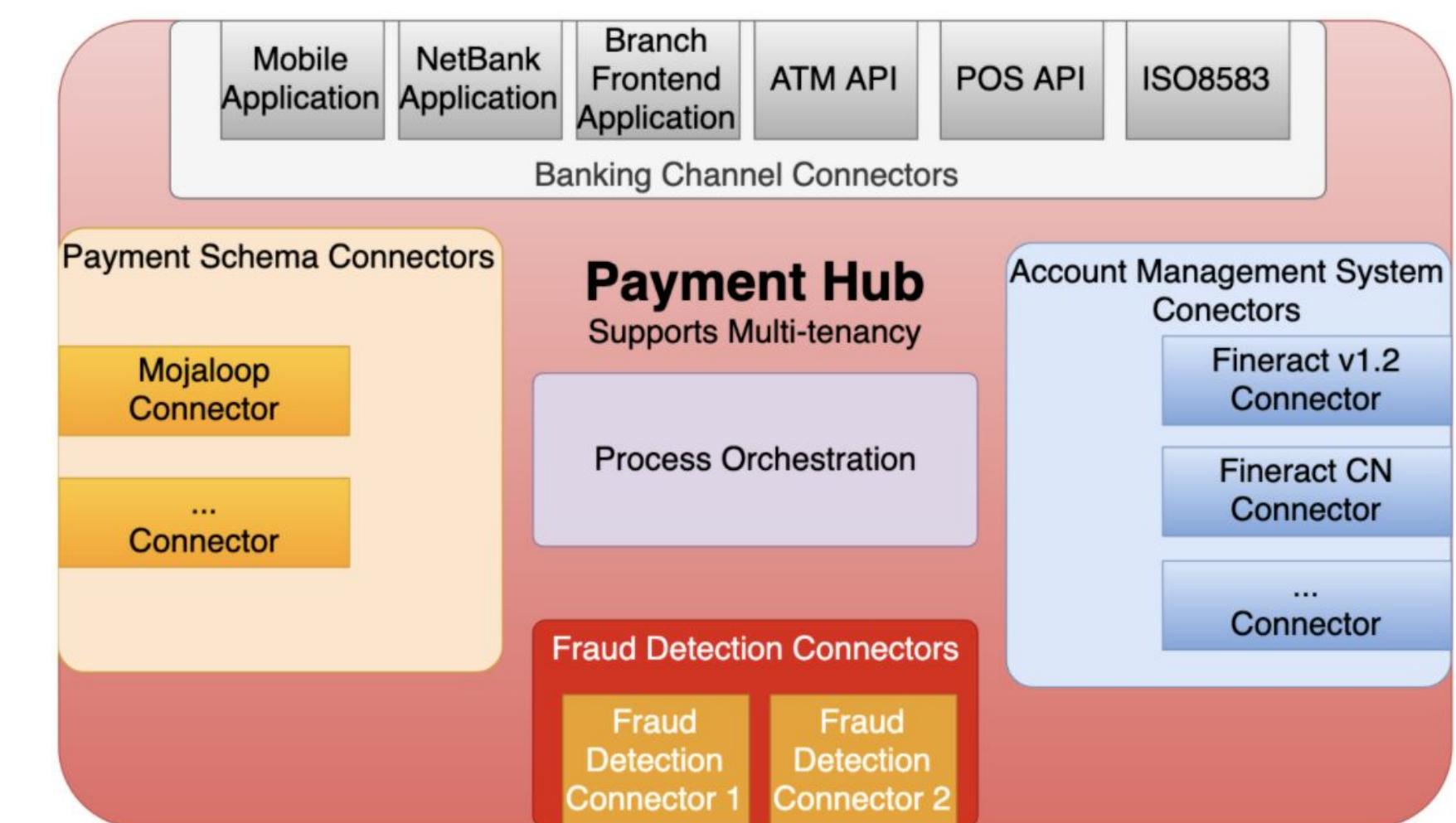
Technical Overview

- Deployment models
- Lab Environment
- Source code
- Installation instructions

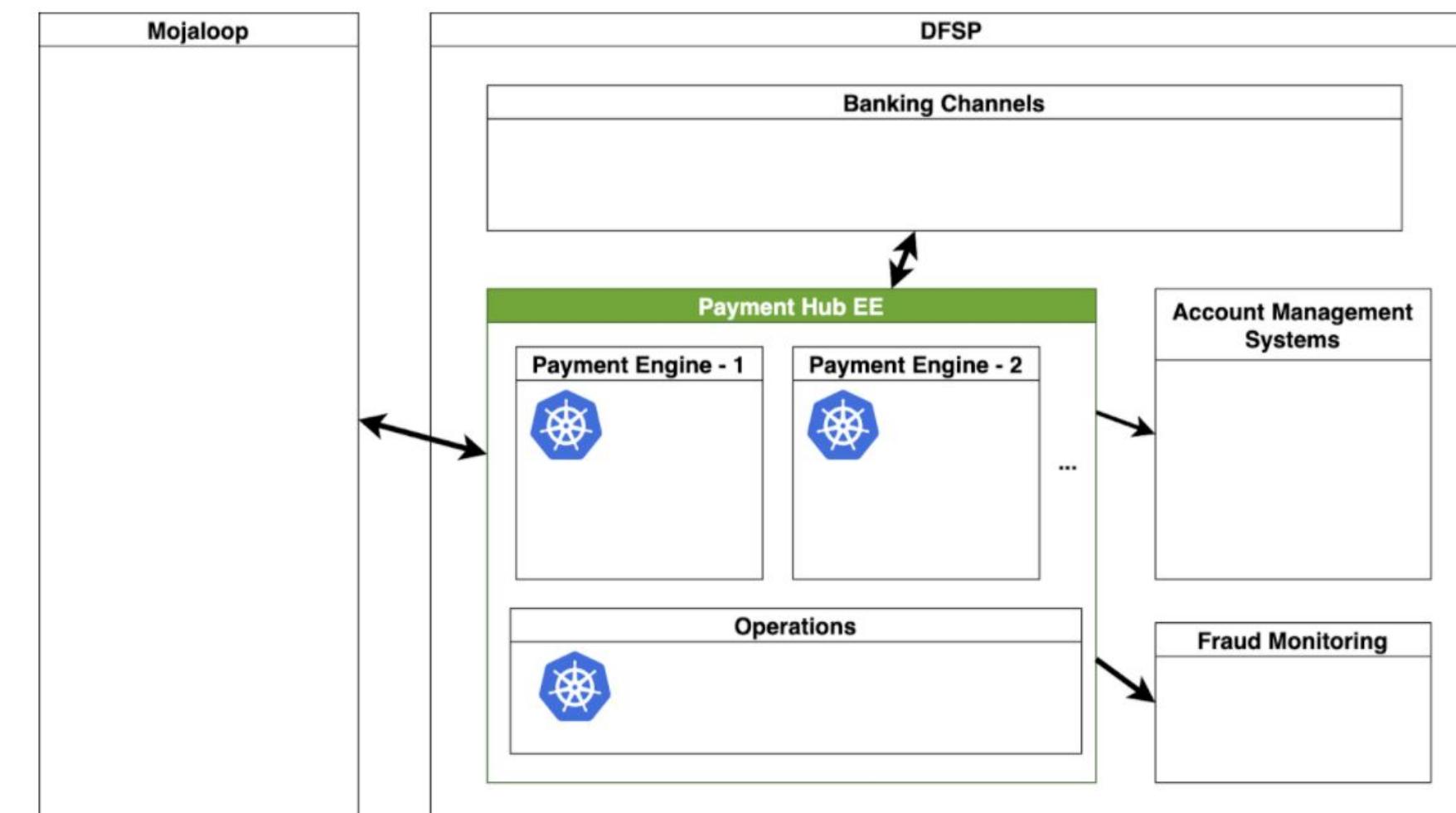
mojaloop
foundation

The screenshot shows a digital documentation interface. On the left is a sidebar with icons for Edits (3), Activities, Share, Design, Teams, Integrations, Analytics, and Advanced. The main content area has a header "LionFintech FPP demo application" with a "How to" section and a "..." menu. Below this is a "PAYOUT HUB EE" section with "Business Overview" expanded, showing "Vision", "Benefits of Payment Hub EE", "Target Users of Payment Hub EE", "Using Payment Hub EE", and "Use cases". A "Technical Overview" section is also present. At the bottom are "New" and "Files" buttons.

Design



Logical Model of the Payment Hub



The role of the Payment Hub EE at a DFSP

Roadmap for Payment Hub EE

Add support for bulk payments

Continue performance testing and tuning

Provide a platform for Cross Network Payments

- . ISO20022 (Payments Clearing and Settlement - pacs.008, pacs.002) realtime payment network integration

Participate in the PISP initiated transactions

Integrate with the SDK with the availability of the asynchronous APIs

Providing stand-in capability for Tier 1 and 2 institutions

Bulk Payment Support

- . Preprocessing bulk payments
 - Lookup Payee's DFSP ID for the transactions to determine target DFSP
 - Splitting the incoming bulk into smaller batches per target DFSP
 - On-us transactions can be handled differently
 - Manage communication with Mojaloop for the batches
 - Aggregate incoming results to provide response to the bank's channel
 - In case transferring from a single account (pension, aid), booking can be individual, aggregated by target DFSP or single grand total