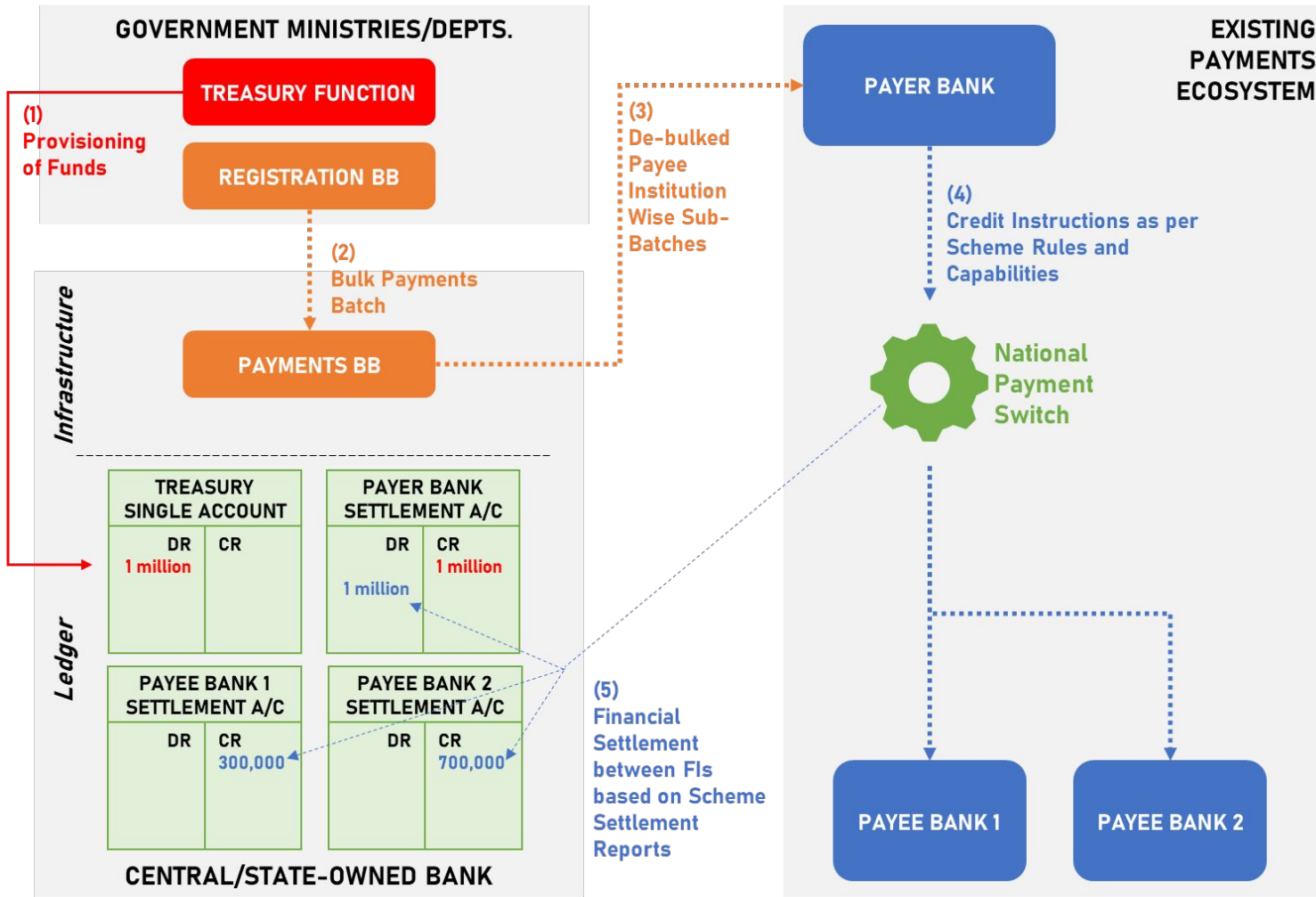


GovStack Architecture



What purpose will this Harmonization serve?

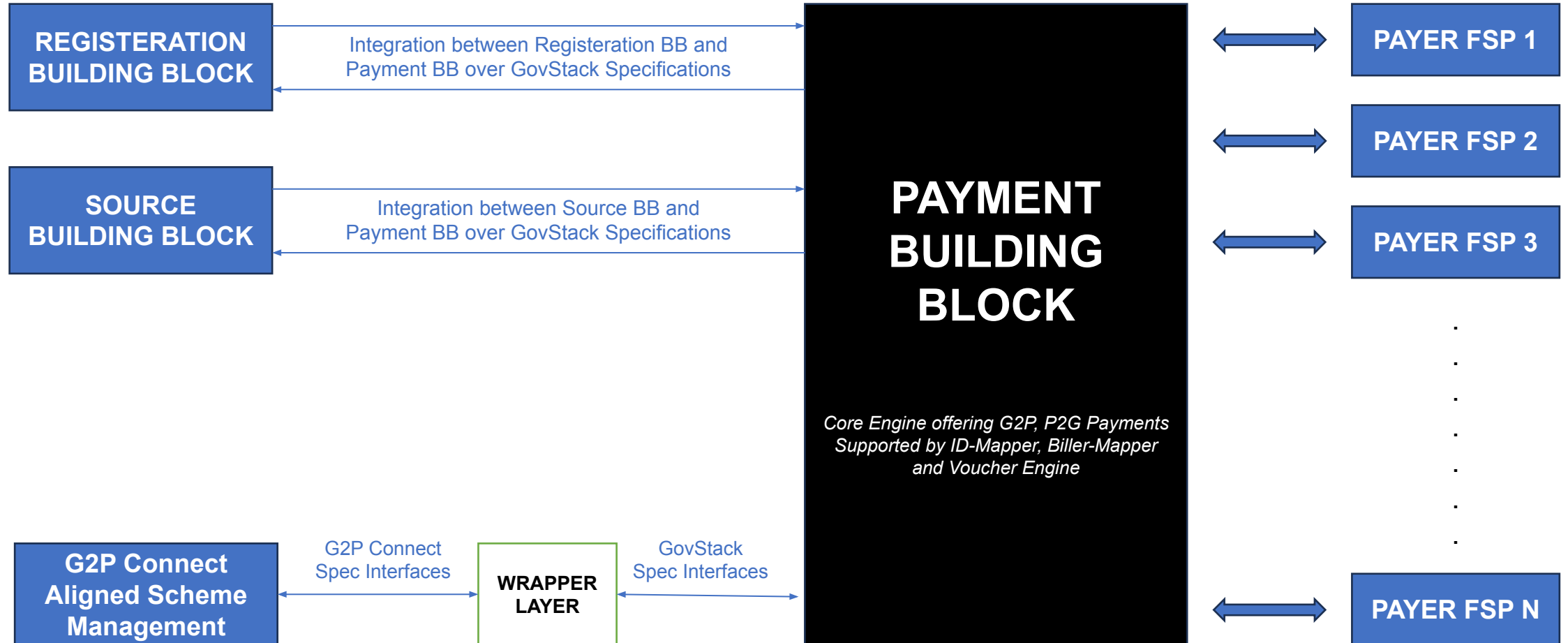
We are looking to cater situations where the “Source BB” or more specifically someone like Scheme Management/Social Benefit Program Management is implemented in alignment with G2P-Connect Specifications.

AND

Would like to interoperate with the Payment Building Block from the GovStack specifications to process payments or conduct disbursements.

What is the objective of this Harmonization Effort?

Approach to Harmonization



Principle 1:

There are no fundamental changes to the APIs themselves that have been developed under the GovStack specifications. There are fundamental differences between how G2P Connect is designed and how GovStack is being designed. To this end, this Wrapper Layer acts as an intermediary that makes it possible for these 2 platforms to work together seamlessly by mapping APIs and Attributes in GovStack Specs to G2P Connect Specs.

Principle 2:

There are various places in our specs where G2P connect passes through additional information which is not held in the Payment Hub, or Account Mapper or the Voucher Engine as designed per GovStack specifications. In such situations, we have proposed that the wrapper layer has *data retention capabilities as well, in certain situations the Wrapper Layer will hold data for a temporary period.*

Principle 3:

There are various places in our specs where G2P connect expects additional information, in some cases this is information which our current Specs do not hold and do not retain. By default, our wrapper looks to pass Fixed or Static information in these fields where possible, where not possible such points have been highlighted for mutual resolution with G2P connect or they dictate change in our API to receive and pass this information forward.

QUESTIONS TO G2P CONNECT

- Are there any objections on the principles defined in the previous slide for harmonization between G2PConnect and GovStack specifications?

Disbursement APIs

DISBURSEMENT APIs

#	Type	Endpoint	Description of Interface	GovStack Equivalent Interface <i>Reference API Specs v1.4</i>
1	Async	POST /disburse	Social protection platform initiating a G2P disbursement. Initiate payment through disbursement instructions	BulkPayment() POST /{{BulkHostName}}/batchtransactions?type=raw
2	Async	POST /on-disburse	Disbursement initiating systems receive disbursement status info through callback end points. Disburse response through callback	BulkPayment_StatusPush() to provide Batch Summary details. Case 1 : As a call back of Batch Payment. X-Callback-URL registered in the request GET BatchDetails API to fetch transaction level status of each batch and append that in the response back to on-disburse
3	Sync	/disburse/sync/disburse	Initiate payment through disbursement instructions through sync call	GovStack only has the Async interface in this regard, and by principle we do not see any value in implementing bulk payments through synchronous interfaces. Our approach therefore is to stick with an Async interface on the GovStack side of the integration and manage sync-async orchestration on the wrapper layer. BulkPayment() POST /{{BulkHostName}}/batchtransactions?type=raw

BATCH/TRANSACTION STATUS CHECK APIs

#	Type	Endpoint	Description of Interface	GovStack Equivalent Interface <i>Reference API Specs v1.4</i>
4	Async	POST /disburse/txn/status	Status check of previous disbursement transactions using transaction_id and/or reference_id(s)	Payment_Status_Check Request() Synchronous interface in GovStack specs. G2P Connect Specs are Async whereas our specs are Sync.
5	Async	POST /disburse/txn/on-status	Disbursement status to Social Protection, Treasury Systems.	None – Since the Payment_Status_Check Request() in GovStack specs is Sync in this case, there is no comparative endpoint for this interface.
6	Sync	/disburse/sync/search	Fetch disbursement processing status though sync call	There is no comparative endpoint for this interface in GovStack specifications.
7	Sync	/disburse/sync/txn/status	Sync status check of disburse Async APIs	GET BatchDetails API to fetch transaction level status of each batch

OTHER APIS

#	Endpoint	Description of Interface	GovStack Equivalent Interface <i>Reference API Specs v1.4</i>
8	POST /disburse/search	Search disbursement by searchable attributes	There is no comparative endpoint for this interface in GovStack specifications.
9	POST /disburse/on-search	Disbursement search results through callback	There is no comparative endpoint for this interface in GovStack specifications.

QUESTIONS TO G2P CONNECT

- G2P Connect has proposed “synchronous” interfaces for initiating bulk disbursements as well. As a point of principle we do not agree with this and believe that bulk disbursement should not be initiated in a synchronous manner. Our initial question is if we can align on this then can we can choose to not accommodate the synchronous interface at all.
- If we have to support synchronous disbursement interfaces, we have proposed an approach for that further on in [\(3\) SYNC INTERFACE TO INITIATE A DISBURSEMENT](#). However, we would suggest not implementing such a complicated approach and do disbursements only asynchronously as that is how bulk payments tend to behave usually.
- G2P connect has API interfaces that allow for “searching” for particular transactions using some criteria or filter, for example search for transactions based on Payee Financial Address. The GovStack specifications do not have an interface for such functionality, in the GovStack specifications, such functionality is developed at the GUI level where the Hub UI allows for searching for transactions based on different filters. However, this is not available through APIs. Is this absolutely needed? If yes we will look into developing such APIs in the GovStack specifications.

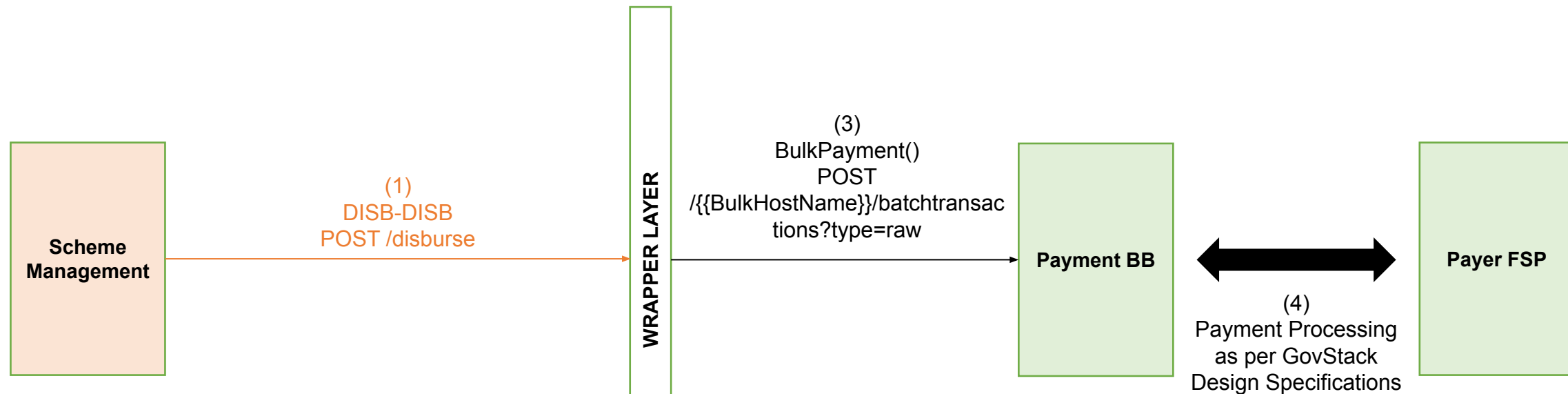
(1) ASYNC CALL TO INITIATE DISBURSEMENT

G2PConnect Interface Type: Async

G2PConnect Endpoint: POST /disburse

GovStack Interface Type: Async

GovStack Endpoint: POST/{{BulkHostName}}/batchtransactions?type=raw



Note: Our design does not still cater to direct invocation of a Payment Switch and Settlement system, so if the Pay-BB were to be used, it still wouldn't integrate directly to pass Post/Disburse instructions ahead to a Switch.

Questions:

- Would G2P Connect aligned Scheme Management be open to directly calling Voucher APIs in the GovStack specification based Payment Building Block? Basically meaning that Scheme management works directly with GovStack specification PBB for just vouchers while choosing not to implement bulk disbursements or other areas.

(2) Store the following attributes that don't map in a database to prevent total loss of information for later retrieval

- scheduled_timestamp
- payer_name
- payee_name
- note
- instruction
- locale

Wrapper Database
Additional Data Storage and Retrieval

(2) ASYNC TO RECEIVE STATUS OF DISBURSEMENT AGAINST REQUEST RAISED IN (1)

G2PConnect Interface Type: Async

G2PConnect Endpoint: POST /on-disburse

GovStack Interface Type: Async

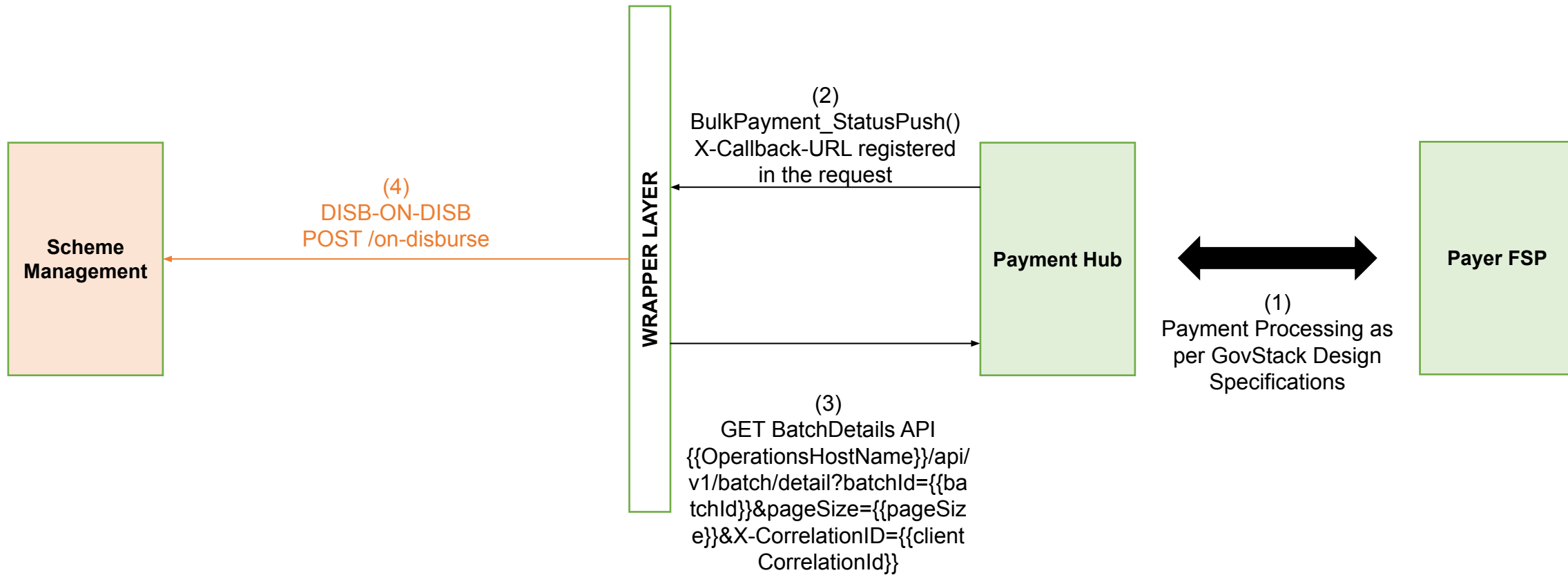
GovStack Endpoint: 2 Endpoints to be orchestrated to prepare response as per G2PConnect POST /on-disburse Expectations.

BulkPayment_StatusPush() to provide Batch Summary details.

Case 1 : As a call back of Batch Payment.

X-Callback-URL registered in the request

GET BatchDetails API to fetch transaction level status of each batch and append that in the response back to on-disburse



POST/on-disburse expects both a Summary and every single transaction detail to be passed in the message body. In our design this is catered through 2 separate APIs.

1. POST/ BulkPayment_StatusPush() which provides summary details.
2. GET/ BatchDetails API which provides individual instruction details.

In this case, the wrapper will need to orchestrate flow and when it receives Status Push, Get Batch Details to collate together a comprehensive report for the DISB-ON-DISB API call.

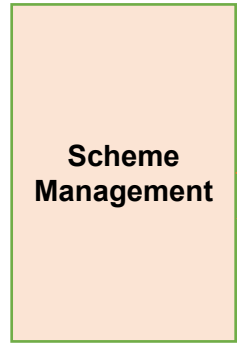
(3) SYNC INTERFACE TO INITIATE A DISBURSEMENT

G2PConnect Interface Type: Sync

G2PConnect Endpoint: /disburse/sync/disburse

GovStack Interface Type: Async

GovStack Endpoint: POST/{{BulkHostName}}/batchtransactions?type=raw

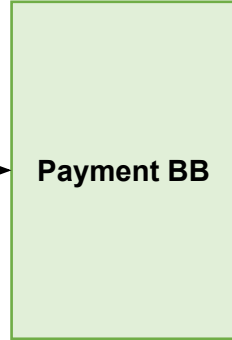


(1) /disburse/sync/disburse REQUEST

Immediate Response putting all Transactions in "RCVD" state



(4)
BulkPayment()
POST
{BulkHostName}/batchtransactions?type=raw



(4)
Payment Processing
as per GovStack
Design Specifications



G2PCONNECT FLOW

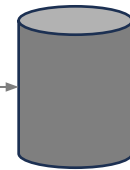
GovStack/G2PConnect FLOW

1. Disburse/sync API is called and creates a blocking call where the response against each instruction is as follows:
 - a. Received OR
 - b. Pending OR
 - c. Accepted OR
 - d. Rejected.
2. Disburse/sync/txn/status is then used to get updated statuses for individual transactions by polling of the status end point whenever needed.

1. Disburse/sync API is called but instead of creating a constant blocking call the Wrapper Layer responds back with "Received" against all instructions.
2. The wrapper layer then maintains state for all transactions in its database, and invokes the BulkPayment() Async call. From here onwards the flow will proceed as standard GovStack specs define.
3. Scheme management may invoke disburse/sync/txn/status to get updates status from the wrapper layer.
4. The wrapper layer will update status of individual txns as it gets updates through GovStack Specs update interfaces.

(2) Store the following attributes that don't map in a database to prevent total loss of information for later retrieval

- scheduled_timestamp
- payer_name
- payee_name
- note
- instruction
- locale



Wrapper Database

Additional Data Storage and Retrieval

Additionally, when Sync-Async conversion is being done, the Wrapper Database will be a full-fledged transaction database as well which will retain individual transactions and their posting status and act as a source of truth for G2PConnect Compliant Scheme Management.

(4 & 5) ASYNC INTERFACES TO GET UPDATED STATUS OF ONE OR MORE DISBURSEMENT REQUESTS

G2PConnect Interface Type: Async

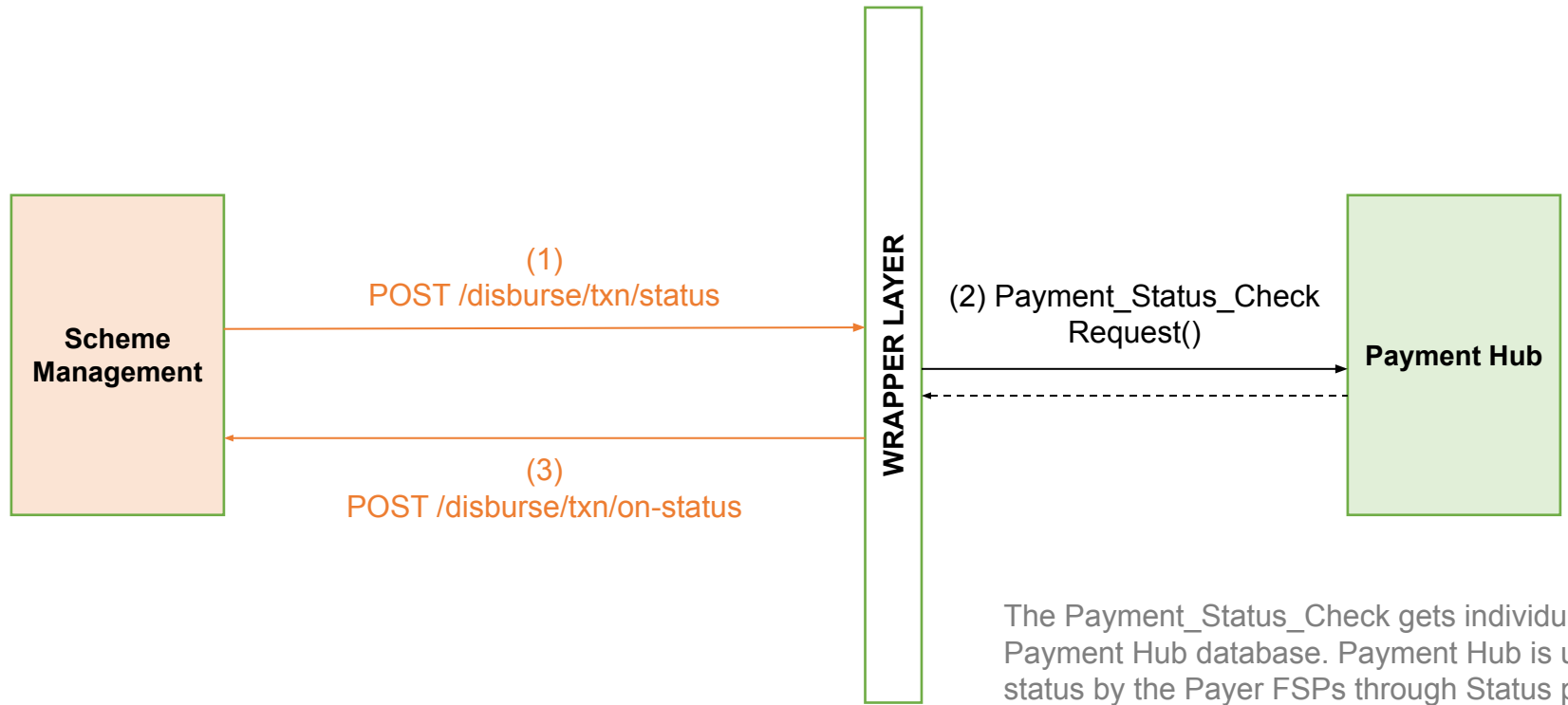
G2PConnect Endpoint: POST /disburse/txn/status

G2PConnect Interface Type: Async

G2PConnect Endpoint: POST /disburse/txn/on-status

GovStack Interface Type: Sync

GovStack Endpoint: Payment_Status_Check Request()



The Payment_Status_Check gets individual transaction status updates from the Payment Hub database. Payment Hub is updated on each individual transaction's status by the Payer FSPs through Status push or a similar Status Check interface.

However, for G2PConnect Compliant Scheme Management, they would get updated status updates of each transaction by calling (1) and getting an acknowledgement, after fetching of result using the sync interface in (2) the Wrapper Layer would invoke the api in (3) to return the response to Scheme Management.

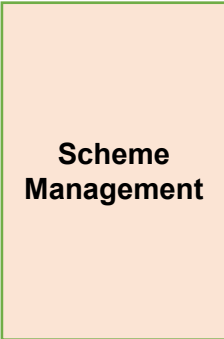
(7) SYNC INTERFACE TO GET UPDATED STATUS OF ONE OR MORE DISBURSEMENT REQUESTS

G2PConnect Interface Type: Sync

G2PConnect Endpoint: /disburse/sync/txn/status

GovStack Interface Type: Sync

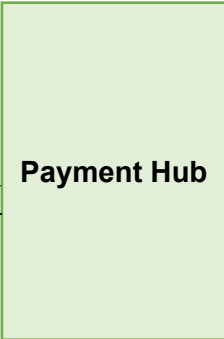
GovStack Endpoint: Payment_Status_Check Request()



(1)
/disburse/sync/txn/status REQUEST



(2) Payment_Status_Check
Request()



(3)
/disburse/sync/txn/status RESPONSE

