mojaloop

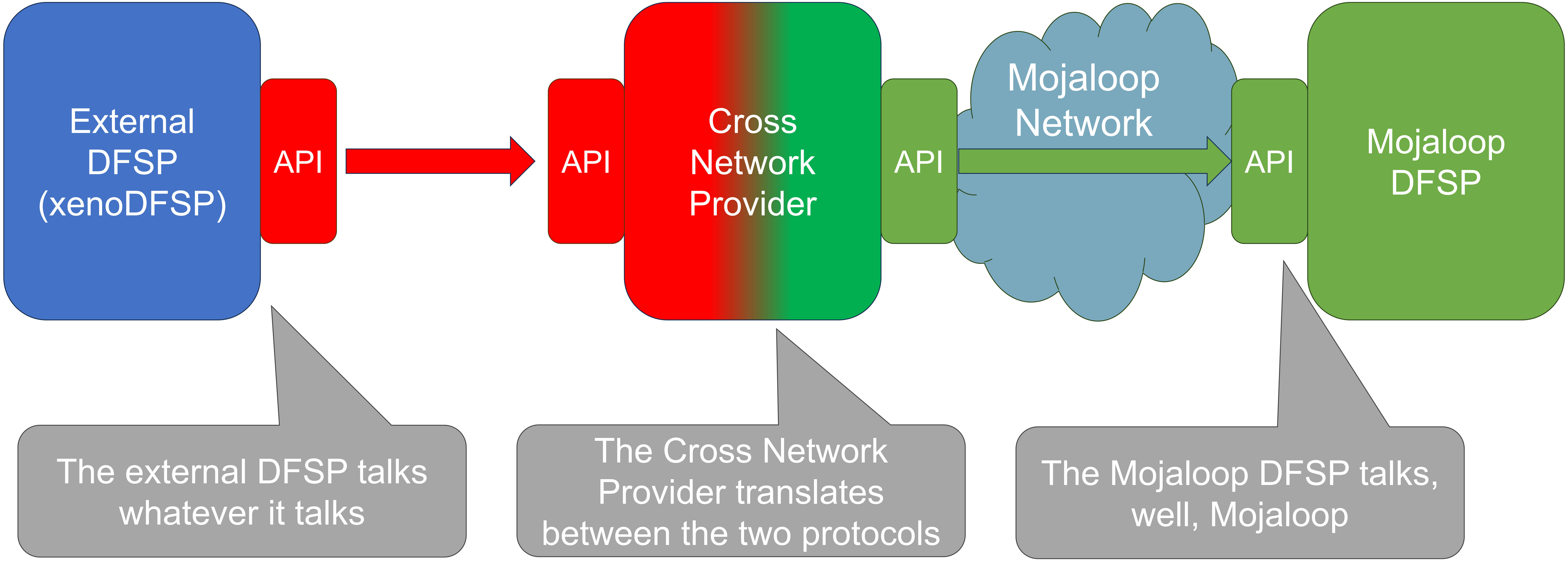
Initiatives in Cross-border payments

Topics

- Maintaining message integrity across schemes
- Settlement as a service
- Identifying parties in cross-jurisdictional payments

Maintaining message integrity across schemes

Our original thinking



The external DFSP talks whatever it talks

The Cross Network Provider translates between the two protocols

The Mojaloop DFSP talks, well, Mojaloop

Consequences of this approach

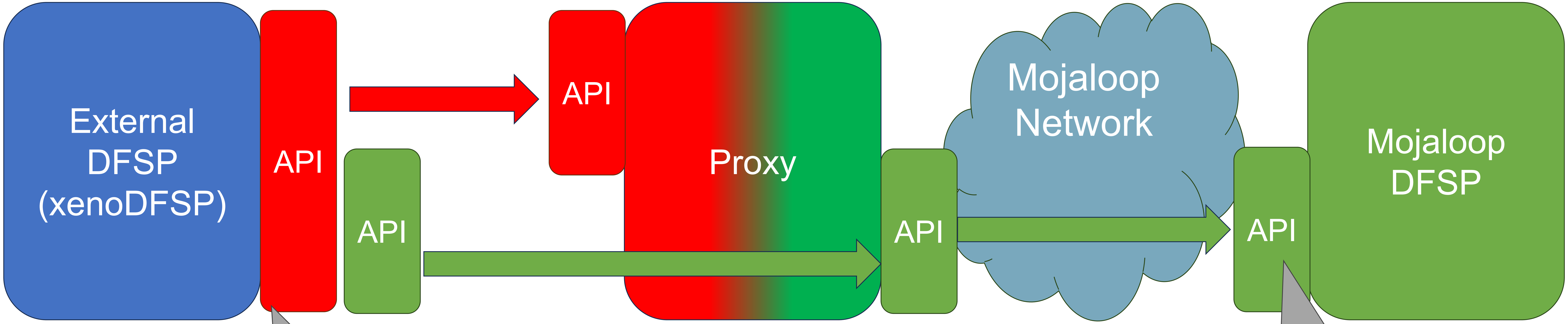
- There’s no end-to-end verification of message integrity
- The CNP settles on behalf of the xenoDFSP…
- … but obligations appear in the Mojaloop scheme as assigned to the CNP

Improvements to the ILF model

The original model

- Our standard CNP model:
- The xenoDFSP communicated with the CNP using the ILF Open Payments API.
- The CNP translated the requests back and forth between Open Payments API and FSPIOP
- The CNP settled on behalf of the xenoDFSP
- As far as the switch and the Mojaloop DFSPs were concerned, they were interacting with the CNP

Improving the Rafiki model



The external DFSP can talk more than one protocol

The proxy not longer needs to translate between the two protocols

The Mojaloop DFSP talks, well, Mojaloop

How does this process work?

- The xenoDFSP issues a payment request
- The Open Payments API establishes a connection with the proxy
- The proxy responds to say which types of payment it can support.
  - At the moment, we will support “ILP” and “Mojaloop”
- The proxy returns “Mojaloop”
  - This means: *You can route payments to me using the Mojaloop protocol, but not any other protocol*.
- Now the Rafiki instance knows: communication for this payment needs to use the Mojaloop protocol
- This decision is encapsulated in the Rafiki interface
  - The xenoDFSP continues to talk Open Payments API to the Rafiki instance.
  - This relies on resource correspondence between the Open Payments API and the Mojaloop API.

Sounds good…

- Now the Mojaloop connector in the Rafiki instance can attach a Mojaloop signature to its messages…
  - And the FSPIOP-Source parameter will be set to the eventual destination, not the proxy
- The Proxy will pass the messages through unmodified…
- And the recipient of the message can check that the message was sent by the xenoDFSP and that it has not been modified.

But wait!

- How is the recipient going to check the bona fides of the message?

Technical excursus

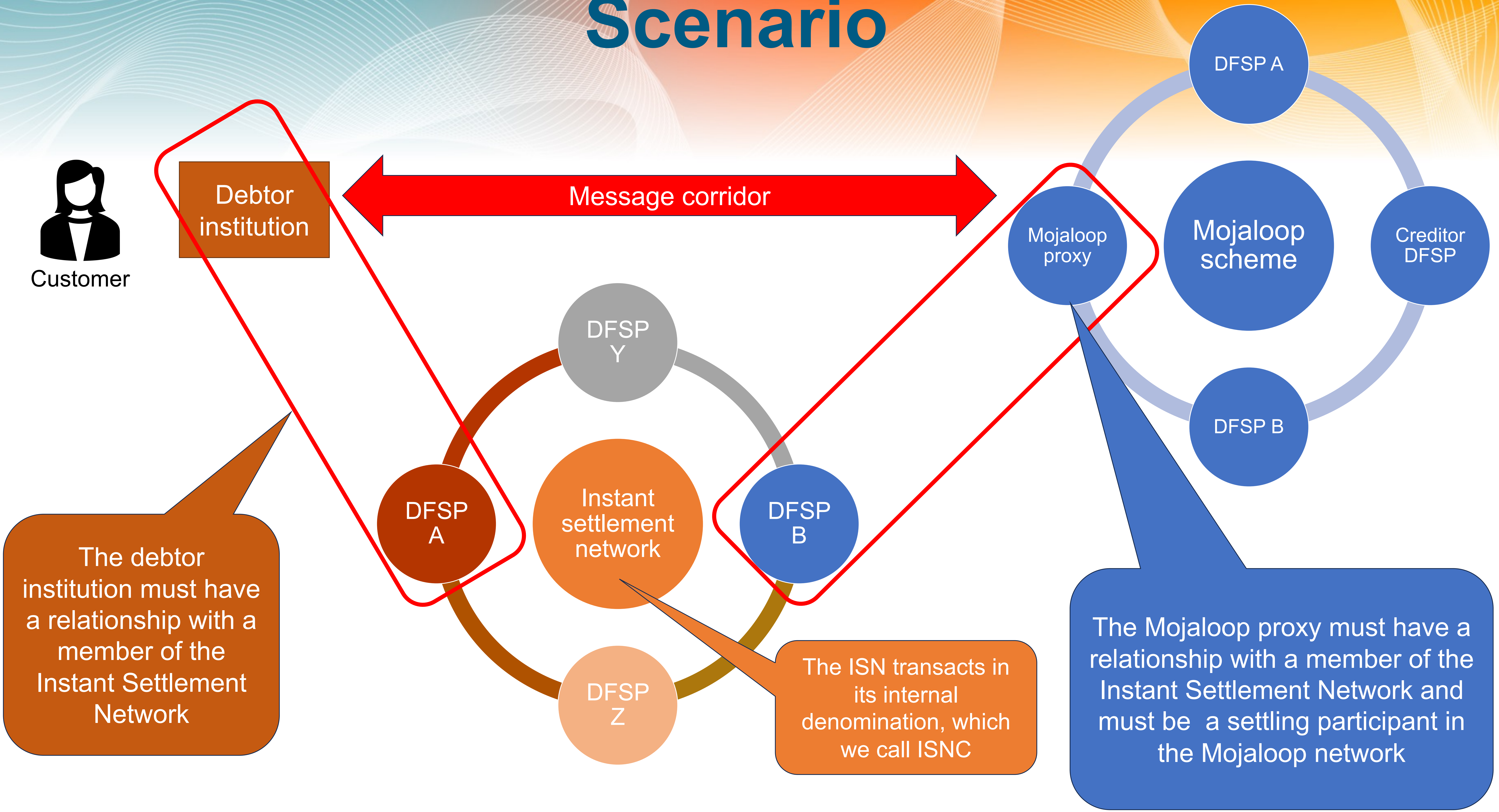
- The recipient of the message needs the sender’s public key to make this check.
- For Mojaloop DFSPs, the scheme’s relationship of trust allows public keys to be shared around the scheme.
- But we don’t want the xenoDFSP to have to register its public key with the Mojaloop scheme
  - We want the relationship of trust to be between the xenoDFSP and the Proxy…
  - … not between the xenoDFSP and the Mojaloop scheme.
- The public keys which are shared have two components:
  - The public key itself
  - A reference to the Certificate Authority which issued the public key
    - This allows participants to satisfy themselves that the key was issued by someone they trust.

How do we propose to get round this?

- We add a new resource to the Mojaloop API
- This resource enables a participant in the Mojaloop network to ask another participant for a specified public key.
    - So a DFSP receiving a message from a xenoDFSP can ask for the xenoDFSP's public key…
    - … and a xenoDFSP receiving a message from a Mojaloop DFSP can ask for the DFSP's public key.
- In both cases, the message containing the public key is signed by the Proxy:
    - The Proxy is trusted by the DFSP in virtue of their shared membership of the Mojaloop scheme…
    - … and is trusted by the xenoDFSP in virtue of the ILP pairing relationship which enables them to exchange payment messages with each other…
    - … which replaces the need for verification of the CA which issued the public key.

Settlement as a Service

Scenario



The debtor institution must have a relationship with a member of the Instant Settlement Network

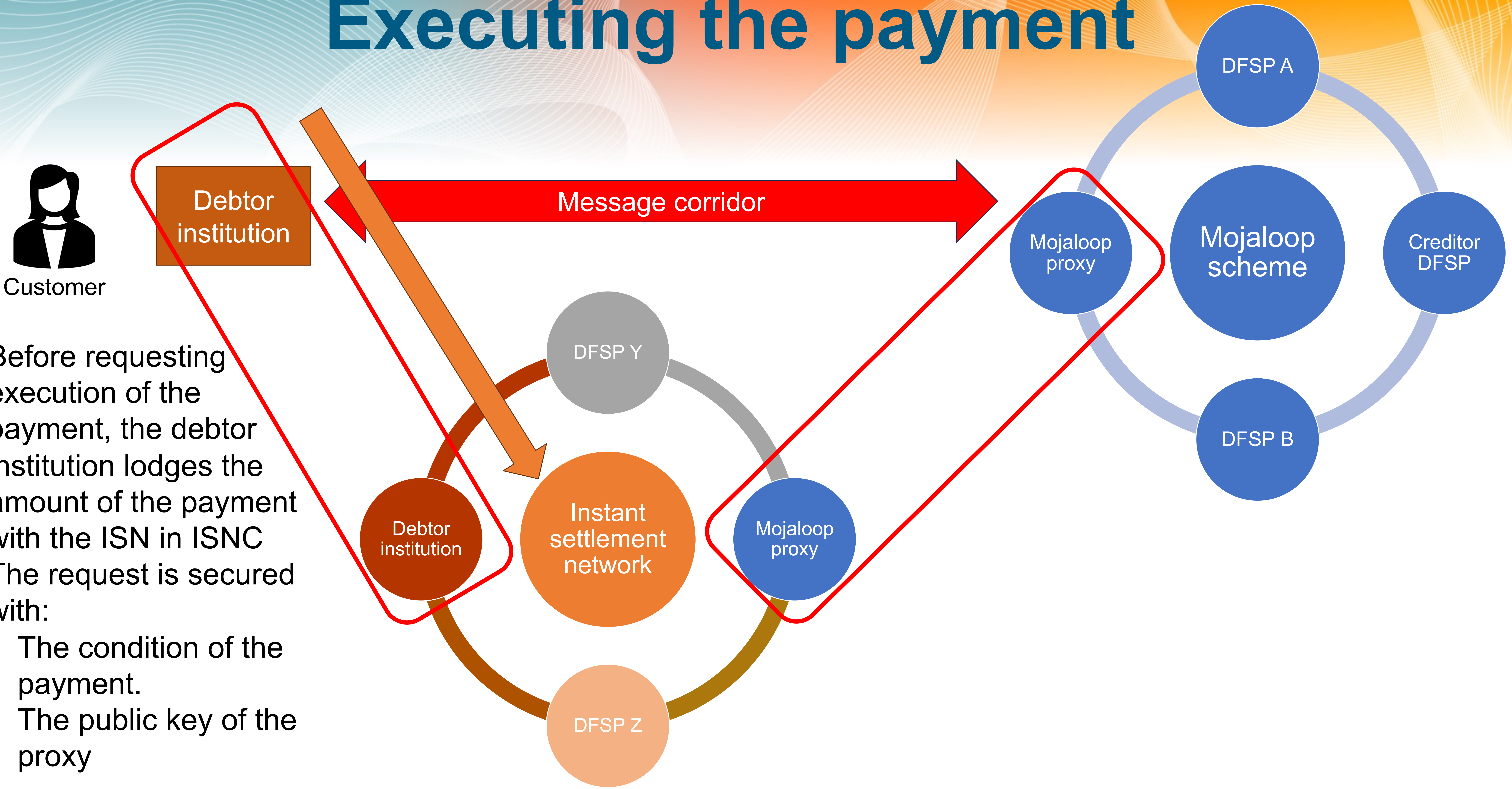
The ISN transacts in its internal denomination, which we call ISNC

The Mojaloop proxy must have a relationship with a member of the Instant Settlement Network and must be a settling participant in the Mojaloop network

Currency conversion

- Currency conversion will be requested by the debtor institution.
  - If the debtor wants to send in their home currency, then the debtor institution should decide what amount of ISNC it can purchase for that amount of local currency…
- In any case, the debtor institution will denominate the currency conversion request as ISNC to target currency
- The proxy will act as the FXP.
- Conversion agreement takes the following form:
  - The proxy defines the rate at which it will convert from the ISN's internal denomination (ISNC) to the target currency.
  - The proxy converts the response to Mojaloop's format and returns it to the debtor institution.
    - This will be denominated in ISNC. Approval says: *you must provide me with this amount of ISNC as collateral for the payment*.
  - The debtor institution requests agreement of terms from the creditor institution using the creditor institution's home currency.
  - From now on, all Mojaloop interactions take place in the target currency.
    - The payment appears to the Mojaloop scheme as a single-currency payment.
- What happens if the creditor institution imposes additional charges on the sender?
  - The debtor institution re-quotes to obtain from the ISN the amount in source currency that it needs to send

Executing the payment



1. Before requesting execution of the payment, the debtor institution lodges the amount of the payment with the ISN in ISNC
2. The request is secured with:
   a. The condition of the payment.
   b. The public key of the proxy

A new kind of liquidity check

- At present, we make the assumption that the liquidity which is checked is:
  - Available for inspection at any time (that is, a static bulk amount which is available to cover any transfer)
  - Guaranteed by the scheme (because only scheme administrators are allowed to modify it.)
- It’s analogous to a bulk purchase of currency
- For SaaS, we expect that a PvP check will be more appropriate
  - A Mojaloop proxy may be providing liquidity check services for multiple sources of inbound payments
  - Managing those requirements will be difficult, risky and hence expensive.
  - It would be more efficient if we could obtain confirmation from the Instant Settlement Network that it was holding funds for a particular payment…
  - … provided we trusted it, of course.
- So we suppose an agreement between a Mojaloop scheme and an ISP which supports a general relationship of trust between the two.

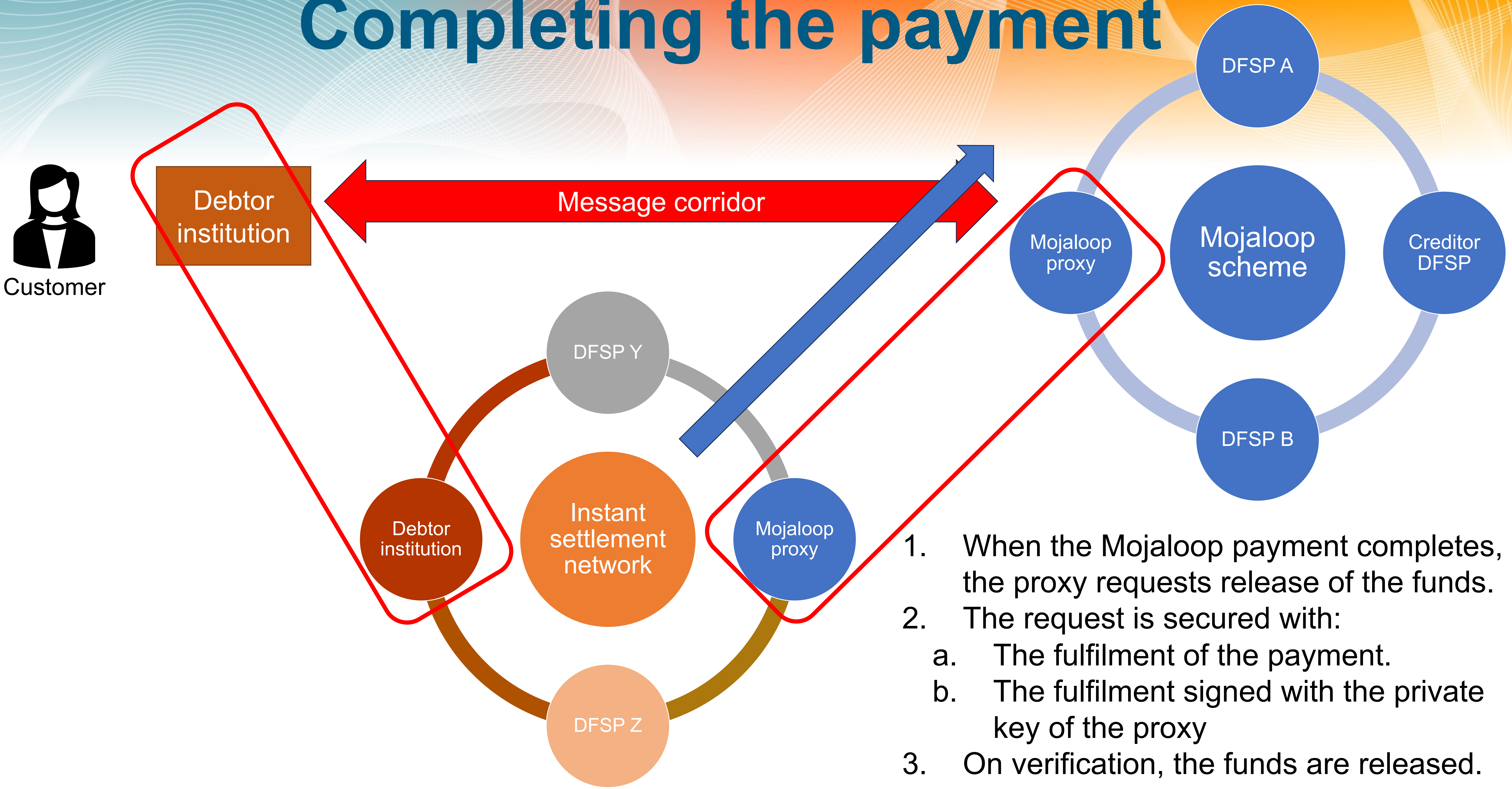
How does the liquidity check work?

- The sending institution deposits funds with the ISN prior to the execution of the payment
    - It identifies those funds to the ISN using the condition attached to the agreed terms of the payment.
- The switch enquires of the proxy whether it has the funds to cover the proposed payment.
- The proxy asks the ISN if it has received funds to cover the transfer
    - It knows the condition associated with the payment because the condition is associated with the transfer execution request which has triggered the liquidity check.
    - These funds are hypothecated to the payment: they are specific to that payment and can’t be used as cover for any other payment.
    - The ISN is responsible for guaranteeing that this is true
- The proxy sends the ISN:
    - A challenge to be checked
    - A signature to the challenge obtained by using its private key
- The ISN can use the public key it received from the debtor institution to verify that the challenge was signed with the correct key.
- If the ISN replies in the affirmative, then this is an equivalent to passing the liquidity check and reserving the funds…
- … except that the proxy doesn’t have to provide the liquidity.
- The liquidity is delegated to the ISN

Completing the payment

- The proxy receives notification from the switch that the payment has completed successfully.
- It uses the fulfilment which it received from the payee DFSP via the switch to request transfer of the funds held by the ISN to its settlement account
- It also signs the fulfilment with its private key.
- The ISN can verify that this is a legitimate request by:
    - Comparing the fulfilment with the condition that the debtor party sent with the original transfer request.
    - Confirming the signature of the fulfilment using the public key sent with the original transfer request.
- Following verification, the funds are unlocked and are directly available to the proxy.
- These funds are in ISNC: the proxy continues to be responsible for ensuring that settled funds are available in the target currency to support the process of settlement.

Completing the payment



1. When the Mojaloop payment completes, the proxy requests release of the funds.
2. The request is secured with:
   a. The fulfilment of the payment.
   b. The fulfilment signed with the private key of the proxy
3. On verification, the funds are released.

Identifying parties in cross-jurisdictional payments

Our motto:

Let

Sleeping

Data

Lie

Objectives

- Allow participants in an IPS to be confident that the parties with whom they are dealing meet compliance requirements…
  - … without sharing PII
- Should not require the participants to be members of a particular type of scheme…
  - … or, indeed, of any scheme at all.
- Retain PII data in the jurisdiction of origin.

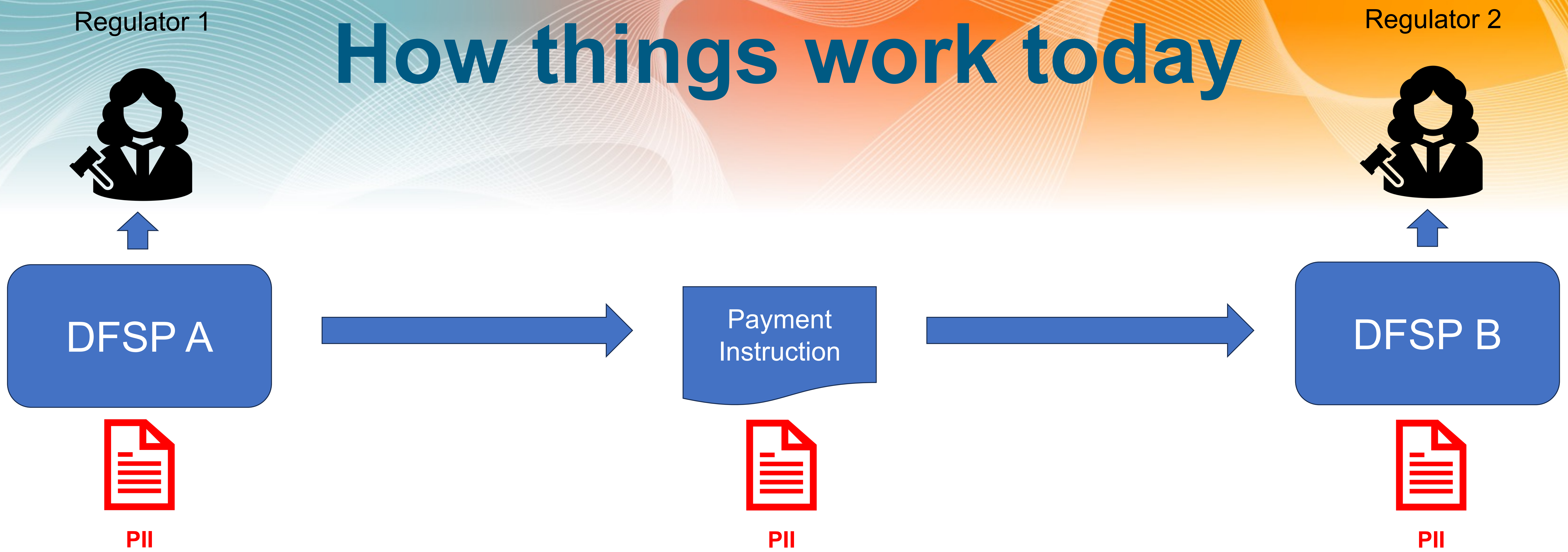
Assumptions

- Only PII data needs to be kept in the country of origin.
- A relationship can be established between the regulators of the jurisdictions to which the two parties to the payment belong
- Participant entities can reliably identify the state of the PII data that they held at a given point in time.

Prerequisites

- The regulator in jurisdiction A must agree with the regulator in jurisdiction B that:
  - The institutions regulated by regulator B are correctly regulated
  - There is a mutually agreed terminology for items of PII in the two jurisdictions
  - If regulator A requires information on a customer of an institution regulated by regulator B, they can obtain that information on demand through a route nominated by regulator B.
  - Requesting such information imposes on regulator A the same responsibilities on the safeguarding of PII as are in force in jurisdiction B

How things work today



- PII requirements must be explicitly agreed by the two regulators.
- A copy of the PII is attached to the payment instruction.
- It is stored by the creditor institution and any intermediaries.
- If a regulator wants to see the PII, it asks the party it regulates (or, perhaps, the IPS to which the party belongs)

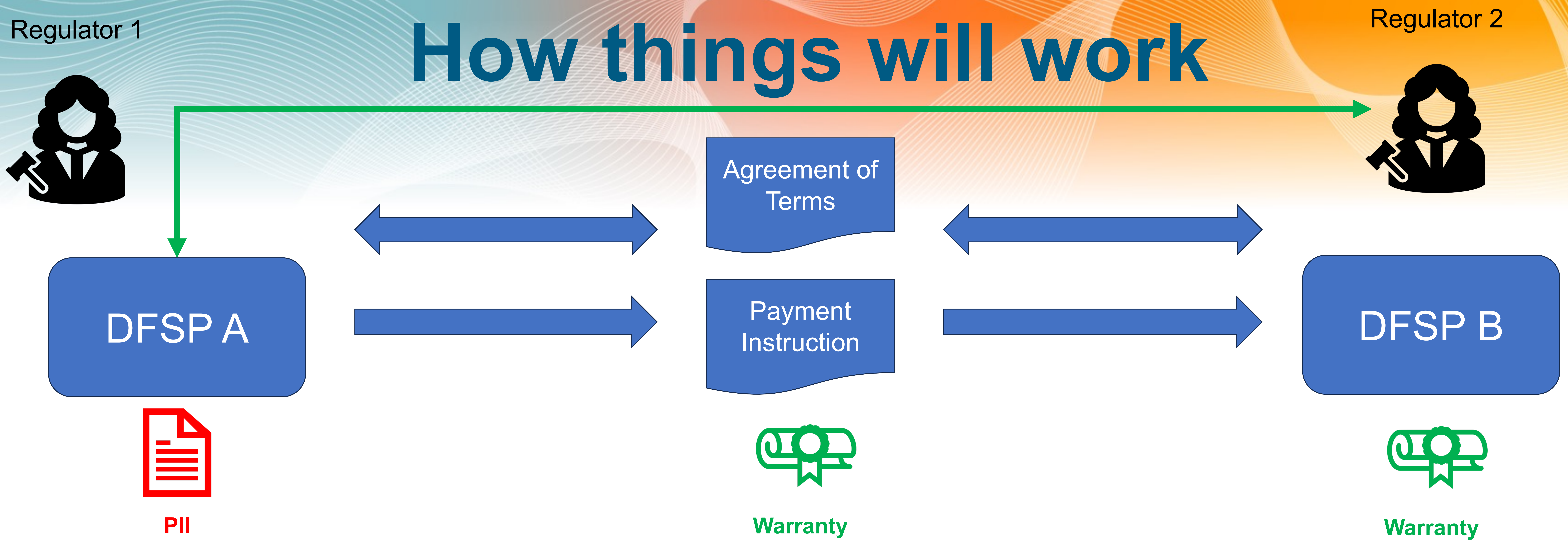
Drawbacks of this method

- PII elements must be agreed between the regulators
- PII content is distributed across multiple locations in multiple jurisdictions.
  - This may violate data sovereignty laws…
  - … and privacy legislation such as GDPR places a heavy burden on small institutions which may be ill-equipped to carry it.

What do we propose?

- PII remains in place at the owning DFSP
- The owning DFSP exchanges a warranty with the other parties to the payment.
    - The warranty says: *I am in possession of the following items of PII and I will make them available to qualified third parties on request.*
- A regulator can request the items of PII associated with a payment
    - This can be either directly from the DFSP or from an IPS to which the DFSP belongs.
    - The regulator provides a public key which the responding DFSP can use to ensure that the information requested can only be read by the requesting regulator.

How things will work



- PII requirements can be agreed *ad hoc* as part of the agreement of terms.
- A warranty that the PII will be provided on request is attached to the payment instruction.
- The warranty is stored by the creditor institution and any intermediaries.
- If a regulator wants to see the PII, it asks the party that owns the PII (or, perhaps, the IPS to which the party belongs)
- Regulator 1 warrants to regulator 2 that they will ensure that DFSP A's KYC procedures are compliant

Implementing in Mojaloop

Payment flow 1: requesting terms

- When the debtor DFSP requests the agreement of terms, it needs to include:
  - An internally generated unique ID representing the set of PII items that it holds for the sender.
  - A series of identifiers representing the items of PII that it holds for the sender
  - A series of identifiers representing the items of PII that it needs to verify at the receiver.
- These will be included in the attributes of a **party** complex type.
- The first two items will be part of the **payer** data item.
- The third item will be part of the **payee** data item.
- When the creditor DFSP responds, it adds a unique ID to the **payee** data item as a warranty that it can provide the PII requested
- It then signs the whole set of information as part of the agreed terms of the transfer

Snippets from POST /quotes

```
"payer": {
            "partyIdInfo": {
                        "partyIdType": "MSISDN"
                        "partyIdentifier": "33791832024",
                        "fspId": "BOUYGUES"
            }
          , "merchantClassificationCode": ""
          , "name": ""
          , "personalInfo": {}
          , "pii": {
                        "warrantyId": "clo2qua8s000108lf2ab2eipc"
                      , "piiItems": [
                                    "DOBI"
                                  , "ADDR"
                                  , "JUID"
                                  , "PSPT"
                                  , "MAIL"
                        ]
```

Here is my warranty that I have these items of information for this customer

Here are the items of PII information that I have for this customer

```
, "payee": {
            "partyIdInfo": {
                        "partyIdType": "MSISDN"
                        "partyIdentifier": "23052849366",
                        "fspId": "MYT"
            }
          , "merchantClassificationCode": ""
          , "name": ""
          , "personalInfo": {}
          , "pii": {
                        "piiItems": [
                                    "ADDR"
                                  , "JUID"
                                  , "PSPT"
                        ]
            }
```

Here are the items of PII information that I want you to supply for your customer

Snippet from PUT /quotes

```
, "payee": {
          "partyIdInfo": {
                    "partyIdType": "MSISDN"
                    "partyIdentifier": "23052849366",
                    "fspId": "MYT"
          }
          , "merchantClassificationCode": ""
          , "name": ""
          , "personalInfo": {}
          , "pii": {
                    "warrantyId": "clo2s5nzw000008jsfago1blb"
                    , "piiItems": [
                              "ADDR"
                              , "JUID"
                              , "PSPT"
                    ]
          }

}
```

The payee DFSP adds a warranty that it can provide the items of PII requested, and signs the transaction object, which contains all of this information.

What else needs to be done?

- Nothing.
- The warranties and the PII items will be available to parties to the payment and, through them, to their regulators.

How does PII get checked?

- Any regulator has an RBAC clearance from the scheme to access payment information where one of the parties is regulated by them.
- The regulator accesses the system via the Admin API.
  - They ask for information about a specified payment, giving the Transaction ID.
  - The switch collects information about the participant(s) who were parties to that payment.
  - It sends a request to each participant requesting the items of PII which the participant warranted that it would provide.
    - In addition, it sends the date and time at which the transfer completed.
    - This says: “I want to see the information as at that time.”
    - DFSPs will need to tag PII information with its validity date and time, so that they can respond accurately to these requests…
    - … but we assume that their regulator will insist on this capacity in any case.
  - The participant gets the content requested and returns it to the regulator via the Admin API response
- This will require a new FSPIOP API endpoint

PII security

- When a regulator requests items of PII, they will send a public key as part of their request.
- The responding participant will encrypt their PII information using this key.
- Now, only the regulator can see the PII information.

Questions and discussion

Answering data sovereignty issues in Mojaloop systems

Our motto:

Let

Sleeping

Data

Lie

Objectives

- Allow participants in an IPS to be confident that the parties with whom they are dealing meet compliance requirements…
  - … without sharing PII
- Should not require the participants to be members of a particular type of scheme…
  - … or, indeed, of any scheme at all.
- Retain PII data in the jurisdiction of origin.

Assumptions

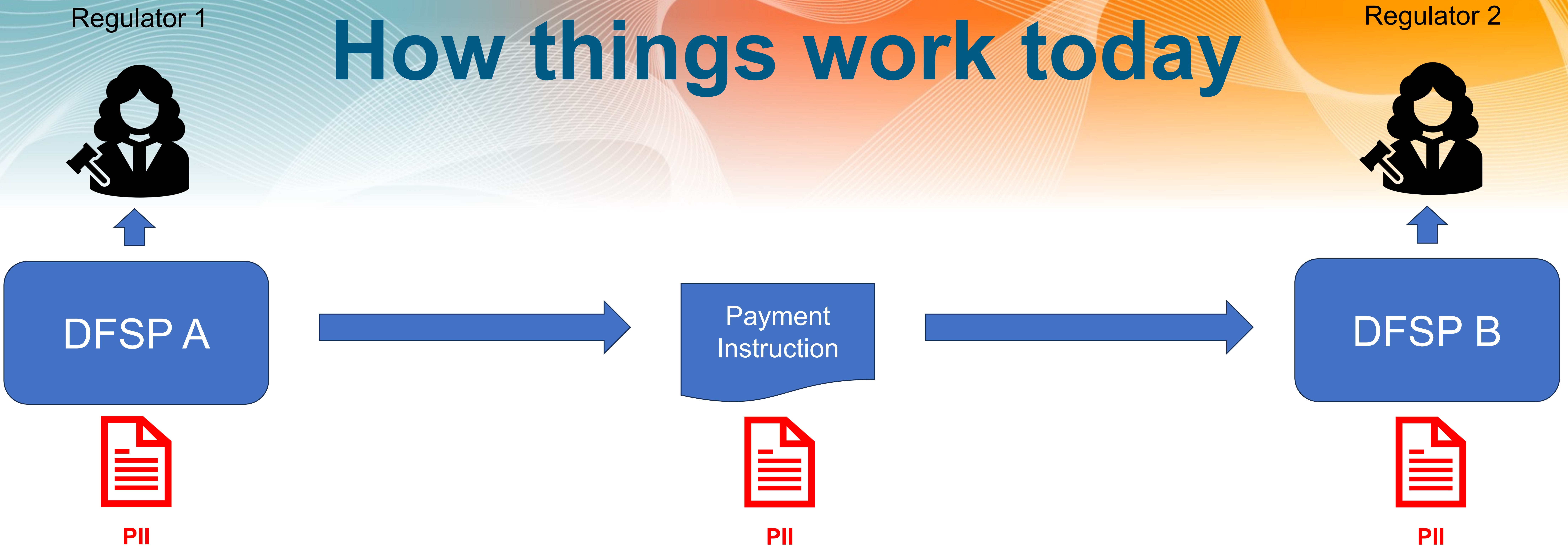
- Only PII data needs to be kept in the country of origin.
- A relationship can be established between the regulators of the jurisdictions to which the two parties to the payment belong
- Participant entities can reliably identify the state of the PII data that they held at a given point in time.

Prerequisites

The regulator in jurisdiction A must agree with the regulator in jurisdiction B that:

- The institutions regulated by regulator B are correctly regulated
- There is a mutually agreed terminology for items of PII in the two jurisdictions
- If regulator A requires information on a customer of an institution regulated by regulator B, they can obtain that information on demand through a route nominated by regulator B.
- Requesting such information imposes on regulator A the same responsibilities on the safeguarding of PII as are in force in jurisdiction B

How things work today



- PII requirements must be explicitly agreed by the two regulators.
- A copy of the PII is attached to the payment instruction.
- It is stored by the creditor institution and any intermediaries.
- If a regulator wants to see the PII, it asks the party it regulates (or, perhaps, the IPS to which the party belongs)

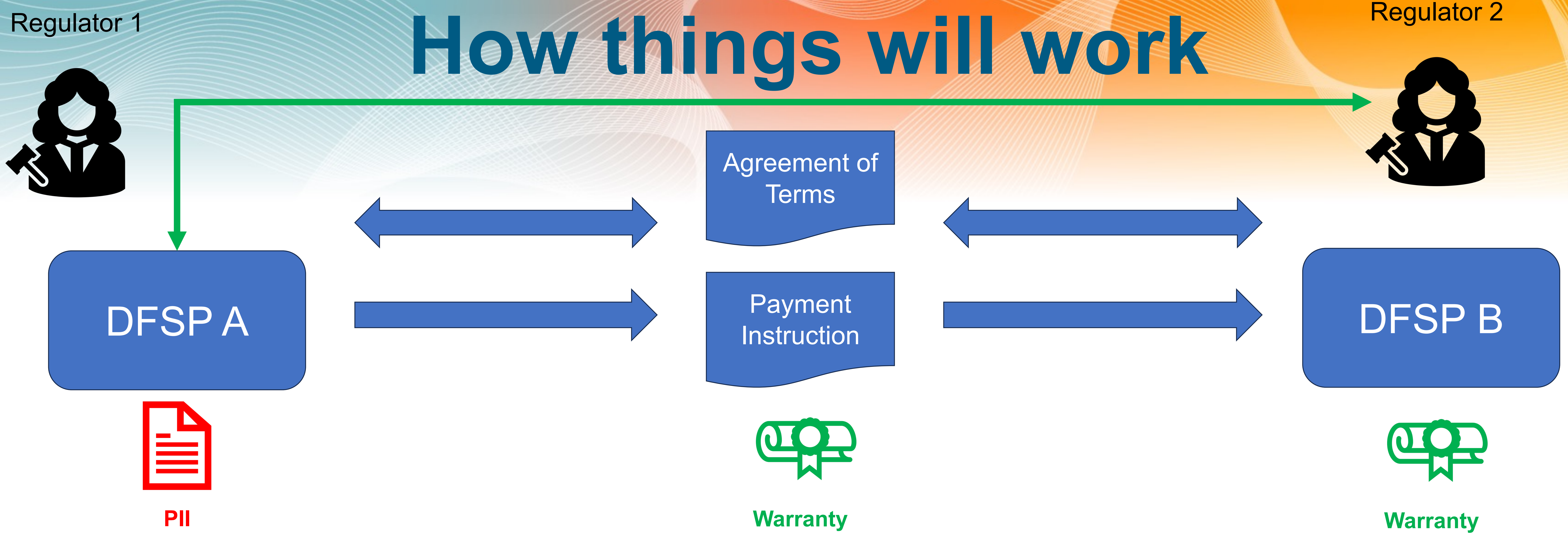
Drawbacks of this method

- PII elements must be agreed between the regulators
- PII content is distributed across multiple locations in multiple jurisdictions.
  - This may violate data sovereignty laws…
  - … and privacy legislation such as GDPR places a heavy burden on small institutions which may be ill-equipped to carry it.

What do we propose?

- PII remains in place at the owning DFSP
- The owning DFSP exchanges a warranty with the other parties to the payment.
  - The warranty says: *I am in possession of the following items of PII and I will make them available to qualified third parties on request.*
- A regulator can request the items of PII associated with a payment
  - This can be either directly from the DFSP or from an IPS to which the DFSP belongs.
  - The regulator provides a public key which the responding DFSP can use to ensure that the information requested can only be read by the requesting regulator.

How things will work



- PII requirements can be agreed *ad hoc* as part of the agreement of terms.
- A warranty that the PII will be provided on request is attached to the payment instruction.
- The warranty is stored by the creditor institution and any intermediaries.
- If a regulator wants to see the PII, it asks the party that owns the PII (or, perhaps, the IPS to which the party belongs)
- Regulator 1 warrants to regulator 2 that they will ensure that DFSP A’s KYC procedures are compliant

Implementing in Mojaloop

Payment flow 1: requesting terms

- When the debtor DFSP requests the agreement of terms, it needs to include:
  - An internally generated unique ID representing the set of PII items that it holds for the sender.
  - A series of identifiers representing the items of PII that it holds for the sender
  - A series of identifiers representing the items of PII that it needs to verify at the receiver.
- These will be included in the attributes of a **party** complex type.
- The first two items will be part of the **payer** data item.
- The third item will be part of the **payee** data item.
- When the creditor DFSP responds, it adds a unique ID to the **payee** data item as a warranty that it can provide the PII requested
- It then signs the whole set of information as part of the agreed terms of the transfer

Snippets from POST /quotes

```
"payer": {
            "partyIdInfo": {
                        "partyIdType": "MSISDN"
                        "partyIdentifier": "33791832024",
                        "fspId": "BOUYGUES"
            }
          , "merchantClassificationCode": ""
          , "name": ""
          , "personalInfo": {}
          , "pii": {
                        "warrantyId": "clo2qua8s000108lf2ab2eipc"
                      , "piiItems": [
                                    "DOBI"
                                  , "ADDR"
                                  , "JUID"
                                  , "PSPT"
                                  , "MAIL"
                        ]
```

Here is my warranty that I have these items of information for this customer

Here are the items of PII information that I have for this customer

```
, "payee": {
            "partyIdInfo": {
                        "partyIdType": "MSISDN"
                        "partyIdentifier": "23052849366",
                        "fspId": "MYT"
            }
          , "merchantClassificationCode": ""
          , "name": ""
          , "personalInfo": {}
          , "pii": {
                        "piiItems": [
                                    "ADDR"
                                  , "JUID"
                                  , "PSPT"
                        ]
            }
```

Here are the items of PII information that I want you to supply for your customer

Snippet from PUT /quotes

```
, "payee": {
          "partyIdInfo": {
                    "partyIdType": "MSISDN"
                    "partyIdentifier": "23052849366",
                    "fspId": "MYT"
          }
          , "merchantClassificationCode": ""
          , "name": ""
          , "personalInfo": {}
          , "pii": {
                    "warrantyId": "clo2s5nzw000008jsfago1blb"
                    , "piiItems": [
                              "ADDR"
                              , "JUID"
                              , "PSPT"
                    ]
          }

}
```

The payee DFSP adds a warranty that it can provide the items of PII requested, and signs the transaction object, which contains all of this information.

What else needs to be done?

- Nothing.
- The warranties and the PII items will be available to parties to the payment and, through them, to their regulators.

How does PII get checked?

- Any regulator has an RBAC clearance from the scheme to access payment information where one of the parties is regulated by them.
- The regulator accesses the system via the Admin API.
  - They ask for information about a specified payment, giving the Transaction ID.
  - The switch collects information about the participant(s) who were parties to that payment.
  - It sends a request to each participant requesting the items of PII which the participant warranted that it would provide.
    - In addition, it sends the date and time at which the transfer completed.
    - This says: “I want to see the information as at that time.”
    - DFSPs will need to tag PII information with its validity date and time, so that they can respond accurately to these requests…
    - … but we assume that their regulator will insist on this capacity in any case.
  - The participant gets the content requested and returns it to the regulator via the Admin API response
- This will require a new FSPIOP API endpoint

PII security

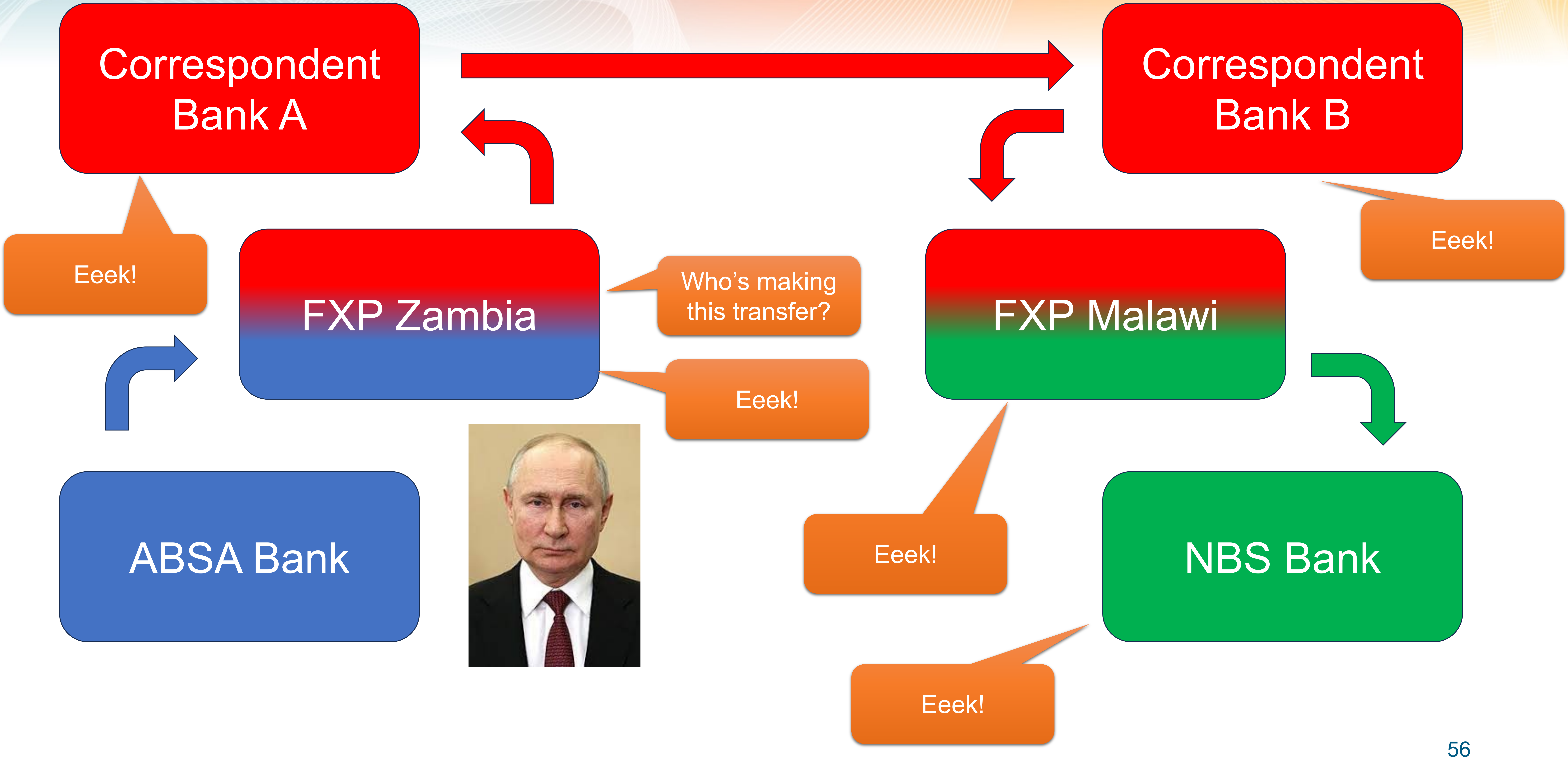
- When a regulator requests items of PII, they will send a public key as part of their request.
- The responding participant will encrypt their PII information using this key.
- Now, only the regulator can see the PII information.

Correspondent banking

Scenarios

- Transfers are between a customer of a bank in Zambia (the debtor) and a customer of a bank in Malawi (the creditor)
- High value, low volume transfers:
  - The money clears directly between the participant institutions using an FXP
  - The funds are converted into and out of USD
- Low value, high volume transfers:
  - The money clears indirectly between the jurisdictional Central Banks using the COMESA Clearing House (CCH)
  - The funds are converted into and out of USD

High value, low volume transfer



Correspondent Bank A

Correspondent Bank B

Eeek!

Eeek!

FXP Zambia

Who’s making this transfer?

FXP Malawi

Eeek!

Eeek!

ABSA Bank

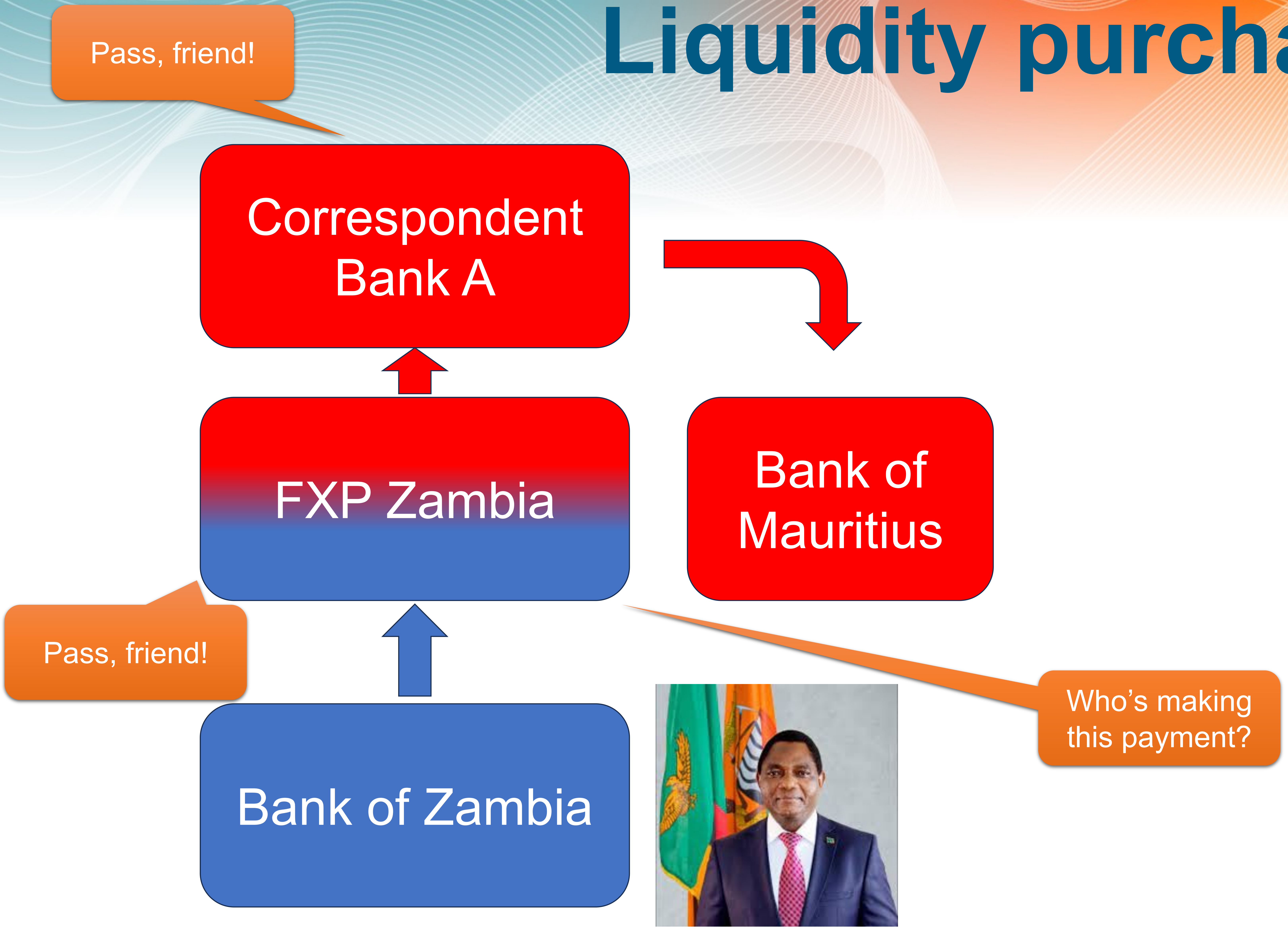
NBS Bank

Eeek!

Characteristics of low value, high volume transfers

- Liquidity cover is purchased in bulk…
- … by banks, not individuals
- Payments do not involve the movement of funds between accounts: they merely record obligations
- Settlements do involve the movement of funds between accounts…
- … but between banks, not individuals.

Liquidity purchase



Pass, friend!

Correspondent Bank A

FXP Zambia

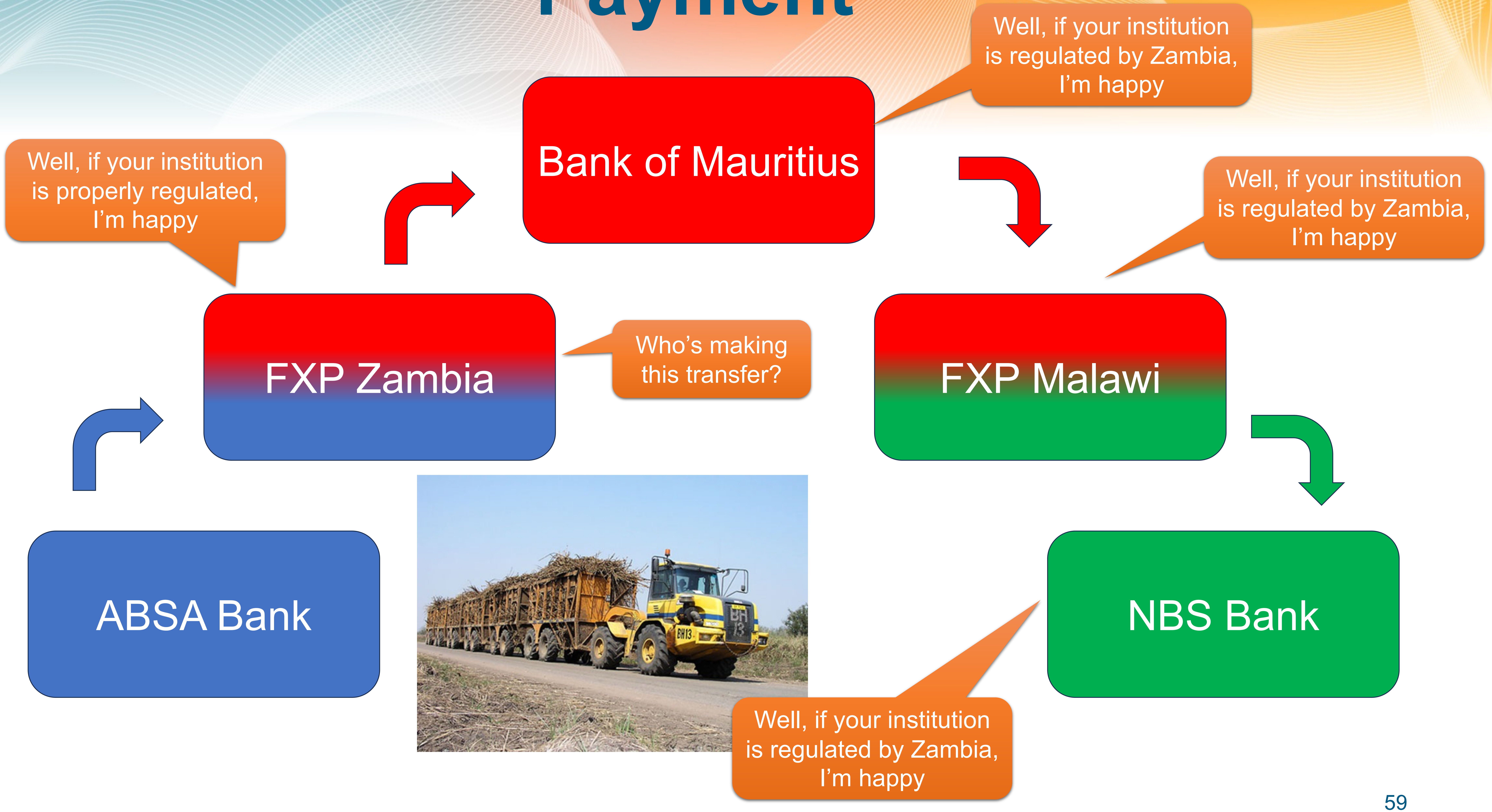
Bank of Mauritius

Pass, friend!

Bank of Zambia

Who’s making this payment?

Payment



ABSA Bank

FXP Zambia

Bank of Mauritius

FXP Malawi

NBS Bank

Well, if your institution is properly regulated, I'm happy

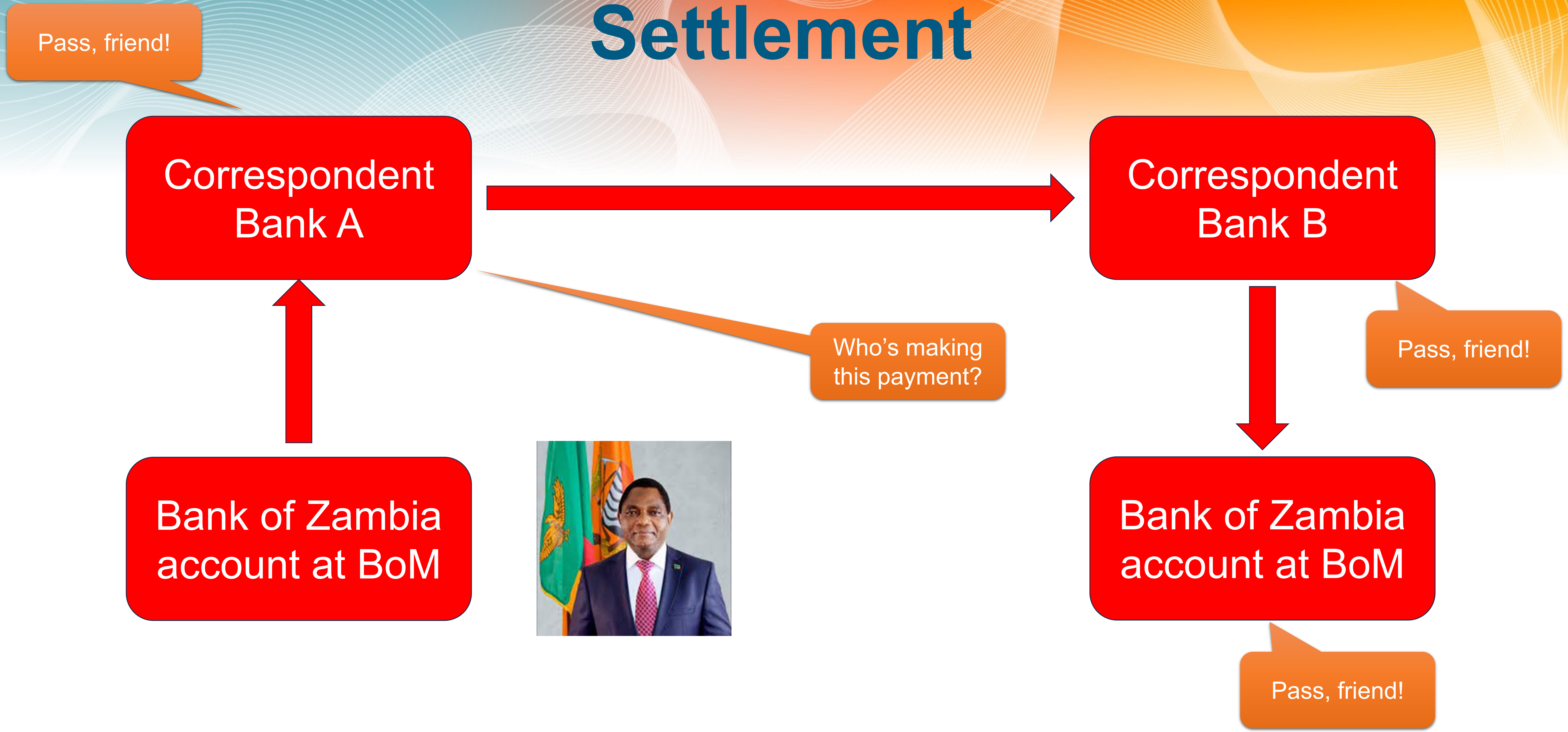
Who's making this transfer?

Well, if your institution is regulated by Zambia, I'm happy

Well, if your institution is regulated by Zambia, I'm happy

Well, if your institution is regulated by Zambia, I'm happy

Settlement



Pass, friend!

Correspondent Bank A

Correspondent Bank B

Who's making this payment?

Pass, friend!

Bank of Zambia account at BoM

Bank of Zambia account at BoM

Pass, friend!

Let’s talk in the break-out…