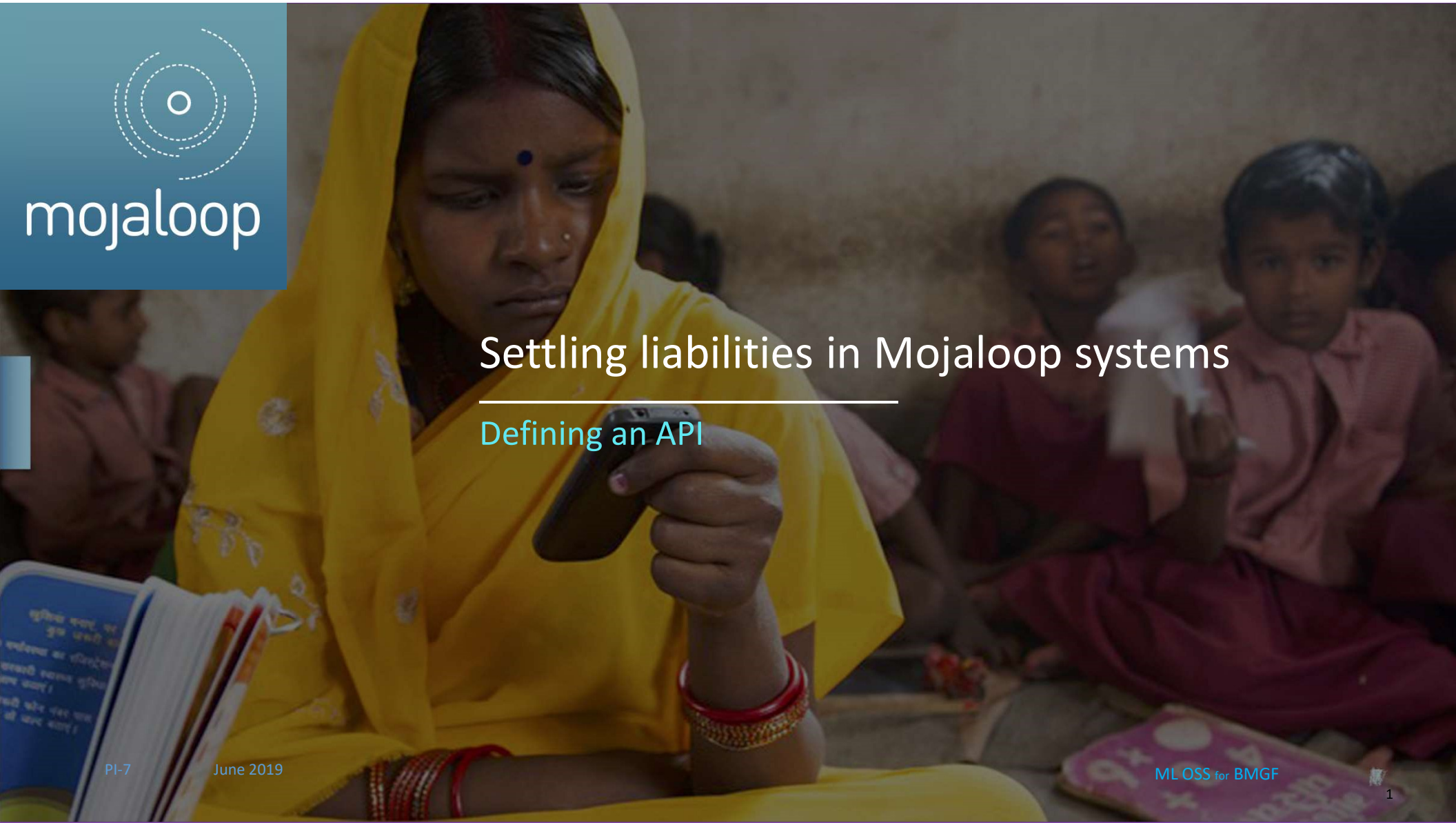


Settling liabilities in Mojaloop systems

Defining an API

Indispensable cultural motto

All things invite
To peaceful counsels and the settled state
Of order, how in safety best we may
Compose our present evils, with regard
Of what we are and where, dismissing quite
All thoughts of war. Ye have what I advise.

John Milton

Paradise Lost, II, 278-83

Objectives

1. To discuss (and, hopefully, define) the things that we need to include
2. To define the settlement models that we intend to support
3. To define the objects that we propose to work with
4. To define the actors we expect to want to use a Settlement API
5. To model the ways in which we expect each actor to use the API

Objectives

1. To discuss (and, hopefully, define) the things that we need to include.
2. To define the settlement models that we intend to support
3. To define the objects that we propose to work with
4. To define the actors we expect to use a Settlement API
5. To model the ways in which we expect each actor to use the API
6. To dip our toe into the dark waters of definition…

What sorts of things do we need to include?

Defining terms

Definition of terms: settlement models

- A settlement model is a definition within the Mojaloop system of a form of settlement.
- Settlement has the following characteristics:
  - Settlement may be immediate or deferred
  - A settlement model may…
    - Settle gross or net
    - Settle bilaterally or multilaterally
    - Require pre-funded liquidity or it may not
  - A settlement model settles a single switch account
  - A switch account records one or more ledger entry types.
    - Each ledger entry type in the scheme belongs to only one switch account.

Definition of terms: settlement accounts

1. A traditional *settlement account* holds actual funds for a DFSP in a bank account at the settlement bank.
   a. This account is debited or credited by the scheme with settlement entries which offset participants' liabilities incurred by cleared transfers.
   b. DFSPs manage their settlement accounts by transferring funds from other accounts they hold.
   c. Balances in these settlement accounts represent liability cover for the participants.
2. A settlement account may also be a pooled account.
   1. A pooled settlement account is a settlement account which holds actual funds in a bank account at the settlement bank representing liability cover for more than one DFSP.
3. Either traditional or pooled settlement accounts *may* be used with either net or gross settlement.

Definition of terms: net and gross settlements

- In a *net* settlement model, multiple individual journal entries are summed together *over a period*, and the net of those transfers is settled by participants.
- In an *gross* settlement model, journal entries are settled discretely, line by line.

Settlement types

- First choice: should my settlement be *immediate* or *deferred*?
- If deferred, the API supports:
  - *Deferred net settlement*, either
    - Multilateral or
    - Bilateral (on roadmap)
  - *Deferred gross bilateral settlement* (traditional RTGS, on roadmap)
- If immediate, the API supports:
  - *Immediate gross bilateral settlement*, using a pre-funded pooled account

Definition of terms: multilateral and bilateral settlements

- In *multilateral* settlement models, each participant settles *with the scheme.*
- In *bilateral* settlement models, each participant settles with every other participant.

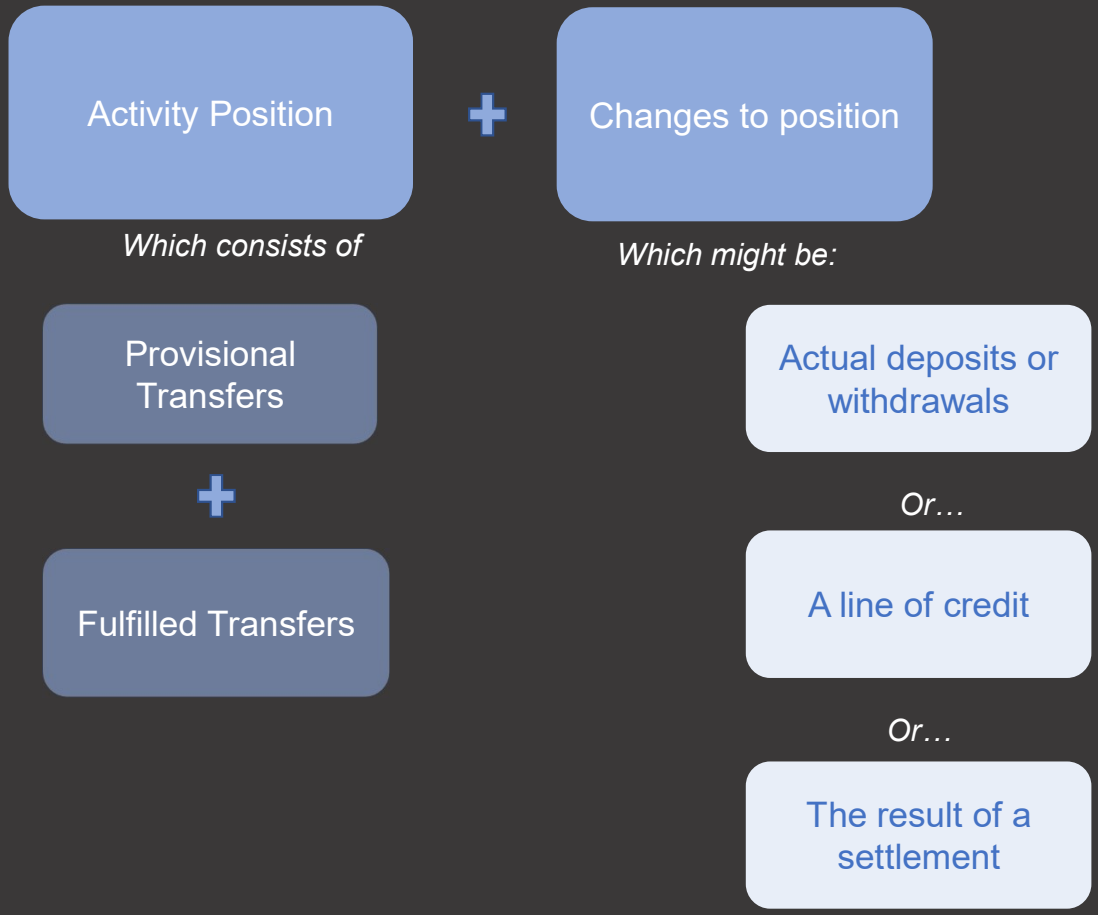
Definition of terms: ledger entry types

1. A *ledger entry type* defines a type of ledger entry which a scheme intends to record and settle.
    a. Ledger entry types belong to a scheme account
    b. In addition to storing actual transfer amounts, ledger entry types can be used to implement scheme-specific charges such as interchange fees.
    c. A ledger entry type may only belong to one scheme account, but a scheme account may contain more than one ledger entry type.

Definition of terms: participant position

1. A *participant position* is a net position of all activity
2. A position applies to a switch account for a scheme participant in a specific currency.
3. A position represents:
    1. The activity in an account which is a consequence of transfers flowing through the system
    2. Adjustments to the account as directed by the scheme
4. The position is calculated as:
    1. The sum of reservations made for prepared but unfulfilled transfers where the participant is the payer.
    2. The sum of fulfilled transfers where the participant is the payee.
    3. Any changes to the account position directed by the scheme.

Components of a participant's position



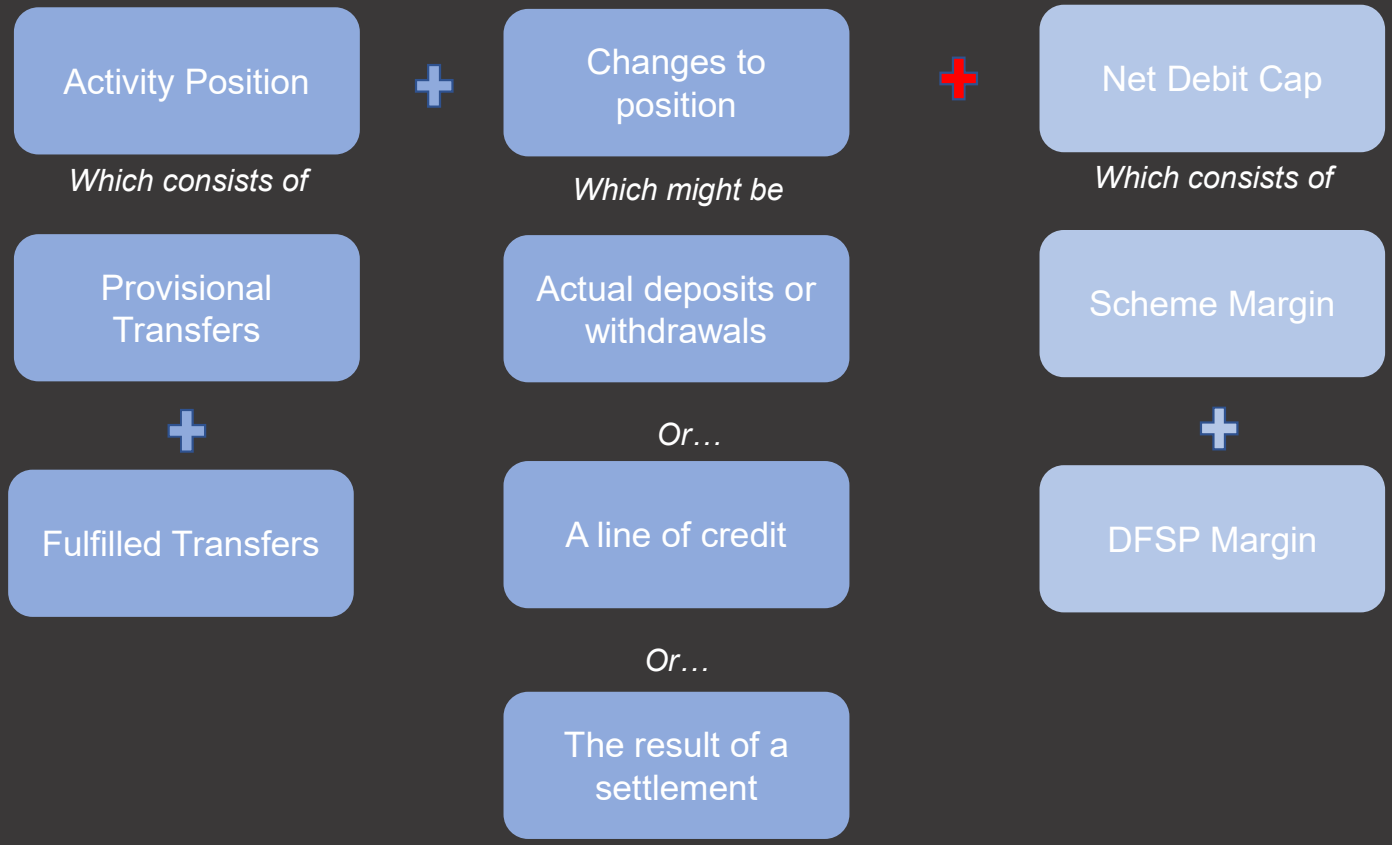
Definition of terms: Net Debit Cap

1. In principle, a participant is not allowed to trade in the system if their position would become negative.
    1. Which means, in this context: *if a proposed transfer would cause the net of their transfer activities in the account to exceed the sum of credits and debits recorded in their switch account.*
2. So the principle is: a participant must always be able to cover their liabilities. We call this a *liquidity cover requirement*
3. Seems clear enough…

Definition of terms: Net Debit Cap

- But in fact the scheme needs to be able to support some variations to the basic liquidity cover requirement:
  1. The *scheme* may want to increase the amount of liquidity cover required.
     1. The scheme may define these variations *either* for individual participants *or* for the scheme as a whole
  2. A *participant* may wish to increase their own liquidity requirement.
- We therefore define a Net Debit Cap (NDC) as follows:
  1. The amount by which the scheme is prepared to permit a variation in the liquidity cover requirement; *plus*
  2. The amount by which a participant is prepared to reduce their own liquidity cover requirement. A participant is not permitted to increase their liquidity cover requirement
- This composite is added to the base liquidity cover available to a participant when the scheme decides to accept a proposed transfer where the participant is the debtor.

Available liquidity cover



If the sum of these values is greater than the amount you wish to transfer, then you may transact

The two sides of the transfer approval process

On the left side, the participant's *net position*, comprising:

- Unfulfilled transfers where the party is the debtor
- Fulfilled transfers
- Deposits and withdrawals recorded by the scheme

On the right side, *liability variations*, comprising:

- The scheme's additional requirements for liability cover over and above the participant's actual position
- Possibly reduced by the party's own reservation.

Definition of terms: settlements

1. A *settlement* represents a call to participants to settle their current obligations to each other.
2. For net settlements, it causes the scheme to calculate the amount that each participant owes to, or is owed by, the scheme (in a multilateral settlement model) or another participant (in a bilateral settlement model) at the moment the settlement is requested.

Definition of terms: settlements

1. The settlement itself (that is, the actual liquidation of obligations) takes place outside the switch.
2. The switch only knows about the progress of the settlement because scheme administrators or automated systems inform it in one of two ways:
    1. They update a participant's status for a specific settlement.
        - This may result in a change in the status for the settlement as a whole: for instance, if all of the participants have now settled and therefore the settlement as a whole is now complete.
    2. They record changes to a participant's position which may be consequent on that participant settling their obligations in a settlement

Settlement models in the market today

Net Settlement Systems

- Almost universally used for retail open-loop systems
- Can be either multilateral (each participant receives an overall statement) or bilateral (each participant settles with every other participant)

Gross Settlement with Individual Accounts

- Universally used for wholesale open-loop systems (RTGS)
- Being used by some RTRP systems (e.g. Mexico)

Gross Settlement with Pooled Account

- Generally used for internal accounting within Mobile Money systems (continuous.)
- Being used in US banking RTP system (continuous.)
- Under consideration for Tanzania TIPS.

Some examples…

- We show 3 DFSPs.
- We assume that the base liquidity cover requirement is used.
- For each transfer:
  - We show 3 types of position: unsettled, settled and aggregate
  - Places where ledger amounts change are highlighted in green
  - Where transfers are rejected, the relevant cells in the NDC area are highlighted in red
  - For the sake of concision, we do not show the gap between acceptance of a transfer and completion of the transfer

First, a little light registration…

	Effect			Deposit/(Withdrawal)			Aggregate Position			Unsettled Position			Settled Position			NDC		
Action	**DFSP A**	**DFSP B**	**DFSP C**	**DFSP A**	**DFSP B**	**DFSP C**	**DFSP A**	**DFSP B**	**DFSP C**	**DFSP A**	**DFSP B**	**DFSP C**	**DFSP A**	**DFSP B**	**DFSP C**	**DFSP A**	**DFSP B**	**DFSP C**
DFSP A joins scheme							0	0	0				0	0	0	0	0	0
DFSP B joins scheme							0	0	0	0	0	0	0	0	0	0	0	0
DFSP C joins scheme							0	0	0	0	0	0	0	0	0	0	0	0
DFSP A deposits $1000				1000			1000	0	0	0	0	0	1000	0	0	0	0	0
DFSP B deposits $1000					1000		1000	1000	0	0	0	0	1000	1000	0	0	0	0
DFSP C deposits $1000						1000	1000	1000	1000	0	0	0	1000	1000	1000	0	0	0

Now, we do a bit of transferring

	Effect			Deposit/(Withdrawal)			Aggregate Position			Unsettled Position			Settled Position			NDC		
Action	**DFSP A**	**DFSP B**	**DFSP C**	**DFSP A**	**DFSP B**	**DFSP C**	**DFSP A**	**DFSP B**	**DFSP C**	**DFSP A**	**DFSP B**	**DFSP C**	**DFSP A**	**DFSP B**	**DFSP C**	**DFSP A**	**DFSP B**	**DFSP C**
DFSP A transfers $100 to DFSP B	-100	100					900	1100	1000	-100	100	0	1000	1000	1000	0	0	0
DFSP A transfers $250 to DFSP C	-250		250				650	1100	1250	-350	100	250	1000	1000	1000	0	0	0
DFSP B transfers $50 to DFSP C		-50	50				650	1050	1300	-350	50	300	1000	1000	1000	0	0	0
DFSP C transfers $75 to DFSP B		75	-75				650	1125	1225	-350	125	225	1000	1000	1000	0	0	0

Now, we try to breach our NDC

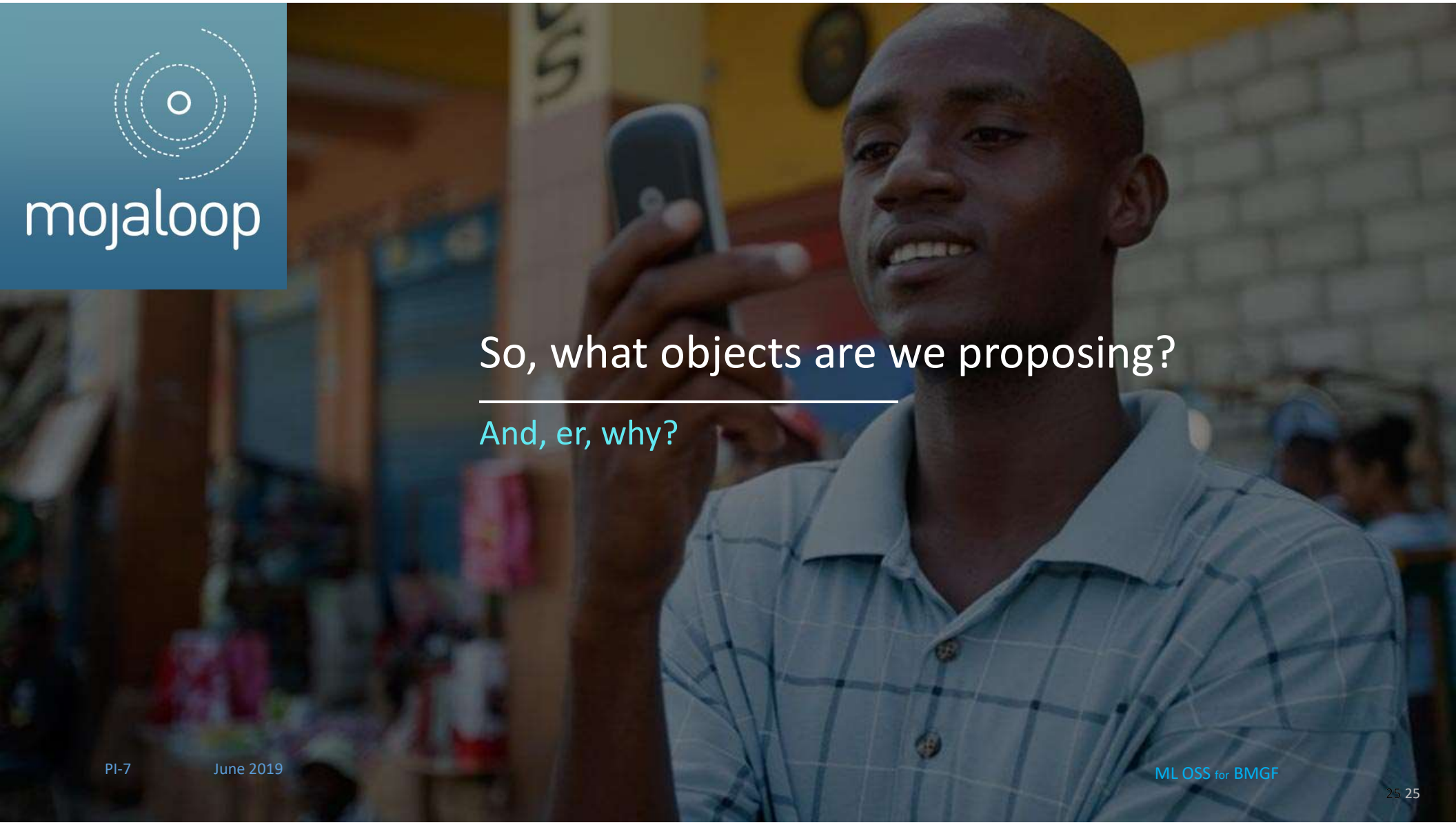
	Effect			Deposit/(Withdrawal)			Aggregate Position			Unsettled Position			Settled Position			NDC		
Action	**DFSP A**	**DFSP B**	**DFSP C**	**DFSP A**	**DFSP B**	**DFSP C**	**DFSP A**	**DFSP B**	**DFSP C**	**DFSP A**	**DFSP B**	**DFSP C**	**DFSP A**	**DFSP B**	**DFSP C**	**DFSP A**	**DFSP B**	**DFSP C**
DFSP A transfers $800 to DFSP C	-800		800				650	1125	1225	-350	125	225	1000	1000	1000	**0**	0	0

So, let's try settling…

	Effect			Deposit/(Withdrawal)			Aggregate Position			Unsettled Position			Settled Position			NDC		
Action	**DFSP A**	**DFSP B**	**DFSP C**	**DFSP A**	**DFSP B**	**DFSP C**	**DFSP A**	**DFSP B**	**DFSP C**	**DFSP A**	**DFSP B**	**DFSP C**	**DFSP A**	**DFSP B**	**DFSP C**	**DFSP A**	**DFSP B**	**DFSP C**
Settlement							650	1125	1225	-350	125	225	1000	1000	1000			
DFSP A transfers $1500 to DFSP B	-1500	1500					650	1125	1225	-350	125	225	1000	1000	1000	**0**	0	0
DFAP A transfers $250 to DFSP C	-250		250				400	1125	1475	-600	125	475	1000	1000	1000	0	0	0
Settlement completed							400	1125	1475	-250	0	250	650	1125	1225	0	0	0

And again…

	Effect			Deposit/(Withdrawal)			Aggregate Position			Unsettled Position			Settled Position			NDC		
Action	**DFSP A**	**DFSP B**	**DFSP C**	**DFSP A**	**DFSP B**	**DFSP C**	**DFSP A**	**DFSP B**	**DFSP C**	**DFSP A**	**DFSP B**	**DFSP C**	**DFSP A**	**DFSP B**	**DFSP C**	**DFSP A**	**DFSP B**	**DFSP C**
DFSP A transfers $600 to DFSP B	-600		600				400	1125	1475	-250	0	250	650	1125	1225	**0**	0	0
DFSP A pays $1000 into settlement account				1000			1400	1125	1475	-250	0	250	1650	1125	1225	0	0	0
Retry $600 transfer	-600		600				800	1125	2075	-850	0	850	1650	1125	1225	0	0	0
Settlement							800	1125	2075	-850	0	850	1650	1125	1225	0	0	0
Settlement completed							800	1125	2075	0	0	0	800	1125	2075	0	0	0



So, what objects are we proposing?

And, er, why?

Settlement models

1. Define a type of settlement and the ledger entry types which it includes

Why?

- The settlement type is decoupled from the ledger entries which record charges so that we can define settlements which include more than one type of charge.
- Gross settlement and net settlement are defined at this level.
- A settlement model may be bilateral or multilateral.
- A settlement model may require liquidity cover or it may not

Scheme accounts

1. Represent the ledgers which collect entries relating to a particular settlement model

Why?

- We want to be able to store the ledger entries relating to a particular settlement type in the same logical container.
- We want it to be possible for scheme administrators to define the ledger entry types which belong to a particular settlement without requiring new code to be written

Ledger entry types

1. Implement different types of charge for individual participants

Why?

- We want to be able to define multiple types of charge (for instance, processing fees, interchange fees.)
- We want it to be possible for scheme administrators to define these charge types and associate them with scheme accounts without requiring new code to be written

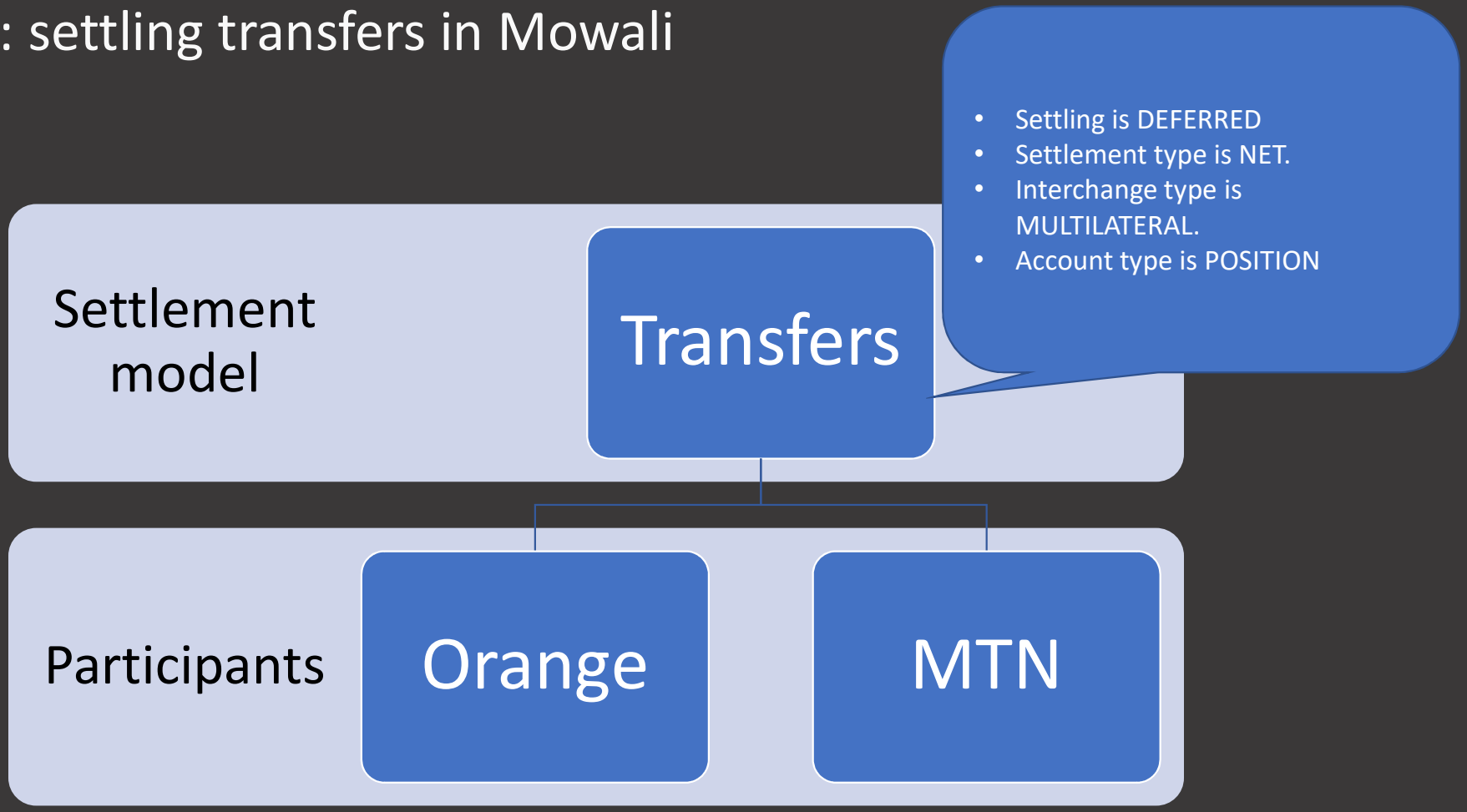
Net Debit Caps

1. Define variations to the liquidity cover requirement for one or more groups of charge types (= scheme accounts)
2. They may be different for each participant in a scheme

Why?

1. The scheme needs to be able to vary the liquidity cover requirement for different types of charge.
2. The scheme needs to be able to encourage participation and manage risk by varying the liquidity cover requirement for individual participants in the scheme.
3. The scheme should be able to change both of these policies *ad hoc*.

Example: settling transfers in Mowali

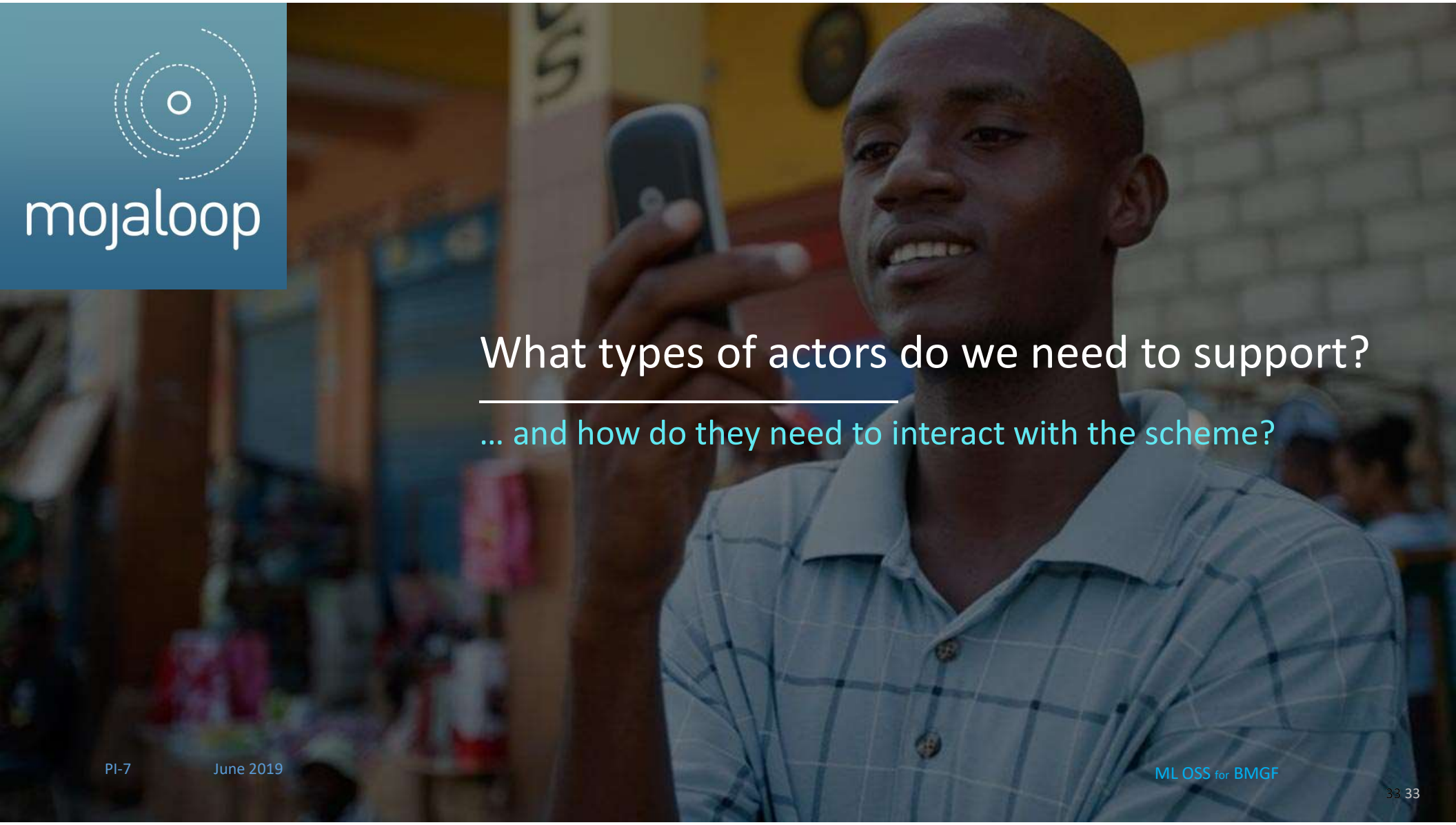


Example: settling transfers in TIPS

- Settling is IMMEDIATE
- Settlement type is GROSS.
- Interchange type is BILATERAL.
- Account type is POSITION

Example: settling interchange fees in TIPS

- Settling is DEFERRED
- Settlement type is NET.
- Interchange type is MULTILATERAL.
- Account type is INTERCHANGE_FEE


mojaloop
What types of actors do we need to support?
… and how do they need to interact with the scheme?

Actors

1. Scheme policy-makers – do not interact directly with the Settlement API
2. Scheme administrators
3. Participant administrators
4. The scheme itself

What do scheme policy-makers need to do?

1. Decide on the settlement model
2. Decide on settlement timetables for deferred settlements
3. Decide on the types of charges to be supported and the rules for applying them
4. Decide the NDC policy

What do scheme administrators need to do?

1. Tell the scheme what types of settlement are required and, for each type:
    1. Whether settlement is gross or net
    2. Whether settlement types are multilateral or bilateral
    3. What scheme account type supports the settlement model
2. Tell the scheme which ledger entry types are associated with which scheme accounts
3. Tell the scheme what documentation to produce for each account for each net settlement and where to send it
4. Tell the scheme what ledger entry types to set up
5. Tell the scheme what scheme account types to set up
6. Tell the scheme how to calculate the ledger entries for each ledger entry type
7. For schemes which contain net settlement:
    a. Tell the scheme what settlement timetable to apply
    b. Request ad hoc settlements
8. Tell the scheme what NDCs to set up and what margins to apply to them
    a. Define alert limits for participants
9. Tell the scheme what deposits and withdrawals have been made to and from external accounts
10. Approve participant requests to withdraw funds from settlement accounts *(outside API)*
11. Review data relating to settlements

What do participant administrators need to do?

1. View historic, current and predicted liquidity requirements
2. Define NDC alert levels
3. Define liquidity cover reductions
4. Inform scheme administrators of deposits into settlement accounts *(outside API)*
5. Request scheme administrators to withdraw funds from settlement accounts *(outside API)*

What does the scheme need to do?

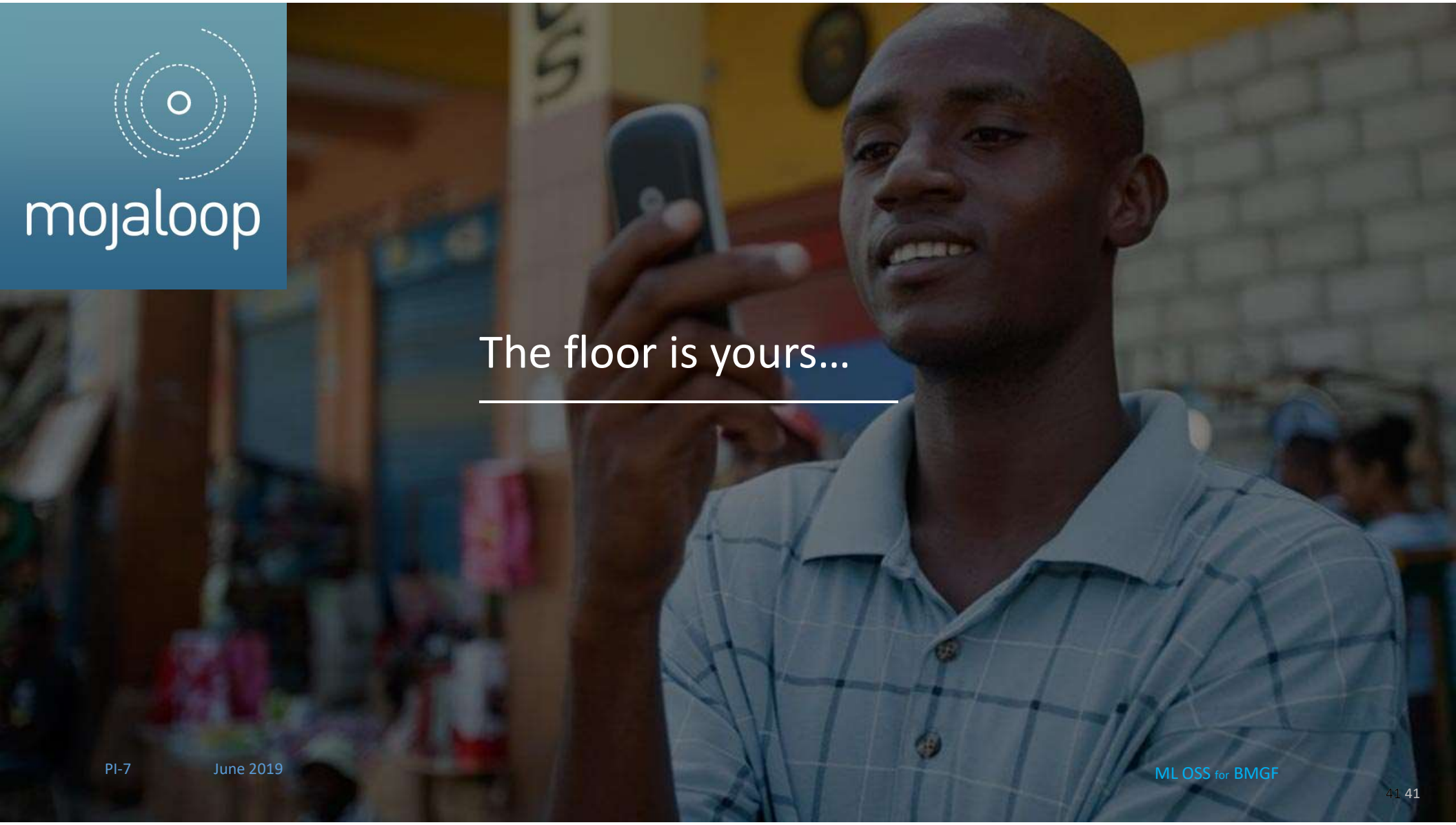
(Deep breath…)

What does the scheme need to do?

1. Settlement models:
    a. Store the required account types and, for each type, the settlement model to be followed
2. Net Debit Caps:
    a. Create an NDC
    b. Apply defined modifications to an NDC
    c. Check that a proposed transfer does not violate an NDC and reject it if it does.
    d. Alert participant administrators if their ledger position moves within a defined margin of their NDC
3. Ledger entry types
    a. Register a ledger entry type and the rules for creating ledger entries of that type
    b. Create a ledger for each participant in a ledger group
    c. Create entries in ledgers:
        i. Based on transfers
        ii. Based on the rules defined for each ledger group

What does the scheme need to do?

1. Settlements (net models only)
    a. Set up a timetable to generate events to create settlements between accounts of a given type
    b. Trigger *ad hoc* settlements
    c. Process a settlement
        i. Calculate the number of settlement instructions to be generated
        ii. Generate the content of each settlement instruction
            i. The net of all unsettled completed entries in ledgers contributing to the settlement point type requested in the currency requested.
        iii. Send each settlement instruction to the URI specified
2. Record the progress of a settlement
3. Inform the switch of the credits and debits to be recorded against a participant's ledger.


mojaloop
The floor is yours…

Appendix

Settlement API

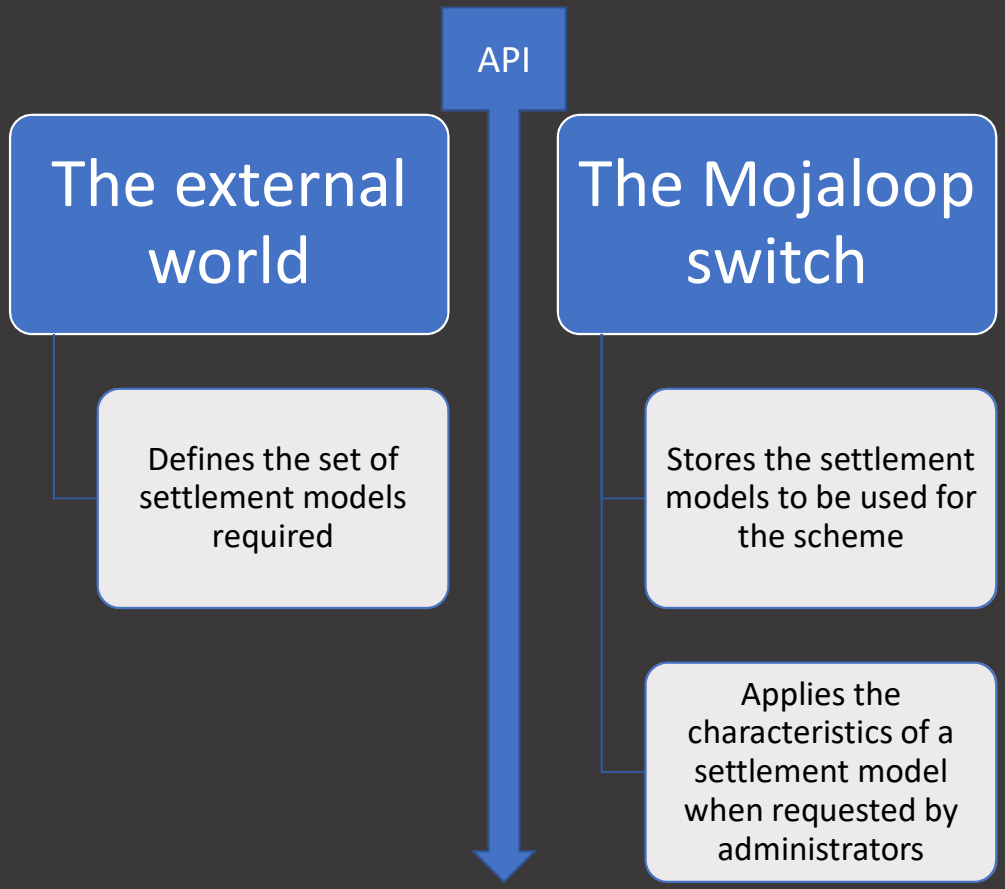
What areas does an API need to support?

1. Settlement model definitions
2. Scheme account types
3. Ledger entry type definitions
4. Settlement definition
5. Settlement instructions
6. Settlement reporting

Settlement models: characteristics

1. Scheme administrators can set up settlement models.
2. A settlement model defines a way in which participants settle with each other: all participants belonging to the same settlement model and currency settle with each other for the defined scheme account type.
3. A settlement model is defined as settling gross or net among its participating accounts
4. Settlement models may not be deleted, but they can be marked inactive. If a settlement model is inactive, no further settlements of that type may be performed.

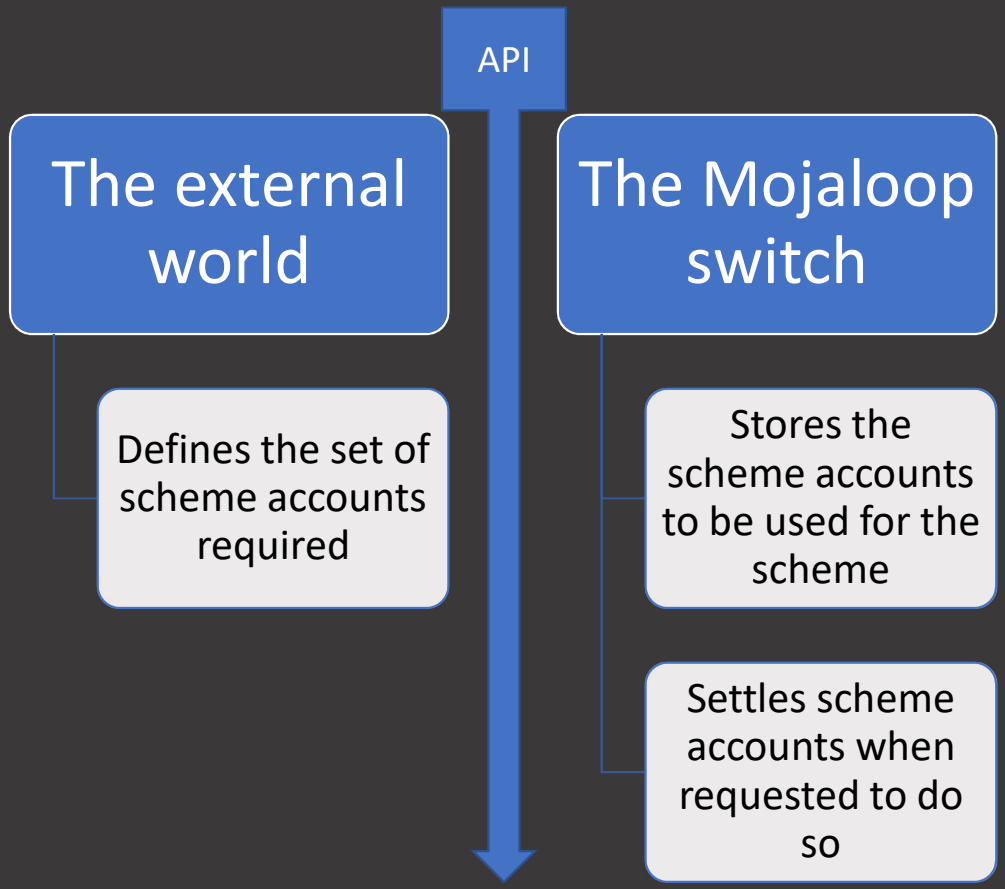
Settlement models: who does what?



Scheme account types: characteristics

1. Scheme administrators can set up scheme account types.
2. A scheme account type represents a ledger which stores all the entries which are settled according to the same settlement model.
3. A scheme account type can contain ledger entries of one or more ledger entry types.
4. Each ledger entry type can only belong to one scheme account type

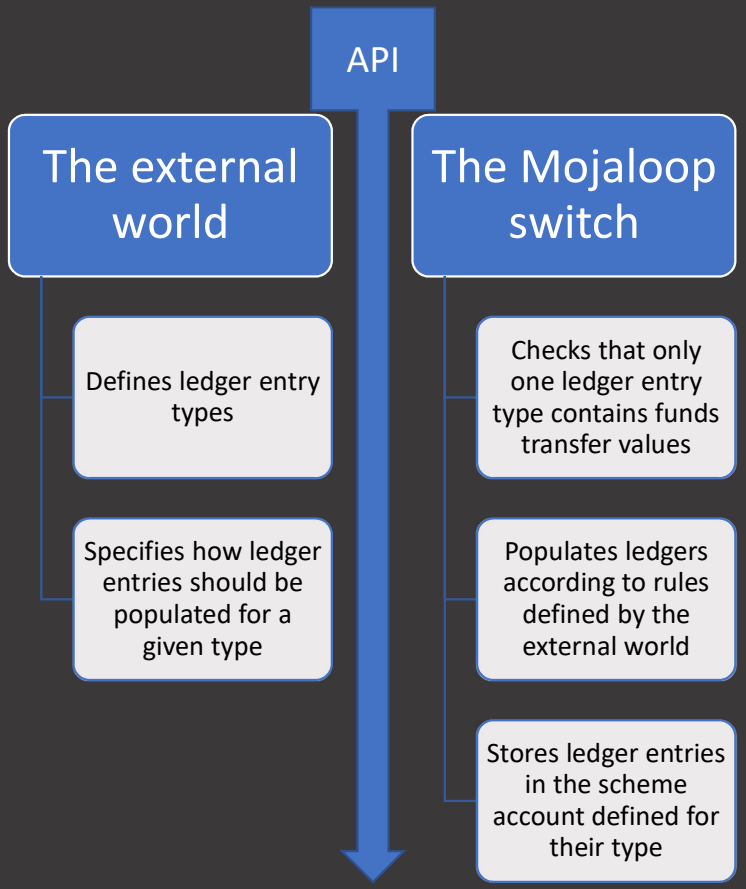
Scheme accounts: who does what?



Ledger entry types: characteristics

1. Scheme administrators can define ledger entry types via the API
2. Ledger entry types may be attached to a scheme account. This association means: values derived from this ledger entry type should be recorded against a participant's entries for the selected scheme account.
3. A ledger type may only be attached to a single scheme account
4. Once they have been created, ledger types may not be deleted. They may be deactivated, or detached from all settlement models.
5. A ledger of a given type can be populated *either* by funds transfer values *or* by values derived from funds transfer values by a deterministic process.
6. Only one ledger group in a scheme can be populated directly by funds transfer values.

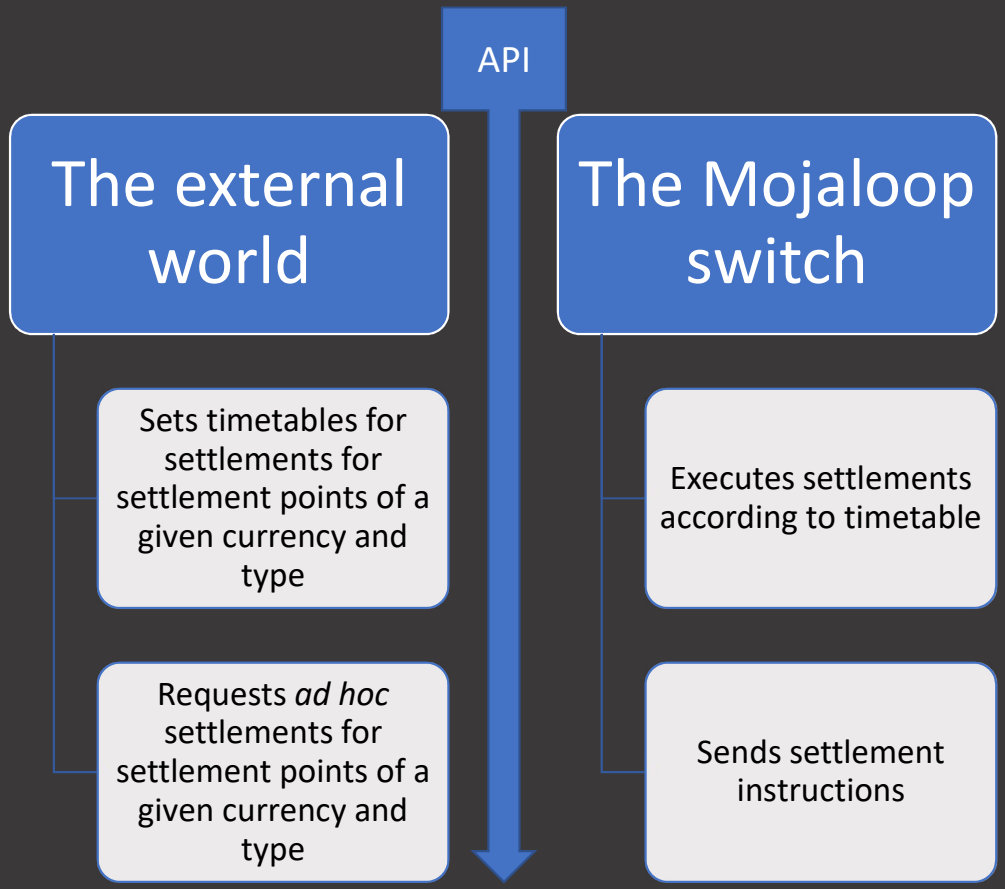
Ledger entry types: who does what?



Settlement definition: characteristics

1. Only applies to net settlements: by definition, gross settlements are transaction by transaction.
2. The timetable for settlements is defined by the scheme
3. Can be on a timetable or *ad hoc*

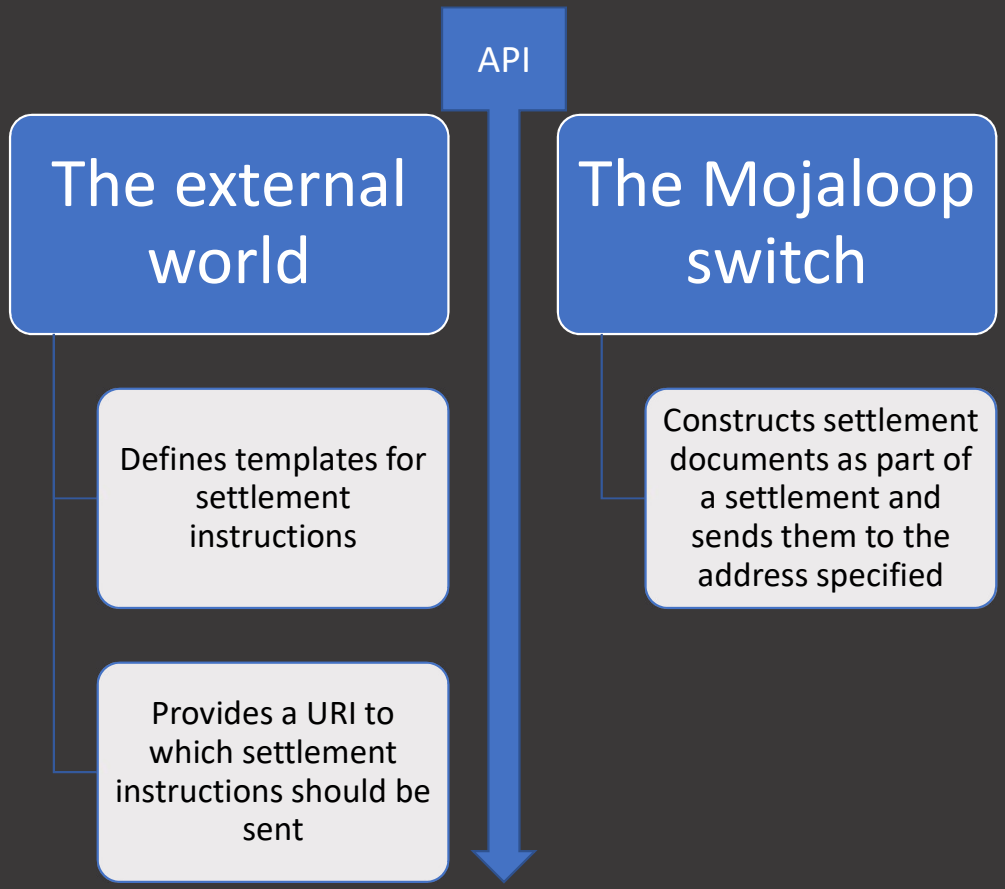
Settlement definitions: who does what?



Settlement instructions: characteristics

1. Apply to settlement accounts
2. Need to define:
   a. The form of an instruction to transfer funds from a net debtor or to a net creditor.
   b. A URL to which that instruction should be sent

Settlement instructions: who does what?



Settlement reporting: characteristics

1. Reports are produced by making GET requests on the following resources, which are described below:
   a. accountMatrix
   b. ledgergroupMatrix
   c. transactionWindowContent
   d. settlementContent
   e. ndcMatrix

So…

- Different types of settlement may be defined by settlement models
  - Each may be characterised as:
    - Settling gross or net
    - Settling bilaterally or multilaterally
    - Settling immediately or with delays
    - Requiring liquidity cover or not
  - Each settlement model has one or more account types associated with it
    - An account type might be “interchange fees” or “transfers”
    - Account types have ledger entry types associated with them
    - Associating a ledger entry type with an account type means: All ledger entries of this type are recorded in this account, and are settled via the settlement model defined for the account type

The process of settlement (1)

- Settlements are requested by specifying a settlement model and a currency.
- If settlement is requested for a settlement model which settles immediately, it will be rejected
- Otherwise…

The process of settlement (2)

1. Aggregate the ledger entries
    1. For the currency selected
    2. For the current settlement window.
    3. For the account type defined for the settlement model
    4. If the interchange type defined for the settlement model is “bilateral”, then aggregate entries at the party and counterparty level
    5. Otherwise, aggregate entries at the party level
2. If the interchange type defined for the settlement model is “bilateral”, then remove duplicate entries after checking that they match
    1. “Duplicate entry” means that the aggregation process will produce a “pays DFSP B” total for DFSP A and a “pays DFSP A” total for DFSP B
3. Produce a report by settlement point in the same format as the report by DFSP: for a net settlement model with individual settlement points, this will be identical with the current model.

The process of settlement (3)

- Record the process of settlement for each participant (in a multilateral settlement model) or for each pair of participants (in a bilateral settlement model) according to a defined process of settlement.
- Record any changes to a participant's position as a consequence of the settlement.

What objects should the API support?

1. settlementModel
2. schemeAccount
3. schemeAccountMatrix
4. ledgerEntryType
5. settlementTimetable
6. settlementContent
7. NDC
8. NDCMatrix

settlementModel

1. Members:
   a. Name: the name of the settlement model
   b. SettlementType: the settlement type (gross or net)to be followed for this settlement model
   c. interchangeType: the interchange type (multilateral or bilateral) to be followed for this settlement model
   d. settlementDelay: the delay (immediate or deferred) to be followed for this settlement model
   e. authorisation: the authorisation model (continuous or discontinuous) to be followed for this settlement model
   f. liquidityCoverRequired: TRUE if liquidity cover is required for this settlement model, otherwise FALSE
   g. account: the name of the account which is settled by this settlement model
2. Actions:
   a. GET
   b. POST
   c. PUT
   d. DELETE

accountType

1. Members:
   a. name: the name of the account type
   b. description: a description of the account type
   c. isActive: set TRUE if the account type is active, otherwise FALSE
2. Actions:
   a. GET
      - Note: there are internal account types which are not available for settlement. Only account types where the *settleable* flag is TRUE should be retrieved
   b. POST
      - Note: there are internal account types which are not available for settlement. The *settleable* flag should be set to TRUE for all account types set using these methods.
   c. PUT
   d. DELETE
      - Internally, this means: set the *isactive* flag to FALSE.

ledgerEntryType

1. Members:
   a. Name
   b. Description
   c. isActive: set TRUE if the ledger entry type is active, otherwise FALSE
   d. accountType: the name of the account to which ledger entries of this type should be posted
2. Actions:
   a. GET
   b. POST
   c. PUT
   d. DELETE
      - Internally, this means: set the *isActive* flag to FALSE

settlementTimeTable

1. Members:
   A. An array of *settlementEntry* objects:
      i. One of:
         a. A regular timetable in cron format
         b. A start time and an end time
         c. A start time only (= settlement window runs from start time until the start of the next settlement window)
         d. An end time only (= settlement window runs from the end of the last settlement window before the end time defined until the end time defined)
         e. Nothing (= settlement window runs from now until superseded by another settlement window)
         f. A start transfer ID and an end transfer ID
         g. A start transfer ID only (= settlement window runs from start transfer ID until the start of the next settlement window)
         h. An end transfer ID only (= settlement window runs from the end of the last settlement window before the end transfer ID until the end transfer ID)
         i. An array of transaction window IDs: the settlement should settle the transactions contained in these transaction windows.
      ii. ID: if the settlementEntry object represents a single settlement, this field contains a unique identifier for the settlement
2. Actions:
   A. GET
   B. POST
   C. PUT
   D. DELETE

settlementContent

1. Members:
    A. An array of *Settlement* objects:
        i. An array of *SettlementParticipant* objects, representing the consequence of the settlement for each of its participants:
            a. Participant name
            b. Participant currency
            c. Net credit or debit as a consequence of the transfers covered by this settlement
2. Actions:
    A. GET