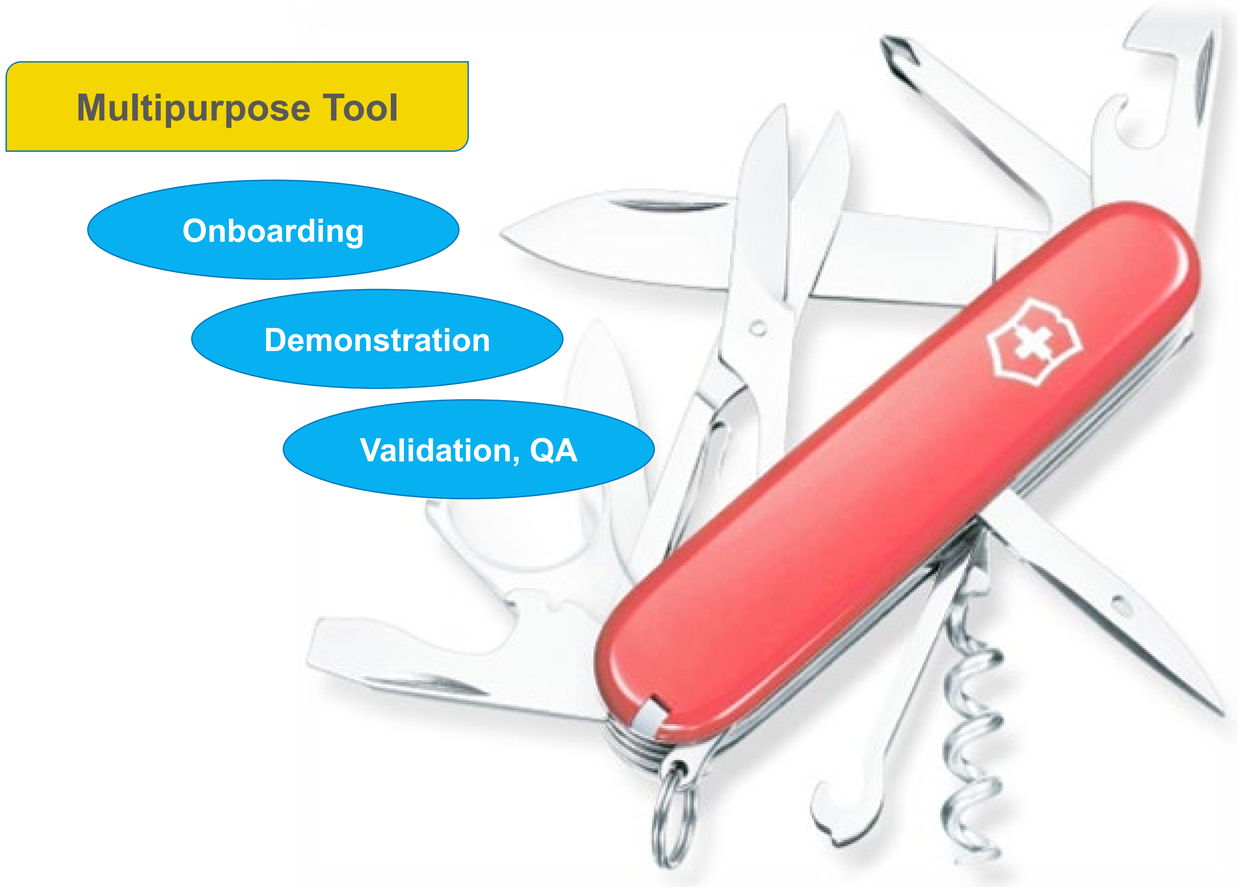
Mojaloop Testing Toolkit (TTK) PI-16

Onboarding, Validation & QA, Demonstration of Mojaloop implementations using the TTK

Agenda

1. TTK updates and overview
2. TTK MTP update
3. TTK Enhancements in PI-15
4. TTK Demos
5. Payment Manager & TTK
6. Payment Manager updates, Roadmap
7. TTK Roadmap



PI 15 Goal: Mojaloop Testing Toolkit (TTK)

Goal	**Publish Mojaloop Training Program course (MTP TTK-101); Support adopter requests and QA, IAC automation using TTK**	
Key Epics Objectives	1. Publish a Mojaloop Training Program course for the Testing Toolkit (MTP TTK-101) **3.** Improved reliability of running TTK and test coverage (>2060 assertions)	**2.** TTK is included as part of Mojaloop helm charts and testing is automated [DevOps / Core] **4.** Stabilizing TTK Hosted mode and validating with at least one Sandbox user
Not Doing now but important next & Risk / Issues	1. Adopter requests of higher priority affect current backlog 2. Risk: Funding support for TTK is reduced	
Success Defined How?	1. Mojaloop Training Program (MTP TTK-101) course for TTK is published. 2. Mojaloop TTK GP tests contain >2100 assertions with test case definition reports available based on versions	3. Mojaloop TTK adopted as the tool for validating Switch deployments by at least one implementer 4. Testing Switch (or Mojaloop) deployments using TTK is automated.

mojaloop foundation

Teams using TTK

1. Mojaloop releases & core maintenance team
2. PISP work-stream
3. ATM / POS feature team
4. Mojaloop support teams
5. Mojaloop product teams
6. Hackathons – HiPiPo, DFSLab, ISO20022
7. Visa Accelerator Program – PoC

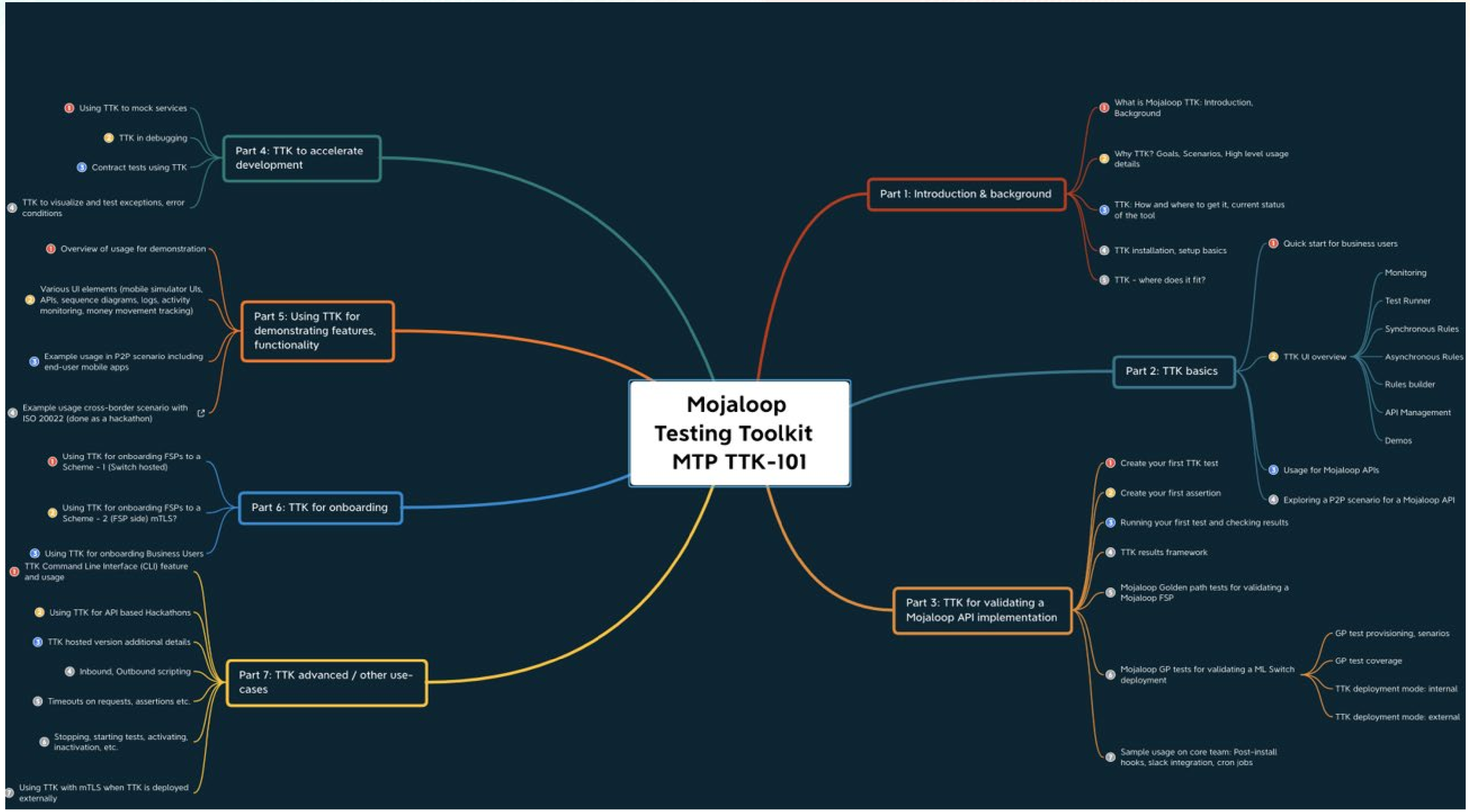
Team

1. Emerson Pereira
2. Georgi Logodazhki
3. Juan Correa
4. Kevin Leyow
5. Lewis Daly
6. Matt Bohan
7. Michael Richards
8. Miguel de Barros
9. Sam Kummary
10. Vijay Guthi
11. Yevhen Kyriukha

TTK Training

1. Prepared content on TTK for **Mojaloop Training Program (MTP)**
2. Focused on TTK usage
3. Currently in review stage
4. Collaborating with MTP team

MTP TTK-101


Mojaloop Testing Toolkit MTP TTK-101
Part 1: Introduction & background
What is Mojaloop TTK: Introduction, Background
Why TTK? Goals, Scenarios, High level usage details
TTK: How and where to get it, current status of the tool
TTK installation, setup basics
TTK - where does it fit?
Part 2: TTK basics
Quick start for business users
TTK UI overview
Monitoring
Test Runner
Synchronous Rules
Asynchronous Rules
Rules builder
API Management
Demos
Usage for Mojaloop APIs
Exploring a P2P scenario for a Mojaloop API
Part 3: TTK for validating a Mojaloop API implementation
Create your first TTK test
Create your first assertion
Running your first test and checking results
TTK results framework
Mojaloop Golden path tests for validating a Mojaloop FSP
Mojaloop GP tests for validating a ML Switch deployment
GP test provisioning, senarios
GP test coverage
TTK deployment mode: internal
TTK deployment mode: external
Sample usage on core team: Post-install hooks, slack integration, cron jobs
Part 4: TTK to accelerate development
Using TTK to mock services
TTK in debugging
Contract tests using TTK
TTK to visualize and test exceptions, error conditions
Part 5: Using TTK for demonstrating features, functionality
Overview of usage for demonstration
Various UI elements (mobile simulator UIs, APIs, sequence diagrams, logs, activity monitoring, money movement tracking)
Example usage in P2P scenario including end-user mobile apps
Example usage cross-border scenario with ISO 20022 (done as a hackathon)
Part 6: TTK for onboarding
Using TTK for onboarding FSPs to a Scheme - 1 (Switch hosted)
Using TTK for onboarding FSPs to a Scheme - 2 (FSP side) mTLS?
Using TTK for onboarding Business Users
Part 7: TTK advanced / other use-cases
TTK Command Line Interface (CLI) feature and usage
Using TTK for API based Hackathons
TTK hosted version additional details
Inbound, Outbound scripting
Timeouts on requests, assertions etc.
Stopping, starting tests, activating, inactivation, etc.
Using TTK with mTLS when TTK is deployed externally

Features, Enhancements in PI-15

1. New features
    a. Grouping **test-cases** with labels
    b. New **Payee Mobile App Simulator** to show inbound transfer
2. Improvements
    a. Combine CGS and Differed settlement collections
    b. Parameterize variables (fspids) in provisioning collection and in GP collection
    c. New **FX API** added (API in design phase, with CCB)
    d. Hosted mode
        i. Optional TLS
        ii. Token authorization
        iii. Bugfixes
    e. Improved payment manager integration
3. **Mojaloop Training Program (MTP)** course for TTK
4. Bug fixes
5. Issues closed (**16**): https://github.com/mojaloop/project/issues?q=label%3Aoss-ttk+closed%3A%3E2021-08-01+updated%3A%3C2021-10-26+

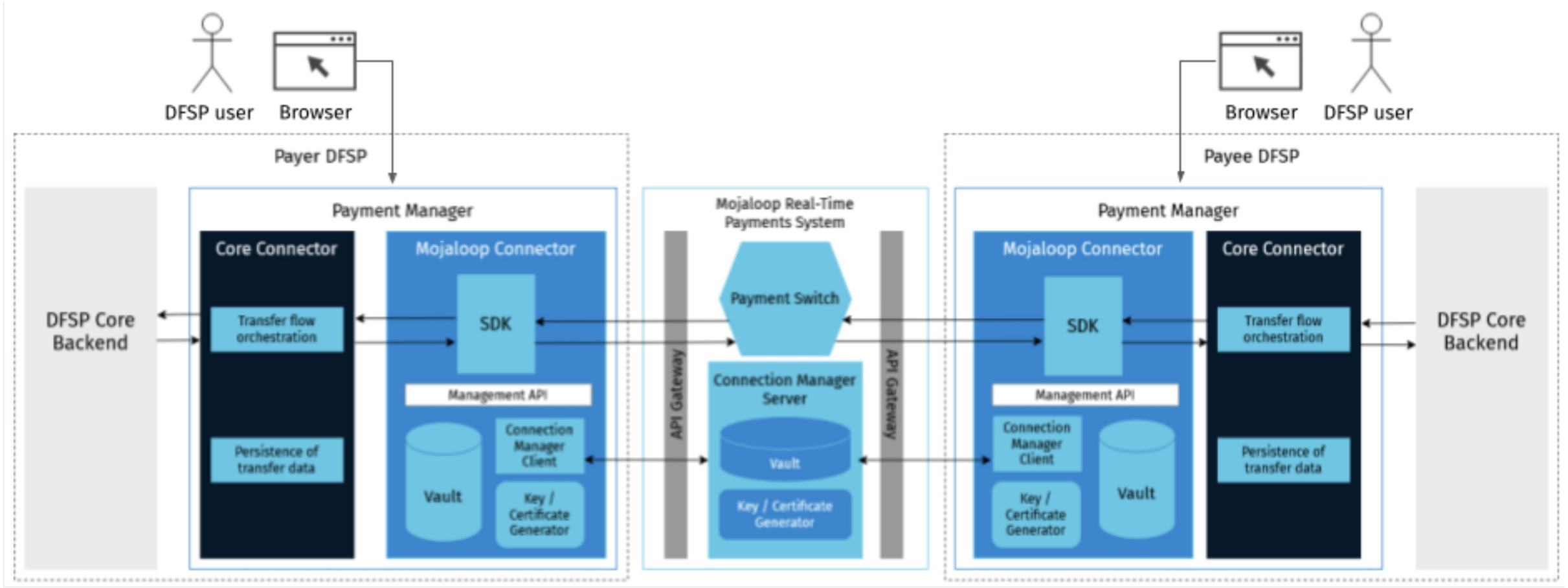
Demos

1. Hosted mode
    1. Token authorization
    2. Docker-compose without TLS
    3. Docker-compose with TLS
2. Payee Mobile Simulator
3. Grouping Testcases with labels
4. Payment Manager and TTK

Payment Manager & Testing Toolkit

By Juan Correa

TTK & On-Prem PM4ML Development

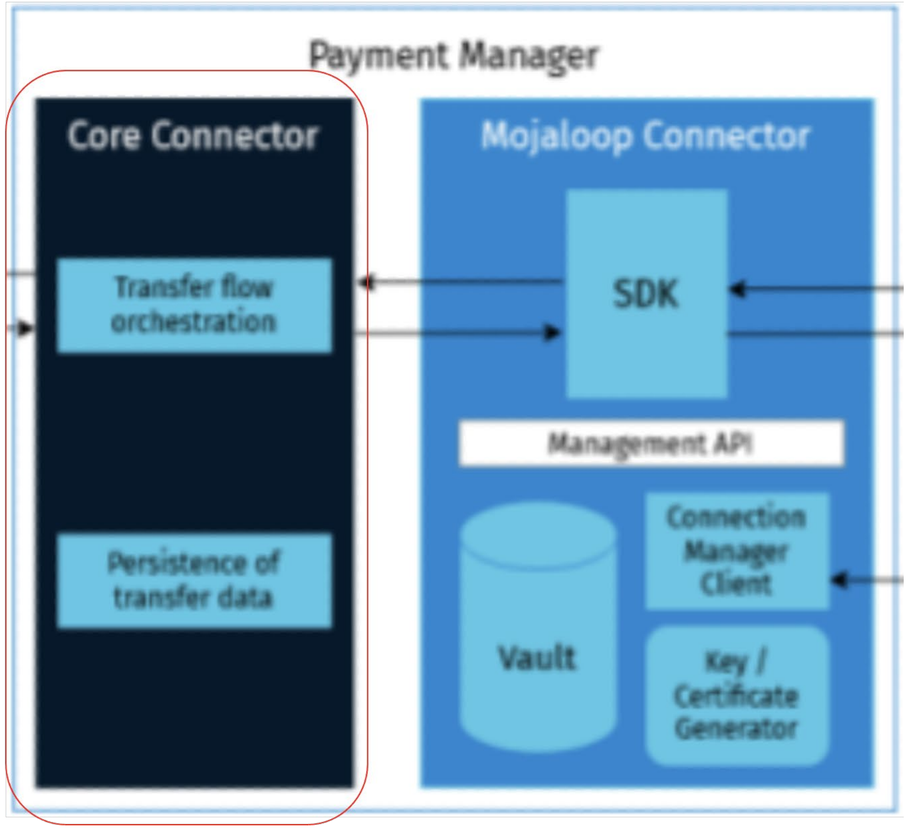


TTK & On-Prem PM4ML Development

mojaloop foundation

Mojaloop Testing Toolkit

&



TTK & On-Prem PM4ML Development

What is TTK?

Mojaloop Testing Toolkit (TTK) is an open source integration testing tool developed with the intention to facilitate testing between a DFSP and a Mojaloop Hub.

https://github.com/mojaloop/ml-testing-toolkit/blob/master/documents/User-Guide.md

How to install on-prem PM4ML for local development

https://github.com/pm4ml/on-premise-deploy

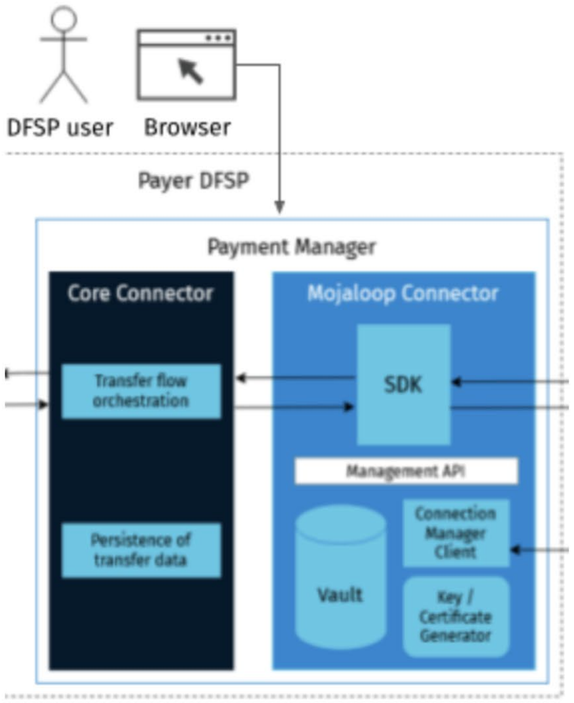
What is a PM4ML Core-Connector?

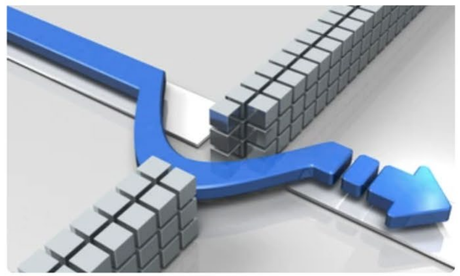
Apache Camel based component that enables integration between a DFSP and a Mojaloop Hub via PM4ML.

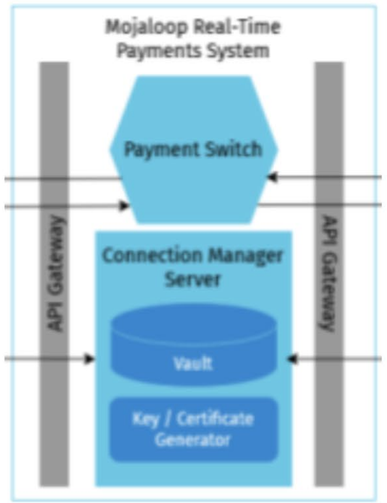
https://pm4ml.github.io/documents/payment_manager_oss/latest/core_connector_rest/index.html

TTK & On-Prem PM4ML Development Use Cases

1. Develop Payer Core-Connector
   a. In this use case the TTK simulates Mojaloop Hub
   b. Allows for local development of CC without access to a Mojaloop Lab

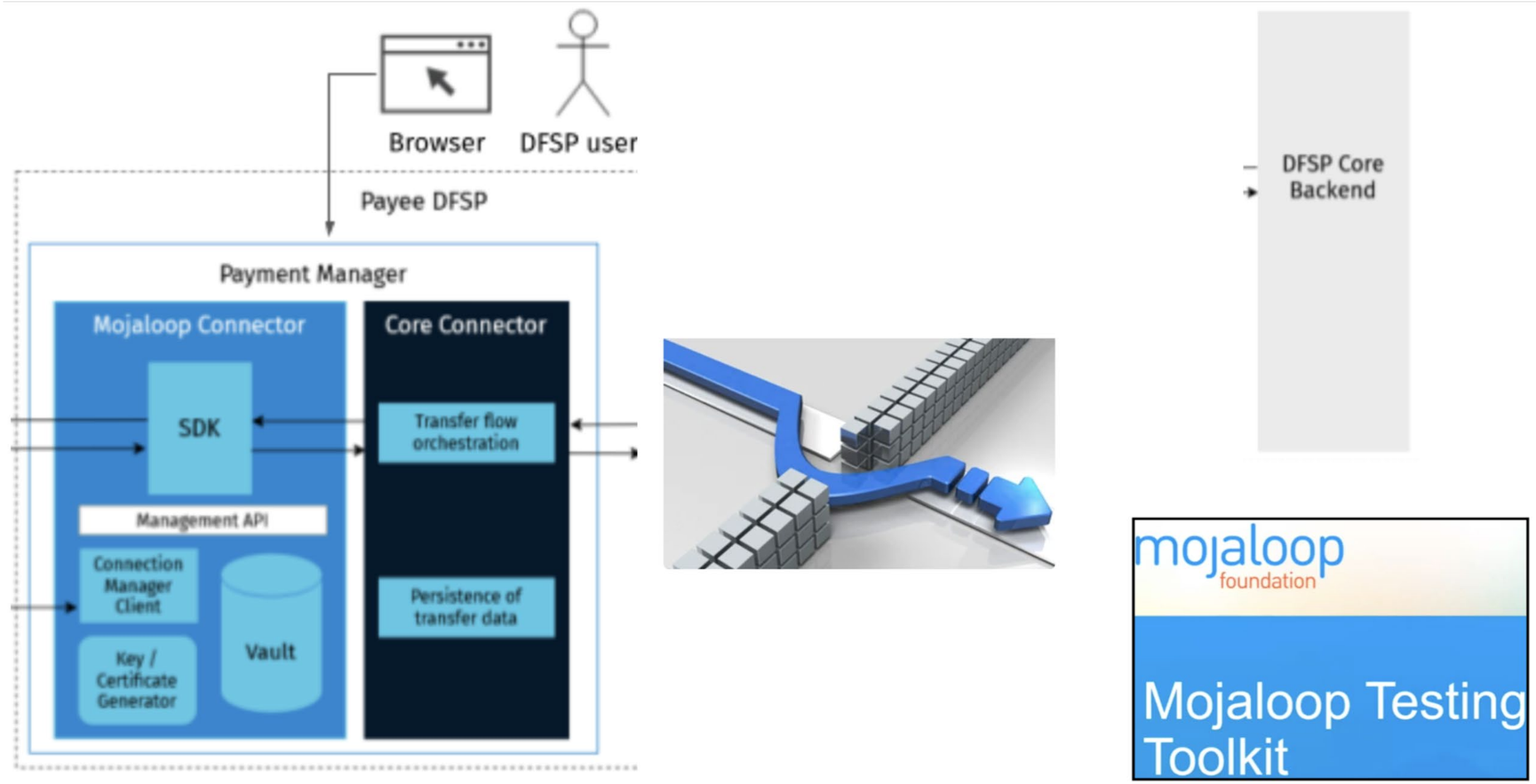






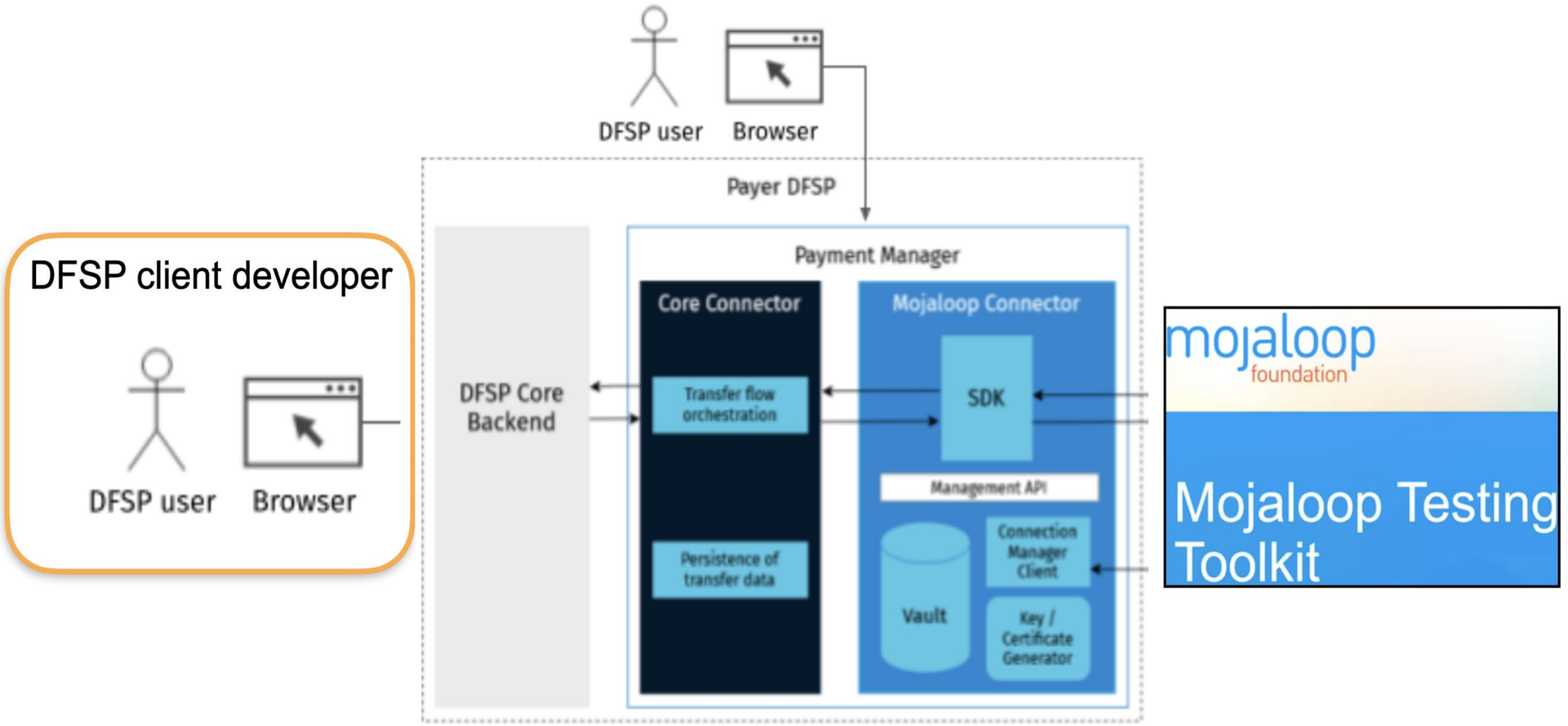
TTK & On-Prem PM4ML Development Use Cases

2. Develop Payee Core-Connector
   a. In this use case the TTK simulates Payee DFSP Core Banking System (CBS)
   b. Allows for local development of CC without access to a CBS API sandbox



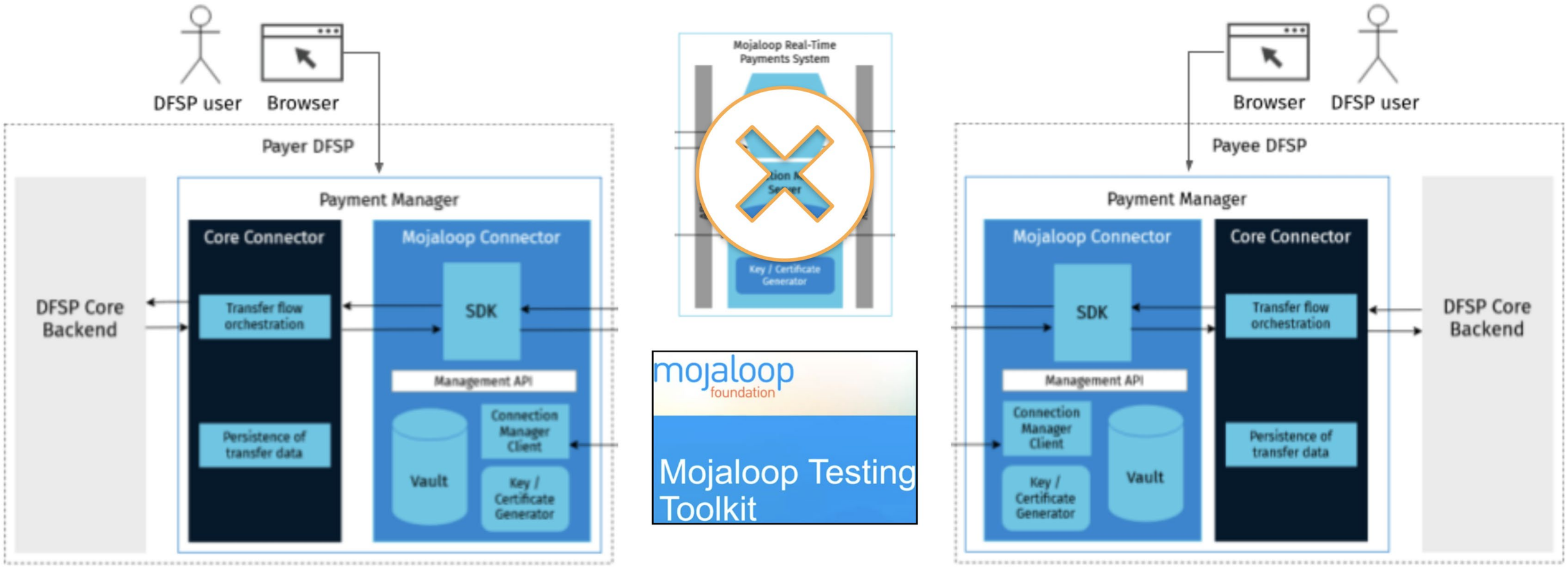
TTK & On-Prem PM4ML Development Use Cases

3. Develop the client to a Payer Core-Connector by DFSP
   a. In this use case the TTK simulates the Mojaloop Hub and Payee Core-Connector
   b. Allows DFSP to start integration with on-prem PM4ML and without need to a Mojaloop Lab



TTK & On-Prem PM4ML Development Use Cases

4. End2End Testing
   a. In this use case the TTK simulates the Mojaloop Hub
   b. Allows for previously independently developed Payer CC, Payee CC, DFSP client to be tested end2end



TTK Roadmap

1. Support implementation teams adopt and use TTK for validation, testing implementations & onboarding FSPs
2. Improve test coverage; use FSPIOP v1.1 as default in tests
3. Improvements in Test Runner
4. Request level runtime information for behavioral testing
5. Support for feature requests from current adopters
6. Mojaloop Training Program: **TTK 101** published

Thank you!

Mojaloop Testing Toolkit