mojaloop

PISP

PI 13 - January 2021

Lewis Daly, Paweł Marzec

Overview

1. Background
2. PI 12 Update
3. Design Updates
4. Roadmap

Background

Background

According to PSD2: (PSD2 = Payment Services Directive 2 in Europe which defines this role explicitly)

PISP - Payment Initiation Service Provider

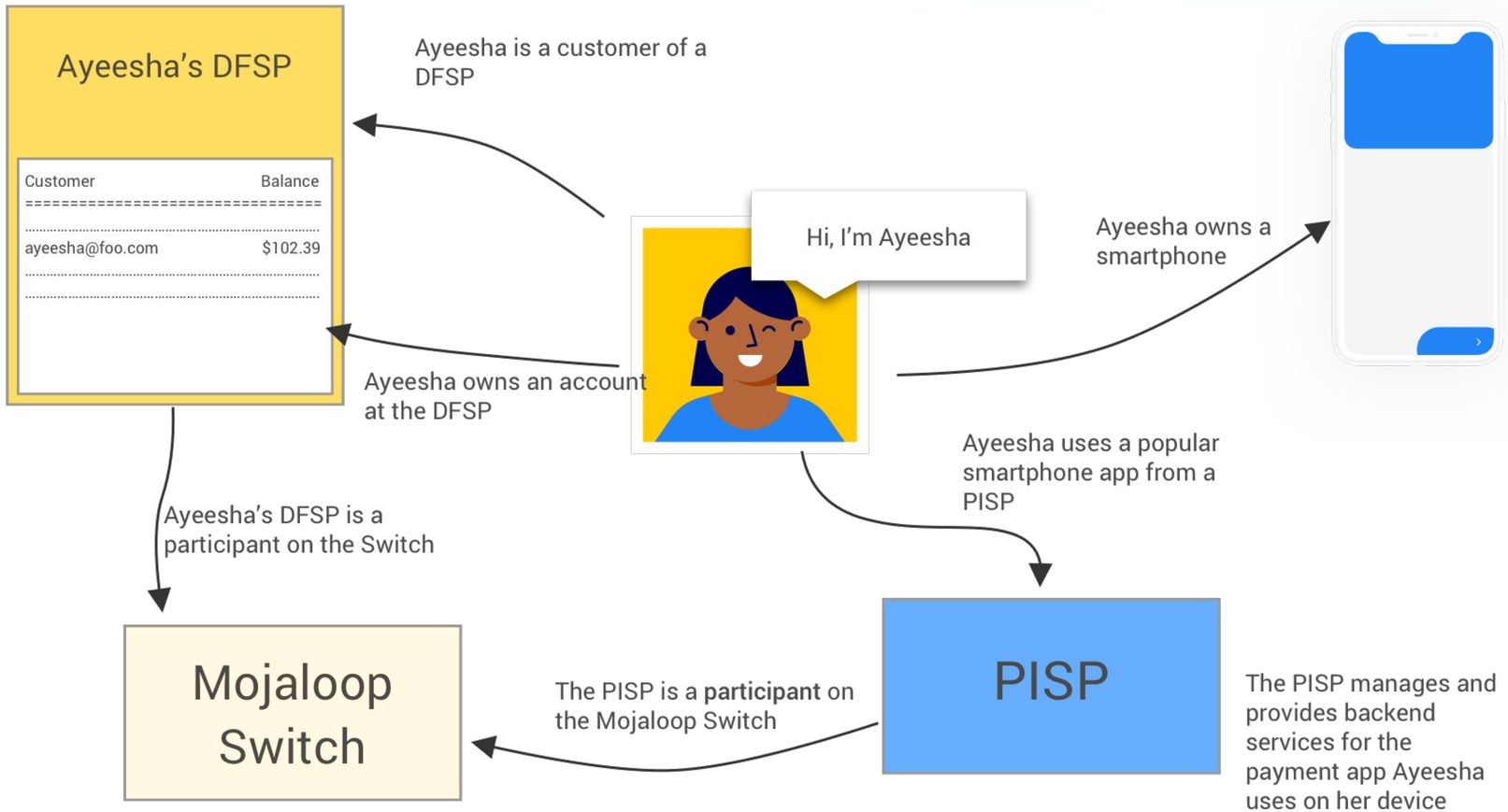
PISPs initiate payments from a user's account (on behalf of the user) to a payee account.

In Mojaloop:

A PISP is a new **participant** role, which:
- Has no liquidity (or settlement) requirements
- Initiates transactions on behalf of users at their DFSP

See the PI9 Presentation for more info.

Background



Original Source: [Mojaloop: 3rd Party Payment Initiation Proposal for Credit Transactions]

It’s been quite a year…

- Jan 2020
  - Initial PISP Proposal
  - Design discussions
- April 2020
  - Kick off development work
  - New Mojaloop Thirdparty API
  - Setting up dev + test structures
- July 2020
  - Focus on:
    - Transfers Flow
    - Auth Service
  - End to end Transfers Demo
  - Linking implementation
- October 2020
  - Error Handling
  - Producing API Draft for SIG
  -

PI 12 Update

PI12 Update

Things were a little quiet for the first half of the PI…

PI12 Update

… but now they're really getting going!

What have we done so far?

- Design and Implement Error Handling for Linking + Transfer
- Add a new CONSENT oracle type
- Solidify SDK + Thirdparty Scheme Adapters for easy integration by DFSPs + PISPs
- Design an out-of-loop OTP and WEB API for testing and demonstration purposes

Third-party API SIG

- We started the Third-party API SIG
  - Fortnightly call on Wednesdays

- API Draft v0.1 open for questions and comments

- Currently working through Sequence Diagrams

Design Updates

Transaction Callback Problem

How do we get the final transaction status back to the PISP after a PUT /transfers?

Resolved: DFSP will send the final transfer status

Why?

- The Sending DFSP is the authority on the transfer
- We don’t want any unnecessary state to be stored in the switch

OPEN-API snippets

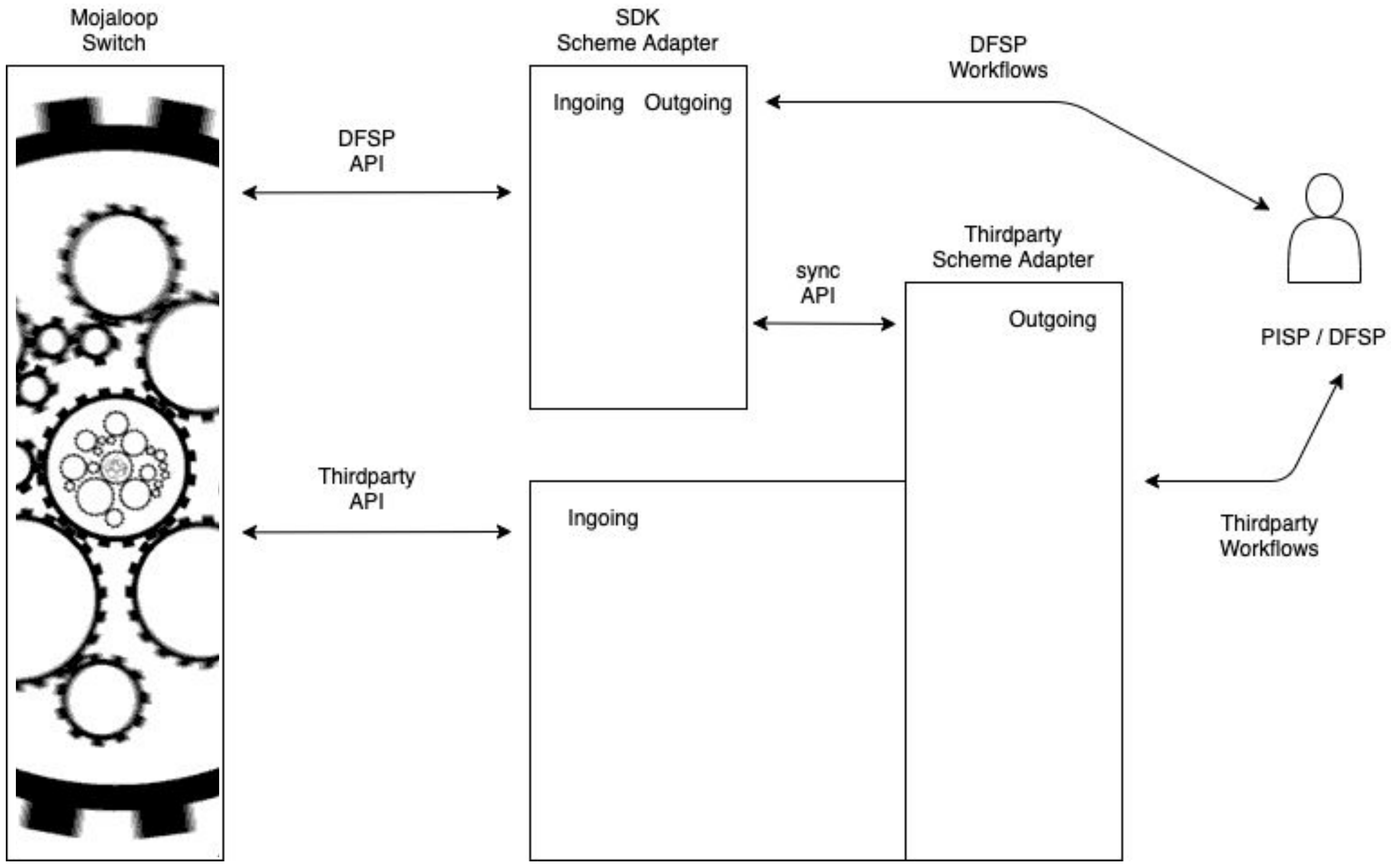
Reusable yaml definition code snippets & autogenerated Typescript interfaces

1. it is the set of basic Mojaloop Data Transfer Object Interfaces - YAML & Typescript
2. one source of truth - common type system
3. compact template definitions files for microservices
4. tooling for Typescript interfaces generation
5. Swagger UI online browser: https://docs.mojaloop.io/api-snippets/

Documentation: https://github.com/mojaloop/api-snippets

Scheme Adapters Tandem

Adapters tandem: two scheme-adapters working together to simplify PISP & DFSP's integration with Mojaloop



- **Ingoing** service accepts notification calls from Mojaloop
- **Outgoing** service accepts synchronous requests from DFSP/PISP
- **Workflow** implements a complex sequence of interactions (async calls & notifications) required by Mojaloop protocols: request to pay, bulk operation etc...
- **DFSP API** - set of endpoints dedicated to DFSP only features
- **Third-party API** - set of endpoints dedicated to PISP only features
- **Separation of concerns**: DFSP & PISP domains

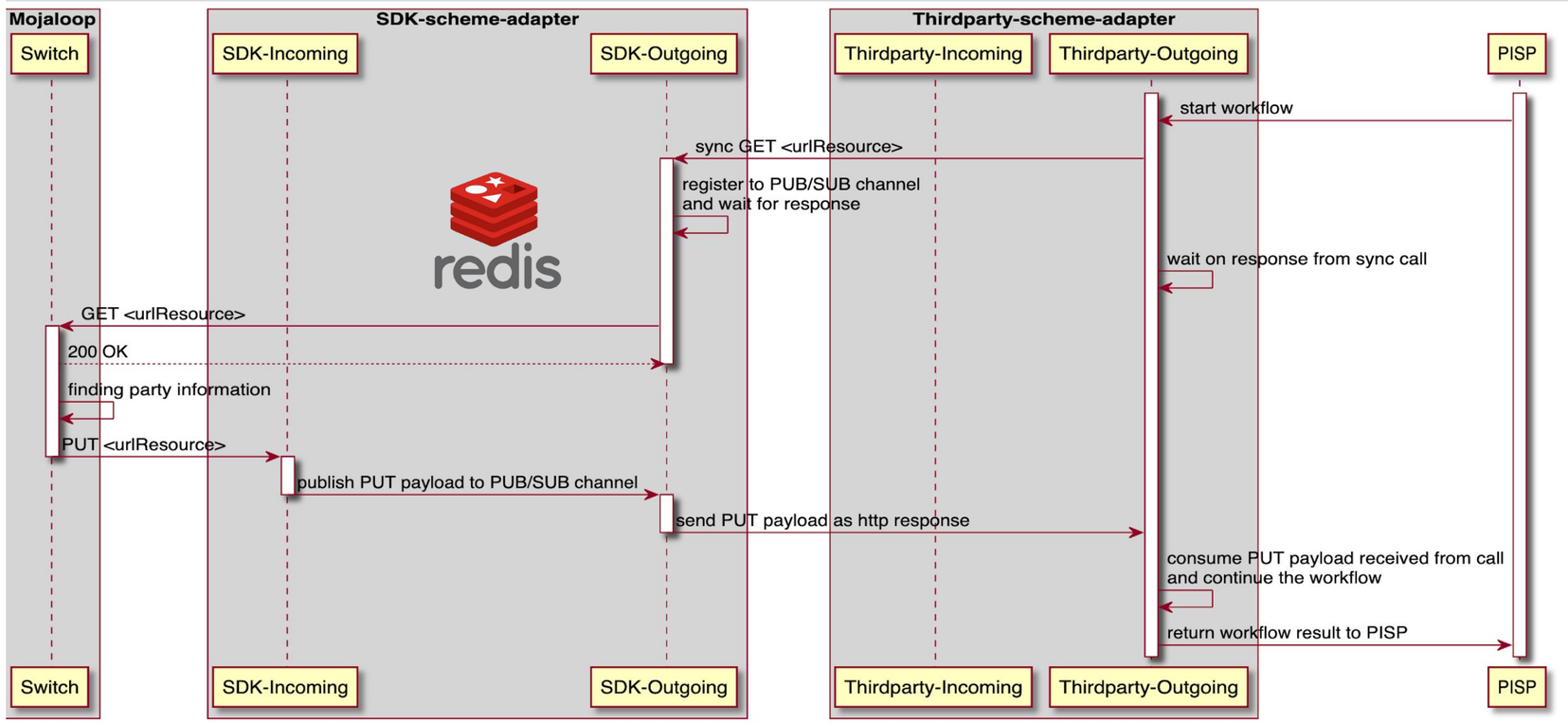
animation via giphy.com
gears within gears
by Charlie Deck:
https://bigblueboo.tumblr.com/

Documentation: https://github.com/mojaloop/pisp/tree/master/docs/scheme-adapters

Scheme Adapters Tandem

Translation of asynchronous to synchronous API using Observer (PUB/SUB) Pattern

- notification API endpoints collision: **GET /parties, POST /quotes**
- **sync API** - SDK-scheme adapter works as a server for Thirdparty-scheme-adapter



- **Incoming** service accepts notification calls from Mojaloop
- **Outgoing** service accepts synchronous requests from DFSP/PISP
- **PUB/SUB message flow** is realised by dedicated to request (UUID/query param/etc), one time created, channel @ RedisDB

Documentation: https://github.com/mojaloop/pisp/tree/master/docs/scheme-adapters

Outputs

Third-party API Draft v0.1

API Draft

- Ready for your perusal and comments

- In open discussion at Thirdparty API SIG

https://github.com/mojaloop/mojaloop-specification/issues/70

1 References

The following references are used in this specification:

Reference	Description	Version	URL
Ref. 1	Open API for FSP Interoperability	1.1	https://github.com/mojaloop/mojaloop-specification/blob/master/documents/v1.1-document-set/API%20Definition_v1.1.pdf

2 PISP API

This section describes the content of the API which will be used by PISPs.

The content of the API falls into two sections. The first section manages the process for linking customer accounts and providing a general permission mechanism for PISPs to perform operations on those accounts. The second section manages the transfer of funds at the instigation of a PISP.

The content of the first section consists of the following operations:

- The PISP requests association with a customer account on behalf of the customer.
- The owner of the customer's account satisfies itself that association really was requested by their customer, and the customer has a chance to confirm or modify directly with the account owning DFSP the types of access and the accounts for which the PISP will have permission. The DFSP then notifies the PISP that the customer has authorised access, and provides a token which the PISP can use to continue the process. This part of the process is performed via direct communication between the PISP application and the DFSP, and does not use the API.
- The DFSP requests confirmation from the PISP that the DFSP's customer has confirmed with the PISP that they authorise the PISP to perform operations on their account. Confirmation requires the PISP to provide two pieces of information:
    - The bearer token that the DFSP sent the PISP as confirmation of the successful completion of the out-of-band customer authorisation described in the previous step.
    - A FIDO challenge, signed using a public key which the DFSP (or the switch) generates and sends to

Roadmap

Roadmap

- Implement non-happy path scenarios for Linking + Transfers

- End to End Demos

- Merge back upstream to Mojaloop Core
  - New services: auth-service, thirdparty-api-adapter, consents oracle...

- Publish v1.0 of Thirdparty API

- Documentation, documentation, documentation

Questions? Comments?

Thanks!