



openFinance API Framework Extended Services Verification of Party (VOP)

Version 1.0

24 January 2025

License Notice

This Specification has been prepared by the Participants of the openFinance Taskforce¹. This Specification is published by the Berlin Group under the following license conditions:

- "Creative Commons Attribution-NoDerivatives 4.0 International Public License"



This means that the Specification can be copied and redistributed in any medium or format for any purpose, even commercially, and when shared, that appropriate credit must be given, a link to the license must be provided, and indicated if changes were made. You may do so in any reasonable manner, but not in any way that suggests the licensor endorses you or your use. In addition, if you remix, transform, or build upon the Specification, you may not distribute the modified Specification.

- Implementation of certain elements of this Specification may require licenses under third party intellectual property rights, including without limitation, patent rights. The Berlin Group or any contributor to the Specification is not, and shall not be held responsible in any manner for identifying or failing to identify any or all such third party intellectual property rights.
- Any right, title and interest in and to the copyright and all related rights in topic-related Scheme Rulebooks, belong to the respective Scheme Manager (amongst others, the European Payments Council AISBL – EPC).
- The Specification, including technical data, may be subject to export or import regulations in different countries. Any user of the Specification agrees to comply strictly with all such regulations and acknowledges that it has the responsibility to obtain licenses to export, re-export, or import (parts of) the Specification.

¹ The openFinance Taskforce brings together participants of the Berlin Group with additional European banks (ASPSPs), banking associations, payment associations, payment schemes and interbank processors.

Contents

Contents.....	ii
1 Introduction.....	1
1.1 From Core XS2A Interface to openFinance API	1
1.2 Verification of Payee Service as one single request use case of party verification	2
1.3 Document Structure.....	3
1.4 Document History	4
2 Business Processes and Operational Rules	5
2.1 Business Process.....	5
2.1.1 Dedicated VOP API.....	5
2.1.2 Integrated VOP Function for Single Payments	6
2.1.3 Dedicated VOP API for batch processing	7
2.1.4 Integrated VOP Function for Bulk Payments	9
2.2 Operational Rules.....	10
2.2.1 API Client and Corporate Authentication for Corporate Direct Access	10
2.2.2 Service Types	10
2.2.3 Consent model for dedicated VOP API	11
3 Dedicated VOP API Definition	12
3.1 Service Onboarding.....	12
3.2 Discovery Services	12
3.3 API Access Methods	12
3.4 Detailed API Service Definitions	14
3.4.1 Verification of Party Request.....	14
3.4.2 Bulk Verification of Party Request	19
3.4.3 Retrieve Bulk Verification of Party Request.....	22
3.4.4 Get Status of Bulk Verification of Party Request	25
4 Integration in Payment APIs	28
4.1 Changes in Payment Request Messages	28
4.2 Changes in Payment Response Message	29
4.3 Changes in (Extended) Status Request.....	32

4.4	New Method for Confirmation of Creditor Name and Payment	34
5	References	36
5.1	Normative References	36
5.2	Informative References	36

1 Introduction

1.1 From Core XS2A Interface to openFinance API

With [PSD2] the European Union has published a directive on payment services in the internal market. Among others [PSD2] contains regulations on services to be operated by so called Third Party Payment Service Providers (TPP) on behalf of a Payment Service User (PSU). These services are

- Payment Initiation Service (PIS) to be operated by a Payment Initiation Service Provider (PISP) TPP as defined by article 66 of [PSD2],
- Account Information Service (AIS) to be operated by an Account Information Service Provider (AISP) TPP as defined by article 67 of [PSD2], and
- Confirmation on the Availability of Funds Service (FCS) to be used by a Payment Instrument Issuing Service Provider (PIISP) TPP as defined by article 65 of [PSD2].

To implement these services (subject to PSU consent) a TPP needs to access the account of the PSU. The account is managed by another PSP called the Account Servicing Payment Service Provider (ASPSP). To support the TPP in accessing the accounts managed by an ASPSP, each ASPSP has to provide an "access to account interface" (XS2A interface). Such an interface has been defined in the Berlin Group NextGenPSD2 XS2A Framework.

This XS2A Framework is now extended to premium services. This interface is addressed in the following as **openFinance API**. This openFinance API differs from the XS2A interface in several dimensions:

- The extended services might not rely anymore solely on PSD2.
- Other important regulatory frameworks which apply are e.g. GDPR.
- The openFinance API can address different types of **API Clients** as access clients, e.g. TPPs regulated by an NCA according to PSD2, or corporates not regulated by an NCA.
- The extended services might require contracts between the access client and the ASPSP.
- While the client identification at the openFinance API can still be based on eIDAS certificates, they do not need to be necessarily PSD2 compliant eIDAS certificates.
- The extended services might require e.g. the direct involvement of the access client's bank for KYC processes.

Note: The notions of API Client and ASPSP are used because of the technical standardisation perspective of the openFinance API. These terms are analogous to "asset broker" and "asset holder" resp. in the work of the ERPB on a SEPA API access scheme.

Note: In implementations, the API services of several ASPSPs might be provided on an aggregation platform. Such platforms will be addressed in the openFinance API Framework as "API provider".

The following account access methods are covered by this framework:

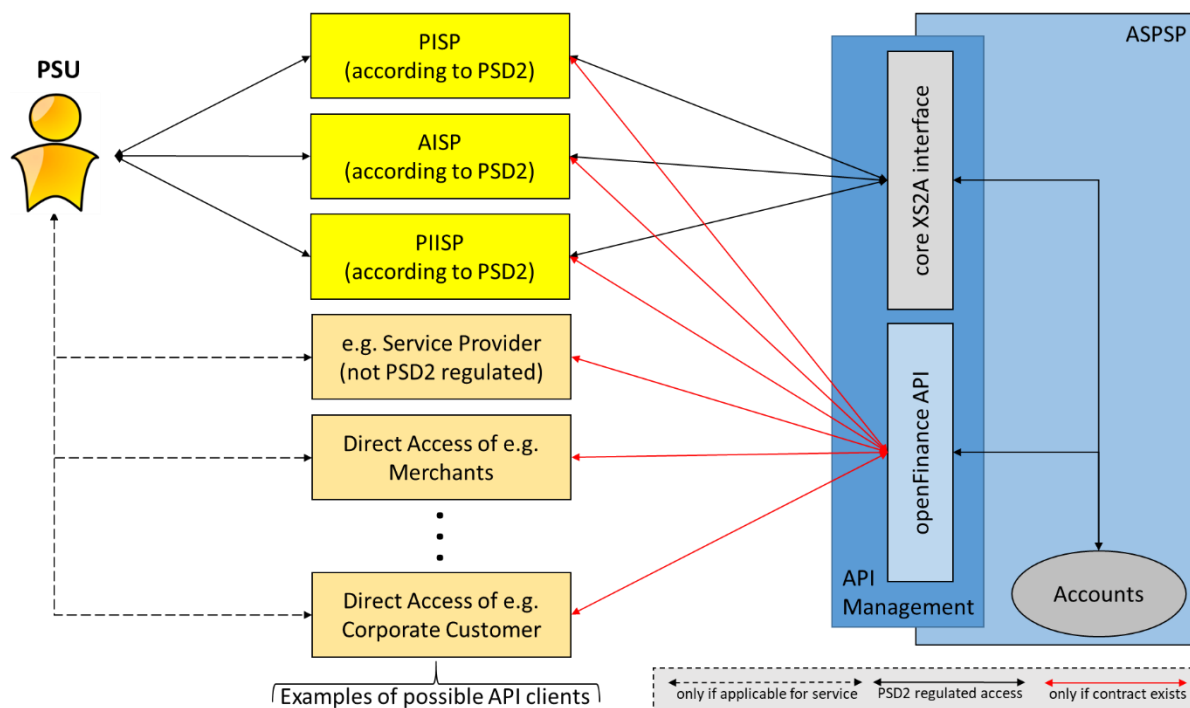


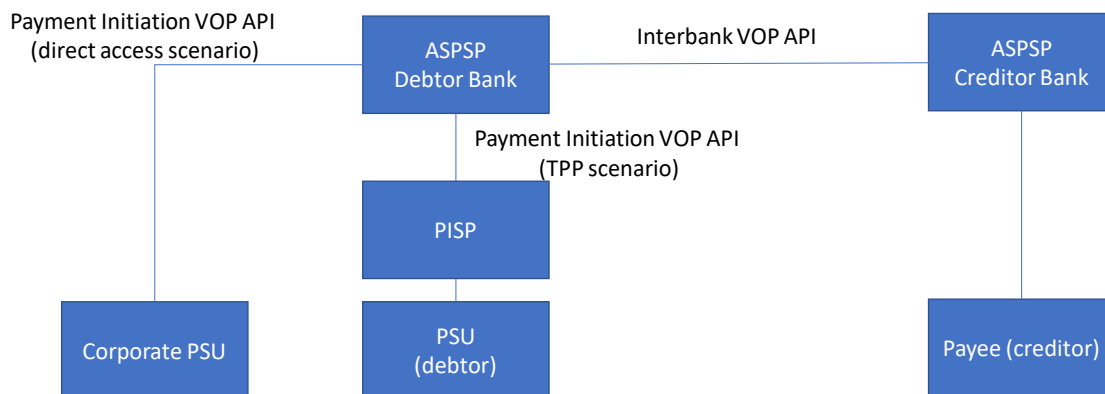
Figure 1: Core XS2A interface and openFinance API

The ASPSP may restrict the access to the services offered at its openFinance API and require dedicated onboarding. The requirements for the rights to access to services offered at the openFinance API are out of scope of this document. These requirements will be described in a dedicated operational rules document [oFA-IG-ADM].

1.2 Verification of Payee Service as one single request use case of party verification

The agreed regulation on instant payments requires a verification of payee function under certain circumstances. The regulation addresses potentially all European credit transfers (not only SEPA variants).

The regulation addresses the interbank sphere and potentially also the initiation sphere. For this reason, it can be assumed in the following, that the related interface will be realised as an API. The following picture provides an overview on the situation.



This document introduces more generally a "verification of party" function (VOP), since the verification function could also be applied to direct debit services in future or more generally to counterparties as such. This document describes a Payment Initiation VOP API as an **extended service** within the NextGenPSD2 API V1.3.x as well as the V2 Compliance API defined in the openFinance API Framework. It makes use of data models, party authentication and error handling as provided by the openFinance API Framework, cp. Section 5.1. In the description, there are sometimes restrictions to the case of a "verification of payee" services, where appropriate.

Note: The openFinance API Framework is extended in 2025 to not only address TPP based scenarios, but also **direct access for corporates**. Thus, the Payment Initiation VOP API defined in this document will also be available in a direct access scenario for corporate customers as well as public sector customers, see the picture above. This is not further noted explicitly anymore in the text: In all diagrams "API Client" can be read as either "TPP" or "Corporate PSU".

1.3 Document Structure

This document first introduces business processes and operational rules for the VOP API in Section 2. Technical details are then specified in Section 3 resp. Section 4.

1.4 Document History

Version	Date	Changes
Version 0.5	2024-07-23	Draft for public market consultation
Version 1.0	2025-01-24	Version 1.0 after having included market and further internal feedback.

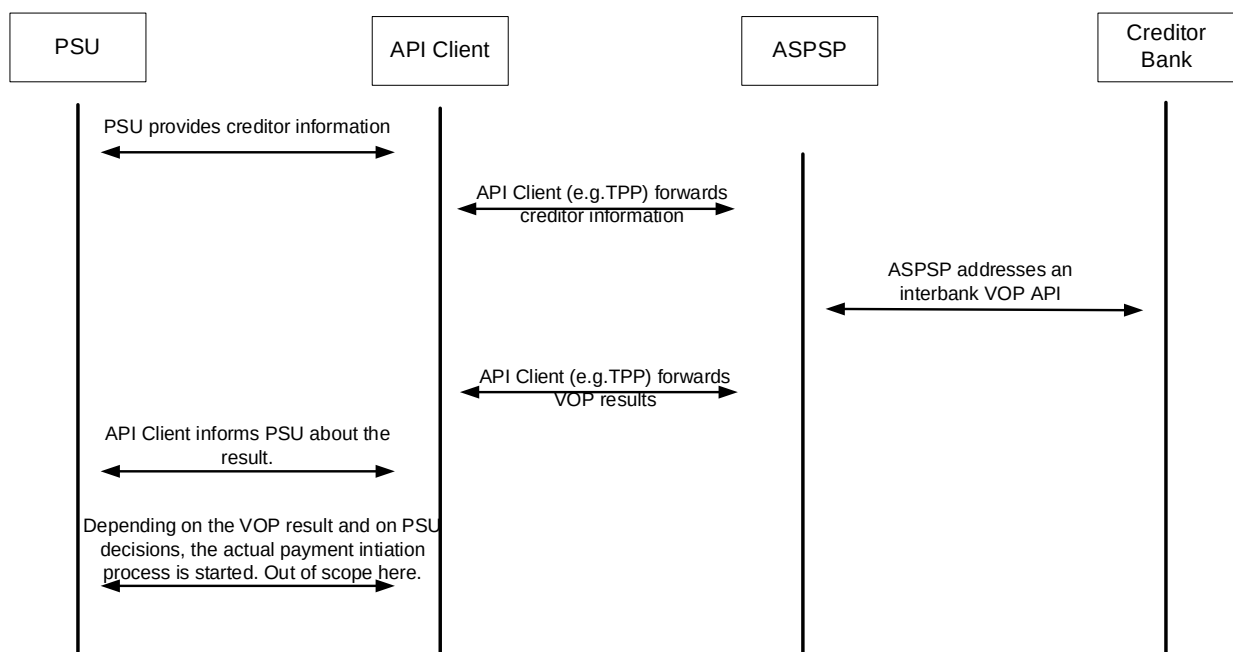
2 Business Processes and Operational Rules

2.1 Business Process

The openFinance API Framework will offer two variants for verification of payees services in the payment initiation APIs.

2.1.1 Dedicated VOP API

The following business process is supported for VOP Services within the openFinance API Framework for a dedicated VOP API:



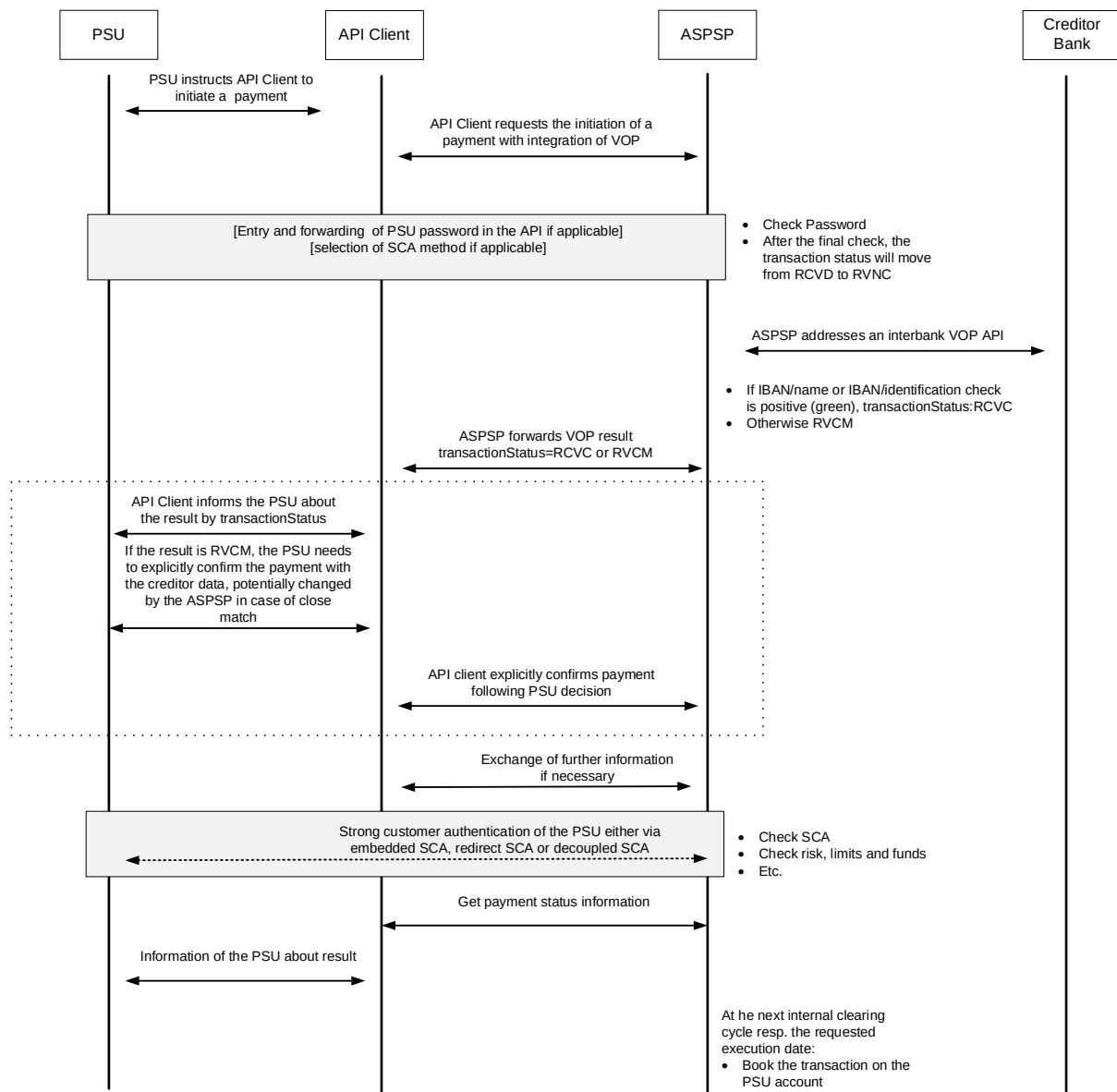
Remark: The scenario, where the TPP as API Client is providing the creditor information on his/her own, is not explicitly covered since it is a payment initiation within a PSP merchant solution. In this case, the creditor is a customer of the TPP and the TPP knows the account owner name already. Nevertheless, also in this case, the VOP API could be addressed by the TPP if needed.

Remark: In the case, the API Client is a corporate or a public sector institution, the PSU is an employee in the diagram of logical roles above.

Note: This document is addressing the VOP API in the initiation sphere between TPP/Corporate Customer and ASPSP, here "Debtor Bank". The interbank VOP API above is only added for clarity of the overall process. This API is not addressed in this document.

2.1.2 Integrated VOP Function for Single Payments

The following business process for verification of payee is supported as an integrated open-Finance function within Payment Initiation Services for single payments:

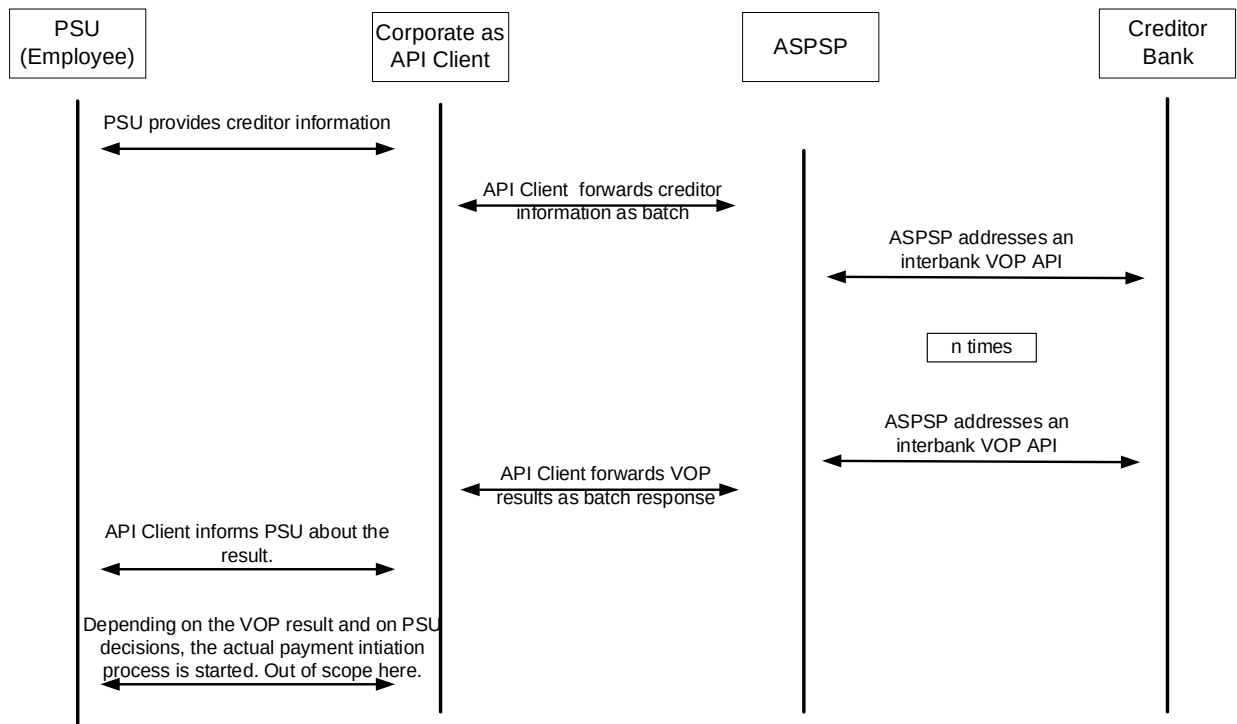


Note: The actions in the dotted box are not needed, if the VOP result is positive – in this case the payment processing can be handled as in the current definition of the payment services.

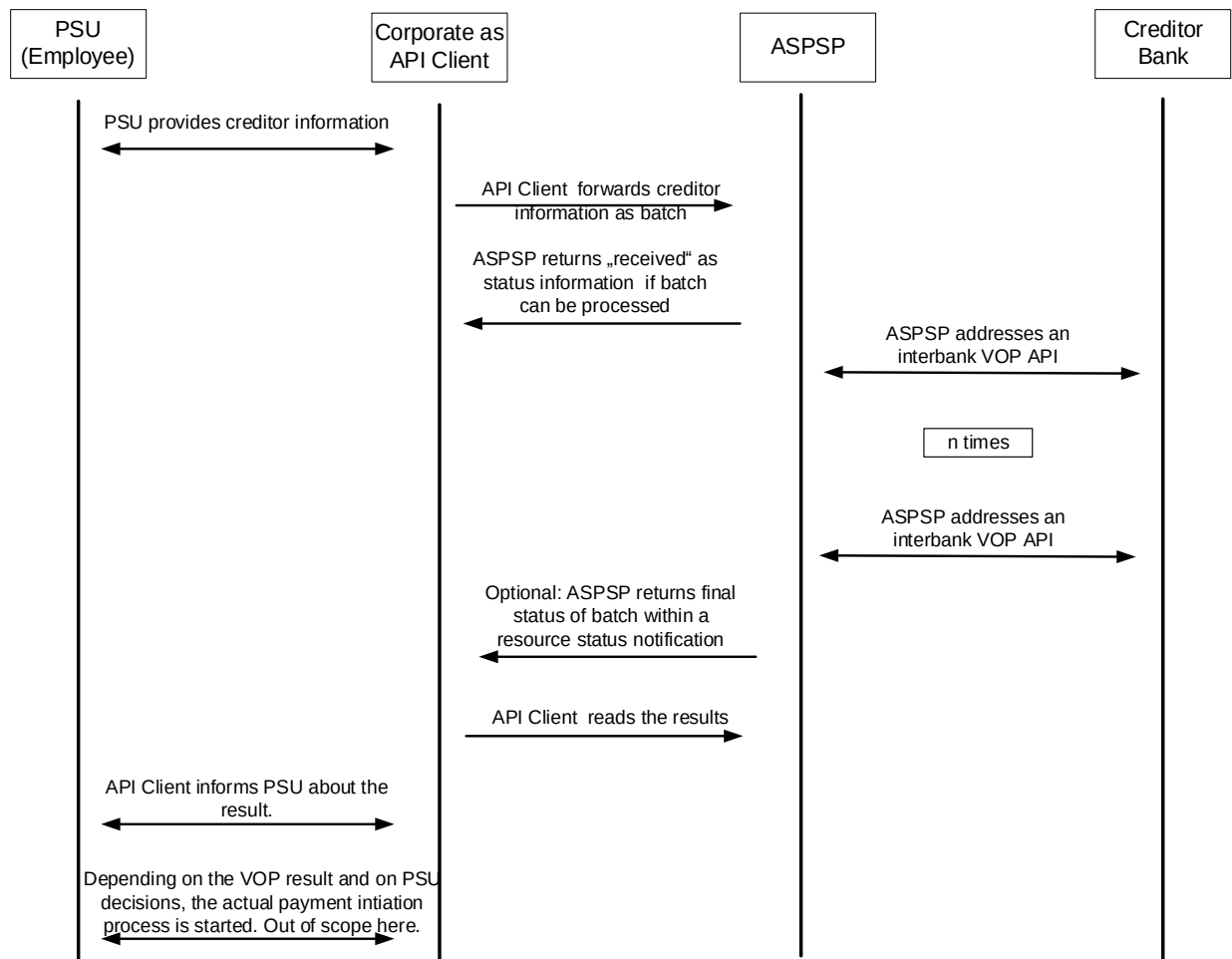
Note: The API Client might initiate the payment initiation with a flag indicating that VOP is not needed (e.g. opt out or account owner known since contract partner). In this case, there is no change/addition to the current payment initiation flows.

2.1.3 Dedicated VOP API for batch processing

For public sector and corporate customers, it might be needed for ASPSP in addition to offer a batch based VOP API. In such cases, the ASPSP might process the related bulk still in a synchronous mode, if there are only a few entries. In this case, the following flow would apply:



If there are many entries, it seems more appropriate to offer an asynchronous process, since the inter-bank VOP service usage might take some time if the related bulk is huge. Thus, the following flow would then apply:

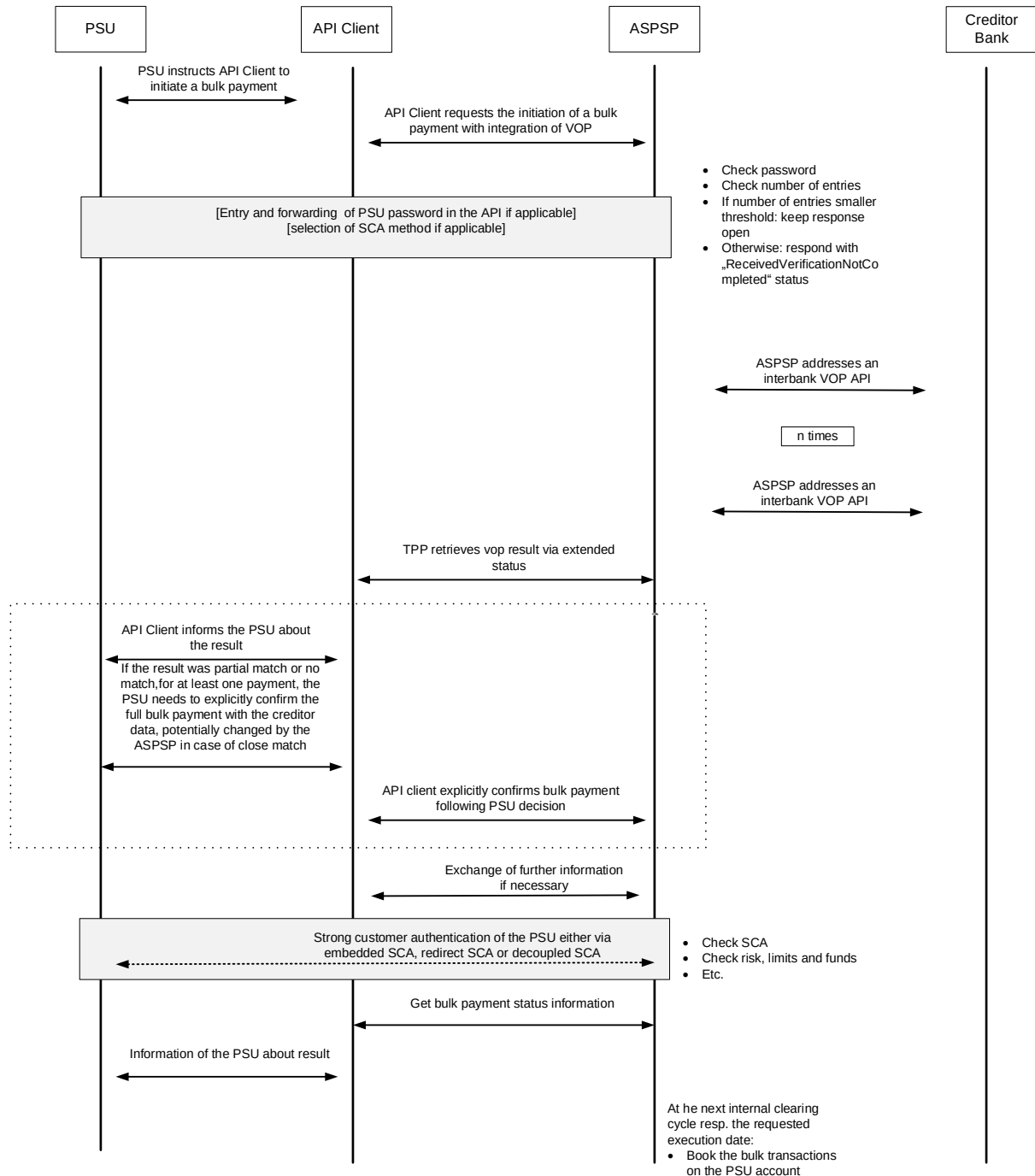


NOTE: It is up to the ASPSP decision/implementation where the threshold of acting asynchronously is. It could also be an ASPSP decision to always provide an asynchronous process for bulk processing.

NOTE: In the above process, the API Client can always read the VOP bulk resource as soon as it has been created and the API Client has received the related resource identification. The content of the resource is then all requests and all the responses already received by the ASPSP from the related creditor banks up to the current point in time. That means that responses can be retrieved only partially if the final status of the VOP bulk resource has not been reached yet.

2.1.4 Integrated VOP Function for Bulk Payments

The following business process for verification of payee is supported as an integrated open-Finance function within Payment Initiation Services for bulk payments:



Note: The actions in the dotted box are not needed, if the VOP result is positive for all bulk entries – in this case the payment processing can be handled as in the current definition of the payment services.

Note: The API Client might initiate the payment initiation with a flag indicating that VOP is not needed (e.g. opt out or account owner known since contract partner). In this case, there is no change/addition to the current payment initiation flows, if the ASPSP is recognising this attribute.

2.2 Operational Rules

2.2.1 API Client and Corporate Authentication for Corporate Direct Access

If the VOP API is used in a Corporate Direct Access model, the following requirements apply on the certificate usage:

- The TLS web site certificate of the API Client is either a certificate of the corporate or of an ERP cloud provider. This certificate authenticates the API Client.
- If the ASPSP mandates message signing, the sealing certificate needs to be owned by the corporate to provide corporate authentication.

As a minimum, the ASPSP shall accept QWACs or QSEALs certificates as defined by [eIDAS]. The ASPSP might accept other certificates in bilateral agreements with the corporate.

2.2.2 Service Types

The following service types are supported for dedicated VOP services:

Service	Service Type	Technical Functionalities	PSU directly involved
Single VOP request	VOP-SNG	Initiation of a related synchronous VOP API request in the inter-bank sphere	No
VOP bulk request	VOP-BLK	Initiation of related VOP API requests in the inter-bank sphere.	No
		Retrieval of related VOP API responses if not provided synchronously	No

2.2.3 Consent model for dedicated VOP API

The consent of the corporate is granted via corporate authentication on API level:

API Service	How the corporate grants consent	How consent of the corporate is verified
Single VOP request	The consent could be granted by applying corporate authentication.	By checking that the corporate authentication is valid.
VOP bulk request	The consent could be granted by applying corporate authentication.	By checking that the corporate authentication is valid.

3 Dedicated VOP API Definition

The technical definitions below make use of the data dictionary of the openFinance API Framework [oFA-DaD]. API Client authentication is handled as defined in [oFA-PFSM]. The service definition below is only taking care of requests which are responded positively with http response code 200 or 202. Processing errors are handled as defined in the openFinance API Framework definitions [oFA-PFSM].

3.1 Service Onboarding

The VOP Service is intended to be the first business service in an openFinance corporate direct access API. This API will support a service onboarding process. The actual service onboarding by the corporate customer will be performed via the onboarding API defined in [oFA-IG-ADM].

Note: Please note that a future version of the document [oFA-IG-ADM] will come with an endpoint for dedicated QSEAL onboarding, to avoid to transport and check the QSEAL in the business services all the time.

3.2 Discovery Services

The document [oFA-Disc] contains implementation details for VOP services and potential integration into payment initiation services.

3.3 API Access Methods

The following tables gives an overview on the HTTP access methods supported by the API endpoints and by resources created through this API.

Conditions in the following tables

It is further defined, whether this method support is mandated for the ASPSP by this specification or whether it is an optional feature for the ASPSP. Please note that this condition is given relative to the parent node of the path, i.e. the condition e.g. on a method on `/v2/payments/{payment-product}/{paymentId}` applies only if the endpoint `/v2/payments/{payment-product}` is supported at all.

Endpoints/Resources	Method	Condition	Description
<code>/v2/party-verification-endpoint</code>	POST	Mandatory	Initiates a verification of party request of type • payee-verifications, • payer-verifications or more generally • party-verifications. Section 3.4.1
<code>/v2/bulk-party-verification-endpoint</code>	POST	Optional	Initiates a verification of party request of type • bulk-payee-verifications, • bulk-payer-verifications or more generally • bulk-party-verifications for all entries of the related bulk. If the related requests cannot all be handled synchronously, the ASPSP will create a resource with identification <code>vopBulkId</code>. Section 3.4.2
<code>/v2/bulk-party-verification-endpoint/{vopBulkId}</code>	GET	Conditional	Read the details of vop bulk resource, if a related resource has been created. Section 3.4.3
<code>/v2/bulk-party-verification-endpoint/{vopBulkId}/status</code>	GET	Conditional	Read the vop bulk status of the vop bulk resource, if a related resource has been created.

Endpoints/Resources	Method	Condition	Description
			Section 3.4.4

3.4 Detailed API Service Definitions

3.4.1 Verification of Party Request

Call

POST /v1/{party-verification-endpoint}

Submits a Verification of Party Request.

NOTE: In contrast to the integrated version below, the dedicated VOP API can be set up more general by offering not only verification of payee service, but more generally verification of party services. For the requirement of the instant payment regulation, it is sufficient to make use of the verification of payee definitions as a subset. Thus, it is sufficient to support the /payee-verifications endpoint to fulfill the instant payment regulation. Nevertheless, the more generic definition was chosen to unify the data models for these different potential services.

NOTE: Different VOP API Access Schemes might come with different rules, e.g. whether a party name is mandated in the request or not. The implementation variant of the addressed VOP API following the openFinance API Work will be described in a related extension to the Discovery API, which will be covered in [oFA-Disc] in future also for VOP services.

Path Parameters

Parameter	Description
party-verification-endpoint	The following values are supported in this specification: - payee-verifications: A service to fulfil related verifications for payees. - payer-verifications: A service to fulfil related verifications for payers. - party-verifications: A general service independent of the underlying payment scheme just addressing a counterparty.

Note: This specification offers different endpoints for the usage of a party verification in different contexts. Dedicated endpoints for dedicated verifications (e.g. payee) are offered, since the access rights to the related services could differ e.g. to a generic party verification service. The data models for these potentially slightly different services are the same.

Note: The next version of this document will also cover an "account-verifications" endpoint, where the attribute "party" will not be applicable in the request body, see below. This request will come with an "accountMatch" attribute only. This new function needs still to be defined in detail by the technical committees of openFinance.

Note: The ASPSP might also offer a v1/pain.001-payee-verifications endpoint for the same need of a verification of payee request. Please note that this endpoint expects a pain.001 XML based body. The response will be a pain.002 response. This is not further specified here.

Query Parameters

No query parameter

Header Parameters

Attribute	Type	Condition	Description
X-Request-ID	UUID	Mandatory	ID of the request, unique to the call, as determined by the initiating party.
Accept-language	String	Optional	The API Client may ask for a language for free text fields in the related response (e.g. in additionalInformation). This header might be ignored by the ASPSP, if the addressed language is not supported. If not supported the default language of the ASPSP will be used.
API-Contract-ID	UUID	Conditional	ID of the underlying service contract between API Client and API Server, if provided by the ASPSP in the related onboarding process as defined in [oFA-IG-ADM].

Remark: A detailed time stamp (X-Request-Timestamp/X-Request-Timestamp) is not added in the TPP / Payer ASPSP interface, since time definitions in VOP schemes are counted between Payer ASPSP and Payee ASPSP in case where the Payer ASPSP is the submitter of the VOP request in the related scheme.

Request Body

Tag	Type	Usage	Description
partyAccount	Account Reference	Mandatory	The data type might be restricted by the ASPSP to e.g. "iban" use only,

Tag	Type	Usage	Description
			which would be Account Reference2 as data type. NOTE; If IBAN or BBAN is not sufficient to identify the related account, e.g. an electronic money account, then the remittance information should be used in addition to fully identify the account.
partyAgent	Agent Description	Optional	
party	Party Description4	Mandatory	This structure shall contain the party name and/or a party identifier, Related API Access Schemes might define further requirements/restrictions, e.g. • Only support the usage of either name or identification. • Or only support organisation identification.
unstructuredRemittanceInformation	Array of Max140Text	Optional	This attribute may contain data to refer to electronic money accounts.
requestingAgent	Agent Description	n.a.	Only applicable in the inter-bank/inter-PSP sphere.

Response Code

The HTTP response code equals 200.

Response Header

Attribute	Type	Condition	Description
X-Request-ID	UUID	Mandatory	ID of the request, unique to the call, as determined by the initiating party.

Response Body

Attribute	Type	Condition	Description
partyNameMatch	Party Name Match Code	Conditional	Only provided, if a party name has been provided.
partyIdMatch	Party Identification Match Code	Conditional	Only provided, if a party identification has been provided.
matchedName	Max140Text	Conditional	This attribute might be provided in case of a near match and contains the name as provided by the addressed party agent.
additionalInformation	Max500Text	Optional	Should be human readable text.

NOTE: Additional information might be structured in future. In this case, more optional attributes will be added to the response.

Example with Name Match

Request

```
POST https://www.testbank.com/openFinance/v1/payee-verifications/
Content-Type:      application/json
X-Request-ID:      99391c7e-ad88-49ec-a2ad-99ddcb1f7723
Date:              Sun, 27 Feb 2018 18:02:37 GMT
```

```
{
  "partyAccount": {"iban": "DE02100100109307118603"},
  "party": {
    "name": "Peter Miller Ltd"
  }
}
```

Response

HTTP/1.x 200
X-Request-ID: 99391c7e-ad88-49ec-a2ad-99ddcb1f7723
Date: Sun, 27 Feb 2018 18:02:39 GMT
Content-Type: application/json

```
{  
  "partyNameMatch": "MTCH"  
}
```

Example with Identification Match

Request

POST https://www.testbank.com/openFinance/v1/payee-verifications/
Content-Type: application/json
X-Request-ID: 99391c7e-ad88-49ec-a2ad-99ddcb1f7723
Date: Sun, 27 Feb 2018 18:02:37 GMT

```
{  
  "partyAccount": {"iban": "DE02100100109307118603"},  
  "party": {  
    "identification": {  
      "organisationId": {"lei": "PetMillTD"}}  
  }  
}
```

Response

HTTP/1.x 200
X-Request-ID: 99391c7e-ad88-49ec-a2ad-99ddcb1f7723
Date: Sun, 27 Feb 2018 18:02:39 GMT
Content-Type: application/json

```
{  
  "partyIdMatch": "MTCH"  
}
```

3.4.2 Bulk Verification of Party Request

Call

POST /v1/{bulk-party-verification-endpoint}

Submits a bulk of related party verification requests.

Path Parameters

Parameter	Description
bulk-party-verification-endpoint	The following values are supported in this specification: • bulk-payee-verifications: A service to fulfil related verifications for a bulk of payees. • bulk-payer-verifications: A service to fulfil related verifications for a bulk of payers. • bulk-party-verifications: A general service independent of the underlying payment scheme just addressing a bulk of counterparties.

Query Parameters

No query parameter

Header Parameters

Attribute	Type	Condition	Description
X-Request-ID	UUID	Mandatory	ID of the request, unique to the call, as determined by the initiating party.
Client-Asynchronous-Process-Preferred	Boolean	Optional	true: Asynchronous process preferred. false: Synchronous process preferred. The ASPSP may ignore this flag.
API-Contract-ID	UUID	Conditional	ID of the underlying service contract between API Client and API Server, if provided by the ASPSP in the related onboarding process as defined in [oFA-IG-ADM].

Request Body

Tag	Type	Usage	Description
partyVerificationRequests	Array of Party Verification Request	Mandatory	A non-empty bulk of related party verification entries.
requestingAgent	Agent Description	n.a.	Only applicable in the inter-bank/inter-PSP sphere.

Response Code

The HTTP response code equals 200 in case of all results are available.

The HTTP response code equals 202 in case processing is not completed.

Response Header

Attribute	Type	Condition	Description
X-Request-ID	UUID	Mandatory	ID of the request, unique to the call, as determined by the initiating party.

Response Body

Attribute	Type	Condition	Description
vopBulkId	UUID	Conditional	Only in case of http response code 202: Technical resource identification of the VOP bulk.
vopBulkStatus	VOP Bulk-Status	Mandatory	
partyVerificationResponses	Array of Party Verification Response	Conditional	If verification results (even partially) are available, then this attribute shall be provided. The content is a non-empty array.

Attribute	Type	Condition	Description
_links	Links	Conditional	Only in case of http response code 202. It will provide at least a hyperlink of href type "status".

Example

Request

POST https://www.testbank.com/openFinance/v1/bulk-payee-verifications/
 Content-Type: application/json
 X-Request-ID: 99391c7e-ad88-49ec-a2ad-99ddcb1f7723
 Date: Sun, 27 Feb 2018 18:02:37 GMT

```
{
  "partyVerificationRequests": [
    {
      "uetr" : 98721c7e-ad88-49ec-a2ad-99ddcb1f1234,
      "partyAccount": {"iban": "DE02100100109307118603"},
      "party": {
        "name": "Peter Miller Ltd"
      }
    },
    {
      "uetr" : 98721c7e-ad88-49ec-a2ad-99ddcb1f2345
      "partyAccount": {"iban": "DE02100100109307118612"},
      "party": {
        "identification": {"organisationId": {"lei": "PetMullTD"}}
      }
    }
  ]
}
```

Response in case of results are available:

HTTP/1.x 200
 X-Request-ID: 99391c7e-ad88-49ec-a2ad-99ddcb1f7723
 Date: Sun, 27 Feb 2018 18:02:39 GMT
 Content-Type: application/json

```
{
  "vopBulkStatus" : "completed",
  "partyVerificationResponses" : [
    {
      "uetr" : 98721c7e-ad88-49ec-a2ad-99ddcb1f1234,
```

```
    "partyNameMatch": "MTCH"
  },
  {
    "uetr" : 98721c7e-ad88-49ec-a2ad-99ddcb1f2345,
    "partyIdMatch": "MTCH"
  }
]
```

Response in case processing of batch processing at once decided:

HTTP/1.x 202

X-Request-ID: 99391c7e-ad88-49ec-a2ad-99ddcb1f7723

Date: Sun, 27 Feb 2018 18:02:39 GMT

Content-Type: application/json

```
{ "vopBulkId": "dddd1c7e-ad88-49ec-a2ad-99ddcb1f7723",
  "vopBulkStatus": "received",
  "_links": {"status":
    { "href": "www.testbank.com/openFinance/v1/bulk-payee-
    verifications/dddd1c7e-ad88-49ec-a2ad-99ddcb1f7723/status"},
    "self": {" href": "www.testbank.com/openFinance/v1/bulk-payee-
    verifications/dddd1c7e-ad88-49ec-a2ad-99ddcb1f7723"}}
}
```

3.4.3 Retrieve Bulk Verification of Party Request

Call

GET /v1/{bulk-party-verification-endpoint}/{vopBulkId}

Retrieves the detailed status of a bulk of related party verification requests.

Path Parameters

Parameter	Description
bulk-party-verification-endpoint	The following values are supported in this specification: • bulk-payee-verifications: A service to fulfil related verifications for a bulk of payees. • bulk-payer-verifications: A service to fulfil related verifications for a bulk of payers. • bulk-party-verifications: A general service independent of the underlying payment scheme just addressing a bulk of counterparties.

Query Parameters

No query parameter

Header Parameters

Attribute	Type	Condition	Description
X-Request-ID	UUID	Mandatory	ID of the request, unique to the call, as determined by the initiating party.
API-Contract-ID	UUID	Conditional	ID of the underlying service contract between API Client and API Server, if provided by the ASPSP in the related onboarding process as defined in [oFA-IG-ADM].

Request Body

n.a.

Response Code

The HTTP response code equals 200.

Response Header

Attribute	Type	Condition	Description
X-Request-ID	UUID	Mandatory	ID of the request, unique to the call, as determined by the initiating party.

Response Body

Attribute	Type	Condition	Description
vopBulkId	UUID	Mandatory	
vopBulkStatus	VOP Bulk Status	Mandatory	
partyVerificationRequestsAndResults	Array of Party Verification Request and Response	Mandatory	A non-empty bulk of related party verification requests and related responses, if available.

Example

```
GET https://www.testbank.com/openFinance/v1/bulk-payee-verifications/ddd1c7e-ad88-49ec-a2ad-99ddcb1f7723
Content-Type: application/json
X-Request-ID: 99391c7e-ad88-49ec-a2ad-99ddcb1f7726
Date: Sun, 27 Feb 2018 18:02:41 GMT
```

Response

```
HTTP/1.x 200
X-Request-ID: 99391c7e-ad88-49ec-a2ad-99ddcb1f7726
Date: Sun, 27 Feb 2018 18:02:43 GMT
Content-Type: application/json
```

```
{  "vopBulkId": "ddd1c7e-ad88-49ec-a2ad-99ddcb1f7723",
    "vopBulkStatus": "completed",
    "partyVerificationRequestsAndResults": [
      {
        "uetr" : 98721c7e-ad88-49ec-a2ad-99ddcb1f1234,
        "partyAccount": {"iban": "DE02100100109307118603"},
      }
    ]
  }
```

```
{
  "party": {
    "name": "Peter Miller Ltd"},
  "partyNameMatch": "MTCH",
},
{
  "uetr" : 98721c7e-ad88-49ec-a2ad-99ddcb1f2345
  "partyAccount": {"iban": "DE02100100109307118612"},
  "party": {
    "identification": {"organisationId": {"lei": "PetMullTD"}}
  },
  "partyIdMatch": "MTCH"
}
]
```

3.4.4 Get Status of Bulk Verification of Party Request

Call

GET /v1/{bulk-party-verification-endpoint}/{vopBulkId}/status

Retrieves the detailed status of a bulk of related party verification requests.

Path Parameters

Parameter	Description
bulk-party-verification-endpoint	The following values are supported in this specification: - bulk-payee-verifications: A service to fulfil related verifications for a bulk of payees. - bulk-payer-verifications: A service to fulfil related verifications for a bulk of payers. - bulk-party-verifications: A general service independent of the underlying payment scheme just addressing a bulk of counterparties.

Query Parameters

No query parameter

Header Parameters

Attribute	Type	Condition	Description
X-Request-ID	UUID	Mandatory	ID of the request, unique to the call, as determined by the initiating party.
API-Contract-ID	UUID	Conditional	ID of the underlying service contract between API Client and API Server, if provided by the ASPSP in the related onboarding process as defined in [oFA-IG-ADM].

Request Body

n.a.

Response Code

The HTTP response code equals 200.

Response Header

Attribute	Type	Condition	Description
X-Request-ID	UUID	Mandatory	ID of the request, unique to the call, as determined by the initiating party.

Response Body

Attribute	Type	Condition	Description
vopBulkId	UUID	Mandatory	
vopBulkStatus	VOP Bulk Status	Mandatory	
reasonCode	Status Reason Code	{Optional}	
reasonProprietary	Max35Text	Optional Or}	

Example

```
GET https://www.testbank.com/openFinance/v1/bulk-payee-verifica-
tions/dddd1c7e-ad88-49ec-a2ad-99ddcb1f7723/status
Content-Type:          application/json
X-Request-ID:          99391c7e-ad88-49ec-a2ad-99ddcb1f7727
Date:                  Sun, 27 Feb 2018 18:02:46 GMT
```

Response

```
HTTP/1.x 200
X-Request-ID:          99391c7e-ad88-49ec-a2ad-99ddcb1f7727
Date:                  Sun, 27 Feb 2018 18:02:47 GMT
Content-Type:          application/json
```

```
{  "vopBulkId": "dddd1c7e-ad88-49ec-a2ad-99ddcb1f7723",
   "vopBulkStatus": "completed"}
```

4 Integration in Payment APIs

A variant of a VOP functional support for initiating payments are a direct integration into existing payment APIs as they are defined in [XS2A-IG], [oFA-IG-Com] or [oFA-IG-EPIS].

NOTE: In the Instant Payment Regulation, it is stated that the initiating PSP is responsible for performing the VOP function. For this reason, the below definitions are defined as an extended service and might be used only if agreed between ASPSP and TPP.

4.1 Changes in Payment Request Messages

The only change are new header parameters for the integrated VOP API function, indicating the request to the ASPSP to perform a VOP function: The interpretation is as described below. Please note that this service integration is defined as extended service. That means that these headers will be ignored by the ASPSP if the integration of VOP services into the related payment initiation service is not supported. These headers could also be ignored only for some payment products².

Header Parameter

Attribute	Type	Condition	Description
Client-VOP-Requested	Boolean	Optional	<code>true</code>: The client requests the ASPSP to perform a VOP check, where applicable.³ <code>false</code>: The API Client requests the ASPSP not to perform a VOP check. Reasons for this are e.g. that the API client has already performed the VOP check in a TPP scenario⁴ or is requiring an opt out in a direct access scenario. The related VOP check could have been performed up front via the dedicated VOP API defined in Section 3.4, if provided by the ASPSP. <code>not used</code>: The API Client provides no information on requirements related to a potential VOP request. An ASPSP may ignore this header.

² Please refer also to EU COM DG FISMA Q&A #135.

³ This VOP check could be processed by the ASPSP e.g. to fulfill the Instant Payment Regulation, Article 5c, 1.

⁴ As described in the Instant Payment Regulation, Article 5c 2,3).

Attribute	Type	Condition	Description
Client-VOP-Request-ID	UUID	Conditional	This attribute contains the X-Request-ID of the related request on the dedicated VOP API. This attribute might be ignored by the ASPSP. If the ASPSP offers the dedicated VOP API, it might mandate the usage of this attribute via the ASPSP documentation and then make the usage of this attribute mandatory, if applicable to the related payment product and if VOP has not been opted out.
API-Contract-ID	UUID	Conditional	ID of the underlying service contract between API Client and API Server, if provided by the ASPSP in the related onboarding process as defined in [oFA-IG-ADM].

NOTE: In case of bulk payment initiation, the ASPSP might mandate the usage of `instructionId` or `uetr` for all entries in the bulk to be able to refer to the entries which might lead to the need of PSU confirmation due to a.o. the VOP outcome. At a minimum ASPSP should reject messages, where entries are contained which need a PSU confirmation and which come **without** the `instructionId` **and** the `uetr` attribute.

4.2 Changes in Payment Response Message

For the response messages to the Payment Initiation Request, the following changes apply:

Additional Header Parameter

Attribute	Type	Condition	Description
ASPSP-VOP-Performed	Boolean	Optional	<code>true</code>: The ASPSP has performed VOP successfully. Additional attributes are contained in the message body. <code>false</code>: The ASPSP has not performed VOP or not performed successfully. <code>not used</code>: No information provided..

Additional Body Parameter for Single Payment Initiation Response

Tag	Type	Usage	Description
creditorNameMatch	Party Name Match Code	Conditional	Will be provided, if an account/name match was requested and if the related code in the inter-bank VOP service was "CMTC", "NMTC" or "NOAP".
creditorOrganisationIdMatch	Party Identification Match Code	Conditional	Will be provided, if an account/name match was requested and if the related code in the inter-bank VOP service was "NMTC" or "NOAP".
creditorName	Max140Text	Optional	Provided only in case of "CMTC" as code in creditorNameMatch
_links	Link	Conditional	In case of a no match, near match or not applied match even if requested by the TPP, no authorisation related links will be contained in the response. No authorisation sub-resource is yet created. Instead, the PSU is asked via the TPP to confirm that the payment shall be initiated anyhow. This confirmation is offered via a "creditorNameConfirmation" link element. In case of a match, the _links attribute is as currently defined.

Additional Body Parameter for Bulk Payment Initiation Response

Tag	Type	Usage	Description
originalTransactionInformationAndStatus	Array of Original Transaction	Optional	Provided only in case at least one entry of the bulk leads to a status, where a confirmation is needed.

Tag	Type	Usage	Description
	tion Information And Vop Status		
_links	Link	Conditional	In case of a no match, near match or not applied match even if requested by the TPP, no authorisation related links will be contained in the response. Instead, the PSU is asked via the TPP to confirm that the payment shall be initiated anyhow. This confirmation is offered via a "creditorNameConfirmation" link element. In case of a match, the _links attribute is as currently defined.

NOTE: In difference to the dedicated VOP API, the data attributes for the party here are adapted to the situation of submitting credit transfers as payments, i.e. using creditorName-Match instead of partyNameMatch. The idea is to make the response more easy understandable to the API Client, when integrating it into the payment flow.

NOTE: The following External Payment Transaction Status codes as well as External Group Status codes have been successfully applied by Berlin Group and PSA towards ISO20022 to deal with the situation,

- where the payment or bulk payment needs an explicit confirmation, or
- where the VOP service for a submitted bulk payment has been started already but is not yet finalised for all entries.

The addressed ISO20022 External codes are not yet published by ISO20022, but publication is expected during spring 2025.

Code	Name	Description
RVCM	ReceivedVerification-CompletedWithMis-matches	The verification of party request for the payment counterparty result requires an explicit confirmation by the payer.

Code	Name	Description
RVNC	ReceivedVerification-NotCompleted	The verification of party request is not yet completed.
RCVC	ReceivedVerification-Completed	The verification of party request is completed, no further confirmation required.

NOTE: If the status of a single payment or bulk payment equals RVCM, the related payment or (full) bulk payment will not be processed. In the bulk payment scenario, this means that the PSU can only confirm (or not) the full bulk. A differentiation for several entries is not possible. The latter can only be achieved by cancelling the current bulk payment initiation and start a new one, taking out some of the entries.

4.3 Changes in (Extended) Status Request

By extending the transactionStatus by new codes, the status on bulk and entry level will be separated also by introducing the attribute groupStatus, which will replace the transactionStatus attribute on group level. The groupStatus as well as the transactionStatus come with new valued and the extended status with more attributes as follows.

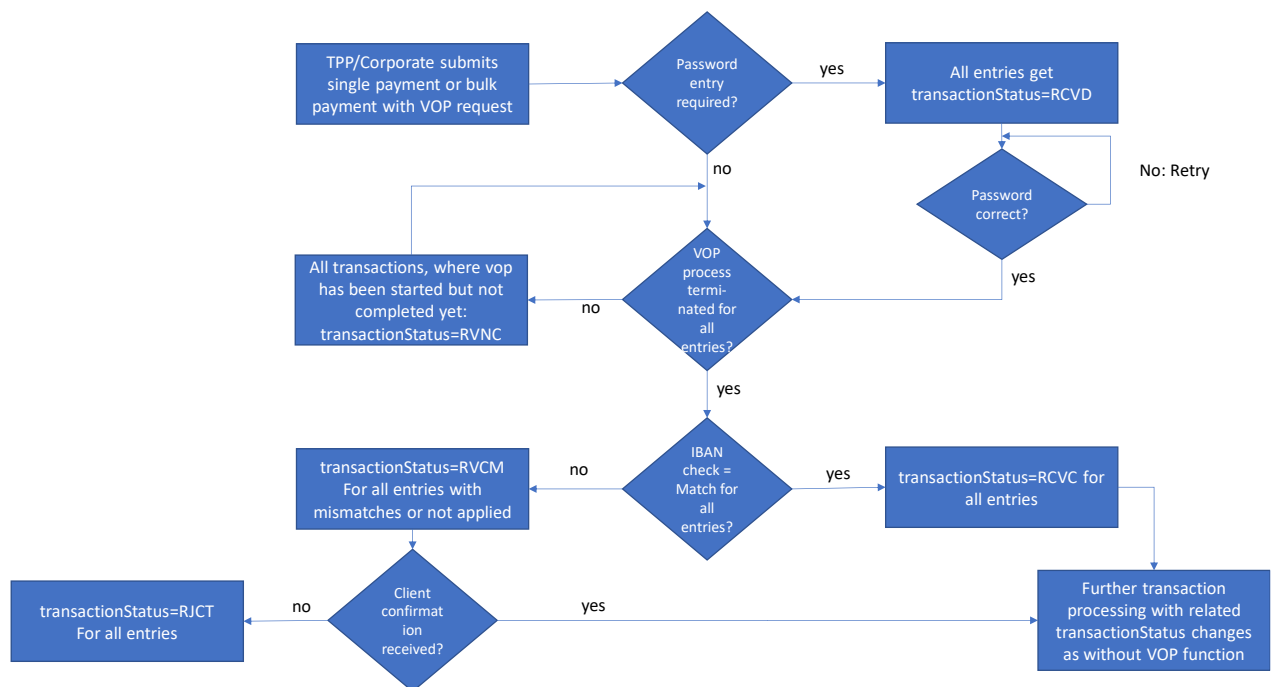
New Group Status Diagram (only relevant for bulk payment)

The newly introduced group status will reflect during the receiving phase the following codes. In the later authorisation phase, the groupStatus will reflect the status of the bulk as before.



New Transaction Status Diagram

The new Transaction Status Codes introduced above are used as described in the following diagram (on entry level in case of bulk payments):



Extended Response Body Parameter

The following attribute in the response will get an extended data type and a new description:

Tag	Type	Usage	Description
originalTransactionInformationAndStatus	Array of Original Transaction Information and VOP Status	Conditional	Provides information on the original transactions to which the status report message refers and which have been rejected or which need an explicit confirmation following VOP processing results.

4.4 New Method for Confirmation of Creditor Name and Payment

In case of no match, near match or no match applied, the PSU has still the right to initiate the payment, after an explicit confirmation towards the ASPSP. If such a confirmation in these cases is not provided, the payment is not initiated.

This explicit confirmation is defined technically as follows:

Call

```
PUT /v2/{payment-service}/{payment-product}/{paymentId}/creditor-confirmation
```

Submits a Confirmation of Creditor Name Request

NOTE: From a standards perspective, this access method applies also to v1/{payment-service}, i.e. to the existing NextGenPSD2 APIs.

Path Parameters

Attribute	Type	Condition	Description
payment-service	Max35Text	Mandatory	The payment service where the VOP has been integrated.
payment-product	Max35Text	Mandatory	The payment product which is addressed by the PSU/TPP and where the VOP has been integrated.
paymentId	Max70Text	Mandatory	The resource identification of the payment initiation, where the VOP has been integrated.

Query Parameters

No query parameter

Header Parameters

Attribute	Type	Condition	Description
X-Request-ID	UUID	Mandatory	ID of the request, unique to the call, as determined by the initiating party.

Request Body

No body.

Response Code

The HTTP response code equals 200.

Response Header

Attribute	Type	Condition	Description
X-Request-ID	UUID	Mandatory	ID of the request, unique to the call, as determined by the initiating party.

Response Body

Attribute	Type	Condition	Description
transactionStatus	Transaction Status	Mandatory	
_links	Links	Mandatory	The ASPSP is putting here all authorisation related links, which are normally provided already in a Transaction Initiation Response message as defined in [oFA-PFSM].

5 References

5.1 Normative References

The specification contained in this document make use of definitions in the related documentation.

- [oFA-DaD] openFinance API Framework, Data Dictionary, Version 2.2.3, 24 January 2025
- [oFA-PFSM] openFinance API Framework, Protocol Functions and Security Measures, Version 2.1, 31 July 2024
- [oFA-IG-ADM] openFinance API Framework, Administrative Services, Implementation Guidelines Version1.0, 21 February 2024
- [oFA-Disc] openFinance API Framework, Discovery Services, Version 1.1, 24 January 2025
- [PSD2] Directive (EU) 2015/2366 of the European Parliament and of the Council on payment services in the internal market, published 23 December 2015
- [eIDAS] Regulation (EU) No 910/2014 of the European Parliament and of the Council on Electronic Identification and Trust Services for Electronic Transactions in the Internal Market, 23 July 2014, published 28 August 2014

5.2 Informative References

- [oFA-IG-Com] openFinance API Framework, Implementation Guidelines, Compliance Services, Version 2.2, 24 January 2025
- [oFA-IG-EPIS] openFinance API Framework, Implementation Guidelines, Extended Payment Initiation Services, Version 2.1, 31 July 2024
- [oFA-IG-RTP] openFinance API Framework, Implementation Guidelines, Request to Pay Services, version 2.0, October 2023
- [XS2A-IG] NextGenPSD2 XS2A Interface, Implementation Guidelines, Version 1.3.14, 24 January 2025