THE Berlin GROUP

A EUROPEAN STANDARDS INITIATIVE

openFinance API Framework

Implementation Guidelines for Extended Services

Push Account Information Services

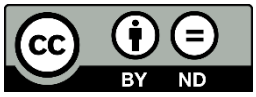
Version 2.0

13 February 2024

License Notice

This Specification has been prepared by the Participants of the openFinance Taskfoce*. 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.

* 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

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 planned to be broadened to extended 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:

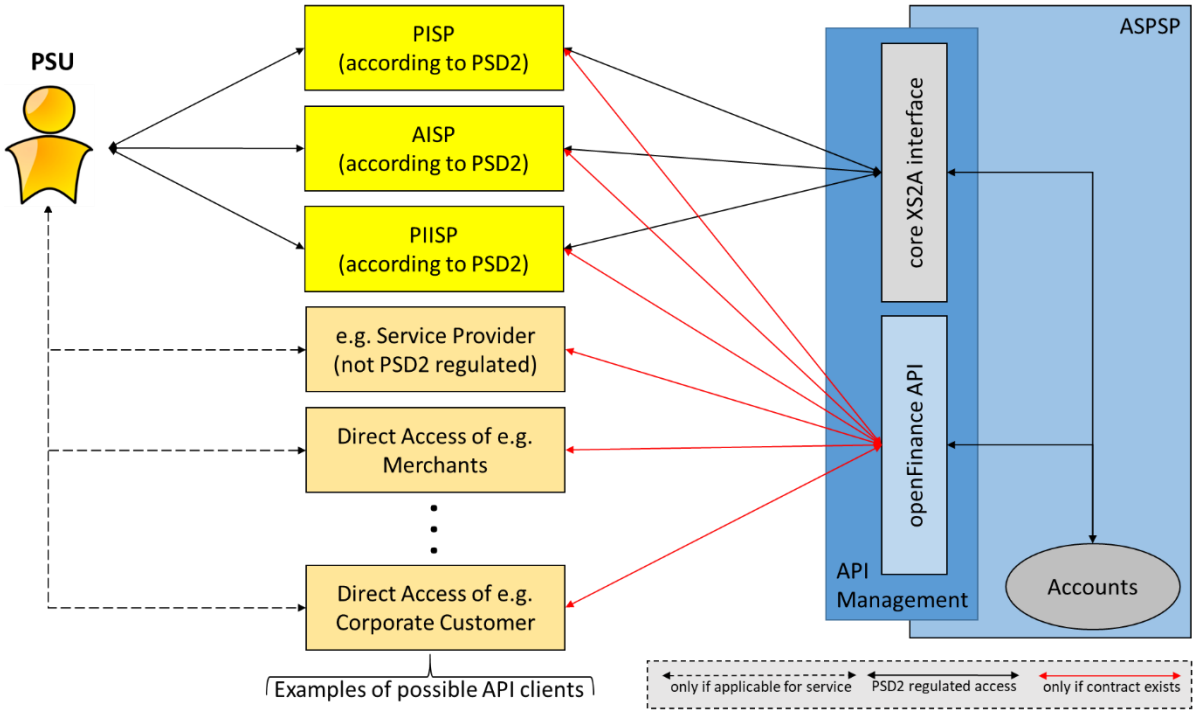


Figure 1: Core XS2A interface and openFinance API

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 are described in a dedicated operational rules document [oFA-OR-ADM].

1.2 Push Account Information Services

The core XS2A Interface as introduced above is for Account Information Services (AIS) in general based on a "pull philosophy": The TPP can **pull** AIS related information from the ASPSP once the PSU has given his/her consent and the PSU is authenticated by SCA methods provided by the ASPSP. Triggers for pulling can be the PSU as such (PSU present at the TPP's application asking for a refresh) or a 4 times offline (PSU not present) access of the TPP to the PSU's accounts per day.

A first "push functionality" was introduced in the core XS2A interface for informing the TPP about technical status changes of major API resources which have been submitted by the TPP itself, e.g. an information that a payment has been authorised by the PSU in a decoupled SCA process. The core XS2A interface does not support pushing new financial account related information like new account entries for e.g. incoming instant payments.

It is now planned to introduce a service to enable ASPSPs to inform PSUs via API Client systems about account information into the openFinance API Framework. This service might require a contract between the API Client and the ASPSP. It is called in the following **Push**

Account Information Service. The openFinance API Framework will support several subservices from the beginning.

As a first subservice, the pushing of account entries is standardised. This subservice is called **Push Account Entry Service**. The Push Account Entry service will be available in two variants:

- Push an account entry directly to an **entry URI** provided by the API Client, or
- Push a trigger for the payment transaction received to a **trigger URI** provided by the API Client. In a next step, the TPP would need to pull the transaction data, using an underlying PSU consent.

A next subservice is a service to push account statements. The statements are containing more information as the transaction reports as provided today within the PSD2 NextGenPSD2 API. This subservice is called **Push Account Statement Service**. This service will allow to transport account statements either in MT94x or in camt.05x format. **RFU**: In future, also JSON based statements will be supported by this standard.

A third subservice is pushing balance related information, e.g. inform the API Client in case a certain balance threshold is met. This subservice is called **Push Balance Information Service**.

A fourth subservice is pushing incoming RTP Requests, e.g. forward all incoming RTP Requests which are due in a certain short time frame or warn about RTP Request reaching the expiry date. This subservice is called **Push RTP Information Service**.

Please note that all these services come with a subscription by the PSU, which is initiated by the API Client. Each subservice might be triggered by events, pre-defined timeslots etc. The trigger criteria, or "criteria" in short in this text, are agreed on during the subscription process.

Please note that this document is using security measures, protocol functions, notations, data types and character sets as defined in the framework documentation [oFA-PFSM] and [oFA-DaD]

Remark: Even if the subscription model followed within this service is a GDPR like consent between ASPSP and PSU on forwarding account information data to a third party, PSD2 requirements might still apply on e.g. SCA related periods. This is under review and up to API Access Definitions or bilateral commercial contracts between API Client and ASPSP. Anyhow, variants where just e.g. hyperlinks pointing to new account information or static trigger text are pushed to the API Client might be easier implemented from a legal/contractual point of view.

Note: The Push Account Information Service for the openFinance API is built in such a way that it can also be used in a direct PSU – ASPSP interface. Some remarks specific to such an implementation are integrated in this document.

1.3 Document Structure

This document specifies the Push Account Information Services in detail.

Section 2 is providing specific, but still abstract information on the application layer of this service like service API access methods or service specific error codes.

Section 3 defines the API for subscribing to Push Account Information Services in detail. Section 4 shortly defines the push messages to be used towards the webhooks registered within the subscription of the related Push Account Information Service.

1.4 Document History

Version	Change/Note	Approved
0.84	Draft for pre-consultation with the Advisory Group	23 September 2021
0.9	Draft for public market consultation	11 October 2021
1.0	Final editorial corrections	16 December 2021
1.1	Section 2.1: Added Clarifications on Signature Headers Section 2.5: Added clarifications on HTTP response code usage Section 3: Authorization header added to all access methods. New Section 3.1 for description of exemplary flows. New Section 3.2 for a mapping of the abstract data model to API attributes. Section 3.3: Added a remark on PSU related header fieldsSection 3.3 as well as 3.7: Several authorisation and status related header fields corrected/added for the request and the related response Section 3.5 GET response: double attribute subscriptionStatus removed. Section 3.8 Corrected Data Type Subscription Entry Status Section 4.2: Added a note that card transactions are not dedicately supported The following sections in the complex data structures have been applied: debtorAccount and creditorAccount corrected to singular, Purpose to purpose;Data type of expiryReminder corrected to "Boolean";	15 February 2022

Version	Change/Note	Approved
	Reference to several Data Type from Payment data model added	
Version 2.0	Adaptation to openFinance API Framework Version 2. Sections about character sets, notations and complex data types as well as sections on certain additional data definitions have been removed. Related definitions are cited from the API Framework documentation or collected in API Framework documentation. Some re-finements in Section 2.2 on URI requirements have been applied. Service Codes have been added for PAIS services in Section 2.4. Status diagrams have been added in Section 2.6. Usage of HTTP headers are moved to the API Framework documentation and omitted in Section 3.3.1 Adding a function to retrieve all subscriptions related to a given PSU, cp. Section 3.4. Flattening data structures for subscription entries, cp e.g. Section 3.7 and 3.8. A JSON variant for pushing account statements has been added in Section 4.2 The extensions additionalCreditorInformation and additionalUltimateCreditorInformation have been transformed to additionalPartyInformation in the related creditor and ultimateCreditor data structures of the final transaction data type for Version 2.0. Some other minor changes have been applied to complex data structures which have been collected in the common data dictionary. All examples have been transformed to Version 2 data model. Especially all resource identifications have been moved to UUID	2024-02-13 openFinance TF

Version	Change/Note	Approved
	formats, following the recommendations of the API Framework.	

2 Application Layer: Guiding Principles

2.1 Signing Messages at Application Layer

The same conditions on signing messages by the TPP as defined in [oFA-PFSM] apply to the subscription process, cp. Section 3. These conditions also imply the signature related http headers.

2.2 Requirements on API Client URIs

This specification makes no requirements on the local endpoint structure of the API Client, i.e. the API Client is free to define host, service and transaction identifiers within the API-Notification-URI implementation, besides some security related items regarding the domain defined by the API Client, see below. Every pushing of account entries is done as a POST command towards the primary address

```
https://<apiClientPrimaryPushURI>
```

or the secondary address

```
https://<apiClientSecondaryPushURI>
```

using additional content parameters defined in JSON encoding.

For security reasons, it shall be ensured that the apiClientPrimaryPushURI and apiClientSecondaryPushURI as introduced above are secured by the API Client eIDAS QWAC used for identification of the API Client as described in [oFA-PFSM].. The following applies:

URIs which are provided by API Clients in the PUSH URIs shall comply with the domain secured by the eIDAS QWAC certificate of the API Client in the field CN or SubjectAltName of the certificate. Please note that in case of example-TPP.com as certificate entry apiClientPrimaryPushAPI or apiClientSecondaryPushAPI like

- www.example-TPP.com/xs2a-client/v2/ASPSPidentifcation/psu-account-id/entries
- pushentries.example-TPP.com/xs2a-client/v2/ASPSPidentifcation/psu-account-id/entries

would be compliant.

Wildcard definitions shall be taken into account for compliance checks by the ASPSP.

Example: https://pushaccountgateway.tpp-name.eu, where tpp-name.eu is the domain of a TPP as API Client.

2.3 openFinance API Structure

The Push Account Information Service consists of two components:

- the **subscription process** to activate (or de-activate) the pushing of account entries on a dedicated account and

- the **actual pushing** of account information.

The structure of the request/response for both processes is described according to the following categories

- Path: Attributes encoded in the Path (only applicable for the subscription process)
- Query Parameters: Attributes added to the path after the ? sign as process steering flags or filtering attributes for GET access methods. Query parameters of type Boolean shall always be used in a form query-parameter=true or query-parameter=false.
- Header: Attributes encoded in the HTTP header of request or response
- Request: Attributes within the content parameter set of the request
- Response: Attributes within the content parameter set of the response, encoded in JSON

The HTTP response codes, which might be used in openFinance API services, are specified in Section 2.5 They are not repeated for every API call definition.

2.4 API Access Methods

The following table gives an overview on the HTTP access methods supported by the API endpoints for the Push Account Entry Service.

Conditions in the following tables

Whether the support of a method is mandated for the ASPSP by this specification or whether is an optional feature for the ASPSP, is denoted in column "Condition". 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/subscriptions/{subscriptionId} applies only if the endpoint /v2/subscriptions is supported at all.

Please note that all methods called by an API Client, which are addressing dynamically created resources in this API, may only apply to resources, which have been created by the same API Client before.

Examples

Please note also, that the Description's column contains a reference to a section where the endpoint is described in more detail. These sections provide examples for all related access methods.

2.4.1 Subscription Endpoint

In the following, all access methods for subscribing to the Push Account Information Service are defined. The following potential instances of a push-accountinformation-subservice will be supported by this Framework:

- push-account-entries for subscribing to the Push Account Entry Service (PAIS-AES)
- push-account-statements for subscribing to the Push Account Statement Service (PAIS-AS),
- push-balances for subscribing to the Push Balance Information Service (PAIS-BI), and
- push-requests-to-pay for subscribing to the Push RTP Information Service (PAIS-RTP).

In addition to the endpoints and access methods introduced in the following, these services support in addition authrorisation related endpoints and access methods as defined in [oFA-PFSM] on subscription as well as on subscription entry level.

Endpoint	Method	Condition	Description
subscriptions/{push-accountinformation-subservice}	POST	Mandatory	This command is activating a push account information sub-service on one or more dedicated accounts. Activation is achieved by establishing a subscription, to which the ASPSP will assign a subscriptionId. Within this subscription, the TPP can submit several subscription entries with several filtering and content criteria selected. The ASPSP will associate each of these subscription entries is with a subscriptionEntryId. This access method requires an SCA. Cp. Section 3.3
subscriptions/{push-accountinformation-subservice}/{subscriptionId}	GET	Mandatory	Show information about the activated subscription and the related subscription entries, specifically delivering the entryId of every subscription entry. Cp. Section 3.5
subscriptions/{push-accountinformation-subservice}/{subcriptionId}/status	GET	Mandatory	Show the status of a subscription. Cp. Section 3.6

Endpoint	Method	Condition	Description
subscriptions/{push-accountinformation-subservice}/{subscriptionId}	DELETE	Mandatory	De-activate the addressed subscription. This access method does not require SCA. Cp. Section 3.10
subscriptions/{push-accountinformation-subservice}/{subscriptionId} }/subscription-entries	POST	Mandatory	Add a subscription entry to an existing subscription. This access method will require an SCA. Section 3.7
subscriptions/{push-accountinformation-subservice}/{subscriptionId}/subscription-entries/{subscriptionEntryId}	GET	Mandatory	Show information about the activated subscription entry See Section 3.8
subscriptions/{push-accountinformation-subservice}/{subscriptionId}/subscription-entries/{subscriptionEntryId}/status	GET	Mandatory	Shows the status of a subscription entry. See Section 3.9
subscriptions/{push-accountinformation-subservice}/{subscriptionId}/subscription-entries/{subscriptionEntryId}	DELETE	Mandatory	De-activates a dedicated subscription entry. This access method does not require SCA. See Section 3.11

2.4.2 Push Account Information Endpoint

Endpoint	Method	Condition	Description
<apiClientPrimaryPushURI>	POST	Mandatory	This command is pushing account information of a dedicated Push Account Information subservice to the addressed URI. This access method shall be supported by the API Client if a subscription of a related "Push Account Information subservice" has been initiated by the API Client in a

Endpoint	Method	Condition	Description
			previous call to the openFinance API of the ASPSP. See Section 4.2
<apiClientSecondaryPushURI>	POST	Optional	This command is pushing account information of a dedicated Push Account Information subservice to the addressed secondary URI if the endpoint under the primary URI is not reachable for the ASPSP. This access method shall be supported by the API Client if a subscription of a related "Push Account Information subservice" including a secondary PUSH URI has been initiated by the API Client in a previous call to the openFInance API of the ASPSP.

2.5 HTTP Response Codes

The HTTP response code by the ASPSP server is communicating the success or failure of a TPP request message. The 4XX HTTP response codes should only be given if the current request cannot be fulfilled, e.g. the syntax of the body content is not correct or it does not match the requirements of the addressed endpoint of the API. The permitted HTTP codes for the ASPSP are documented within [oFA-PFSM].

This specification supports in addition the following HTTP response codes for the push account endpoint of the API Client Push Entry API, communicating the success or failure of an ASPSP request message:

Status Code	Description
200 OK	POST for pushing an account entry.
400 Bad Request	Validation error occurred. This code will cover malformed syntax in request or incorrect data in payload.
401 Unauthorized	The ASPSP is not correctly authorized to perform the request. Retry the request with correct authentication information.
403 Forbidden	Returned if the resource that was referenced in the path exists but cannot be accessed by the ASPSP. This code should only be used for non-sensitive id references as it will reveal that the resource exists even though it cannot be accessed.
404 Not found	Returned if the endpoint that was referenced in the path does not exist or cannot be referenced by the ASPSP. When in doubt if a specific id in the path is sensitive or not, use the HTTP response code 404 instead of the HTTP response code 403.
405 Method Not Allowed	This code is only sent when the HTTP method (PUT, POST, DELETE, GET etc.) is not supported on a specific endpoint.
408 Request Timeout	The server is still working correctly, but an individual request has timed out.
415 Unsupported Media Type	The ASPSP has supplied a media type which the TPP does not support.
500 Internal Server Error	Internal server error occurred.
503 Service Unavailable	The TPP server is currently unavailable. Generally, this is a temporary state.

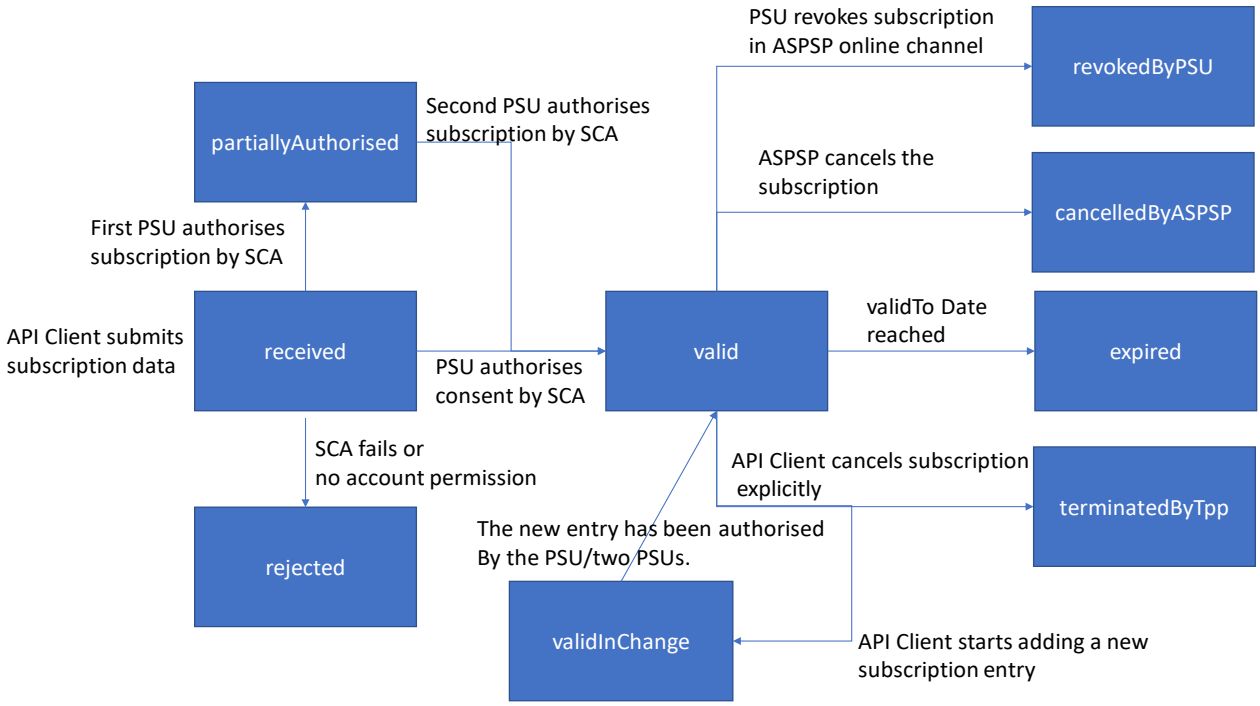
2.6 Subscription and Subscription Entry Status

The subscription data model provides status information on subscription resource as well as on subscription entry level.

All status values besides "received", "partiallyAuthorised" and "validInChange" are final statuses.

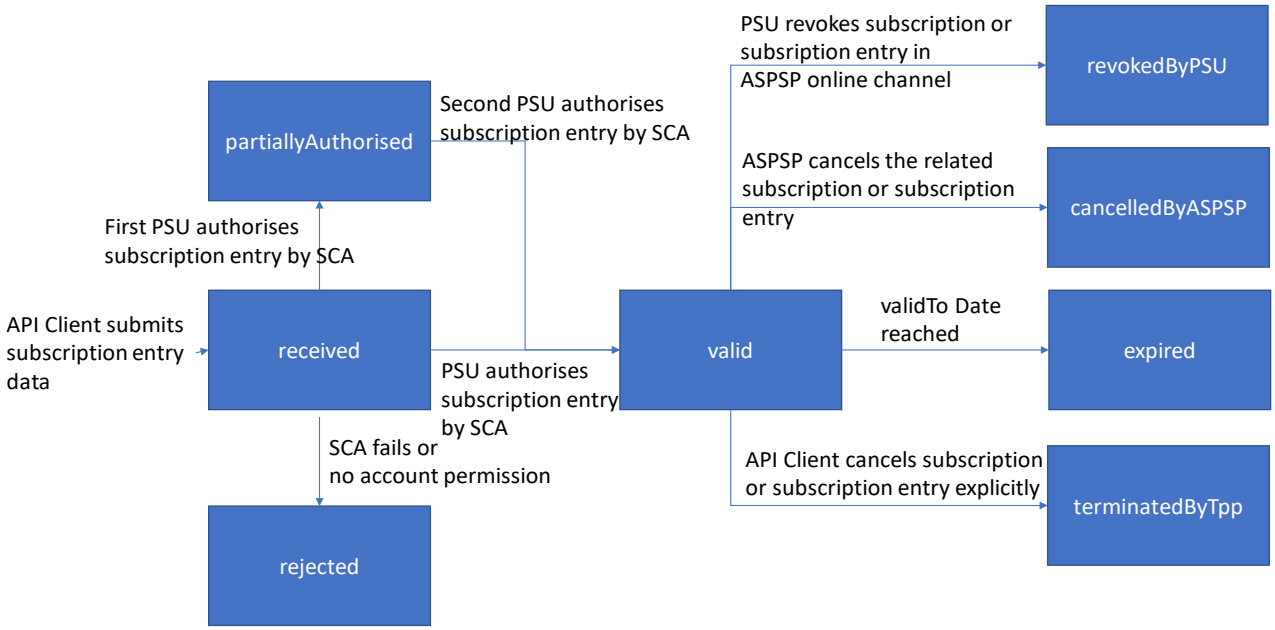
Once the subscription as such has the status "valid", it will switch over to "validInChange" only during authorisation processes for new subscription entries. Invalidation of certain entries (either by expiration time, API Client or PSU) will have no effect on the subscription status.

The following picture provides an overview on the subscription status and regular transitions:



Exeptional status transitions like from received to terminatedByTPP or validInChange to revokedByPSU are not shown in the diagram.

The subscription entry status model is a bit more simple, since a subscription entry as such cannot be changed, but just deleted and newly submitted. Thus, the following subscription entry status applies:



2.7 Resource Availability

A valid subscription resource will be available as long as it is not invalidated either by expiration time, API Client or PSU. After the invalidation, the ASPSP might remove the resource data from the API after some time.

Valid subscription entry resources will be available as long as they are not invalidated either by expiration time, API Client or PSU or never have been fully authorised. After the invalidation, the ASPSP might remove the entry resource data from the API after some time, even if the related subscription is still valid.

2.8 Model for direct access for Corporates

This service is also relevant when the API Client is a direct corporate customer of the ASPSP and potentially gets Account Information directly pushed to the API Client's IT systems via this service. A potential impact is e.g. on using signed requests for proving SCA, performed by signature media issued by the ASPSP to the corporate customer. The related requirements will soon be published in a new version of [oFA-PFSM].

3 Subscription for Account Information Service

NOTE: The Account Information Push Service is an extended service of the openFinance API Framework. This specification makes no assumption whether a contract may be needed for the ASPSP to offer this service to API Clients.

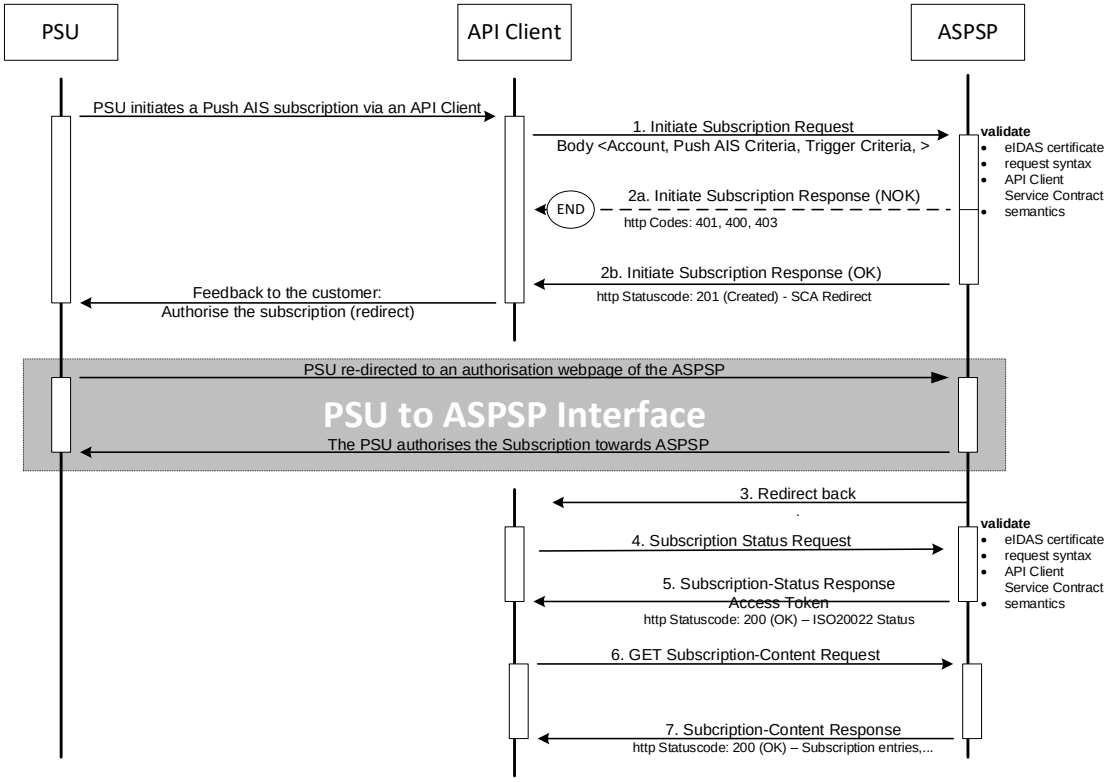
3.1 Push Account Information Service Flows

Please note that the following flows do not cover all possible variances and are exemplary flows, cp. related AIS consent flows in [oFA-Co] .

3.1.1 Push Account Information Subscription Flow

3.1.1.1 Redirect SCA Approach: Implicit Start of the Authorisation Process

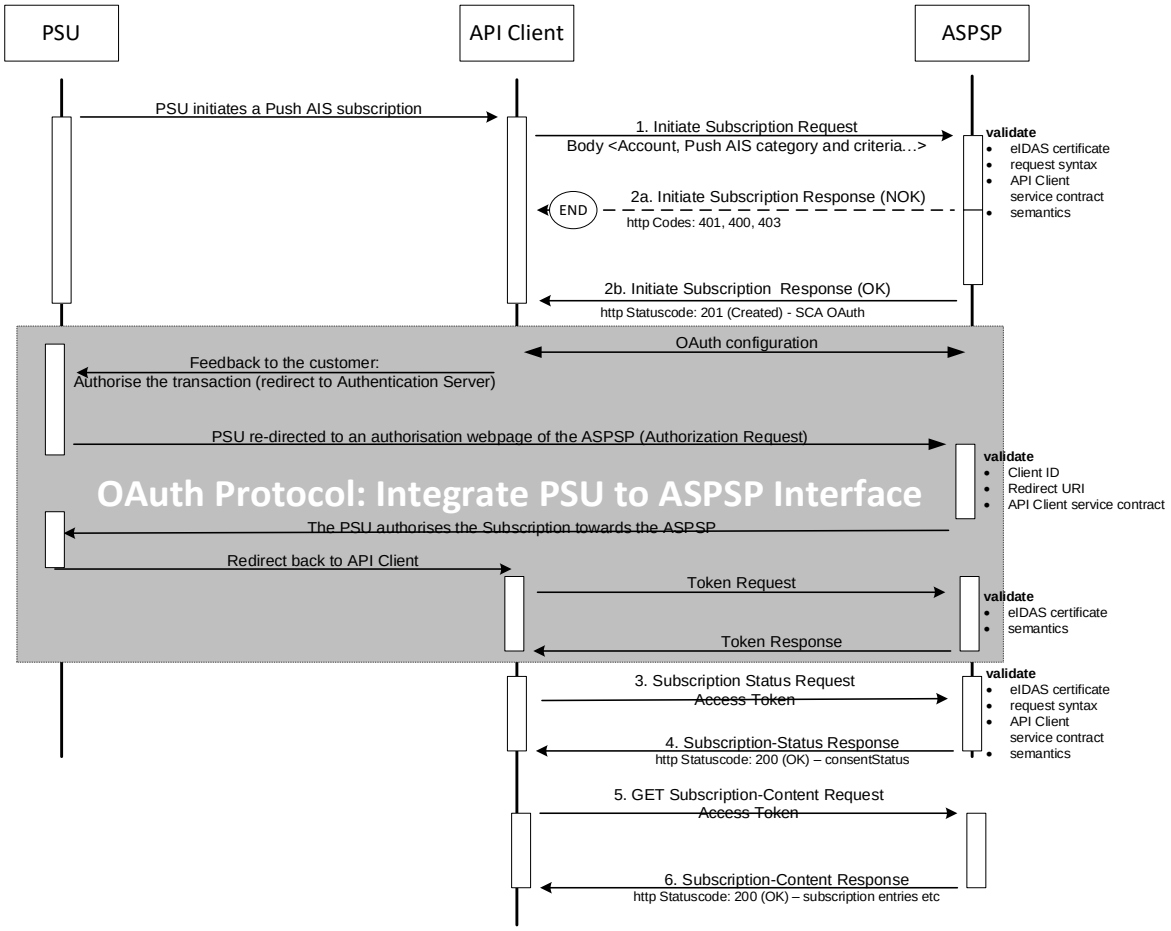
If the ASPSP supports the Redirect SCA Approach, the message flow within the Push Account Information Subscription sub-service is simple. The Push Account Information Subscription Request is followed by a redirection to the ASPSP SCA authorisation site. A status or content request on the created subscription resource might be requested by the API Client after the session is re-redirected to the API Client's system.



3.1.1.2 OAuth2 SCA Approach: Implicit Start of the Authorisation Process

If the ASPSP supports the OAuth2 SCA Approach, the flow is very similar to the Redirect SCA Approach. Instead of redirecting the PSU directly to an authentication server, the OAuth2

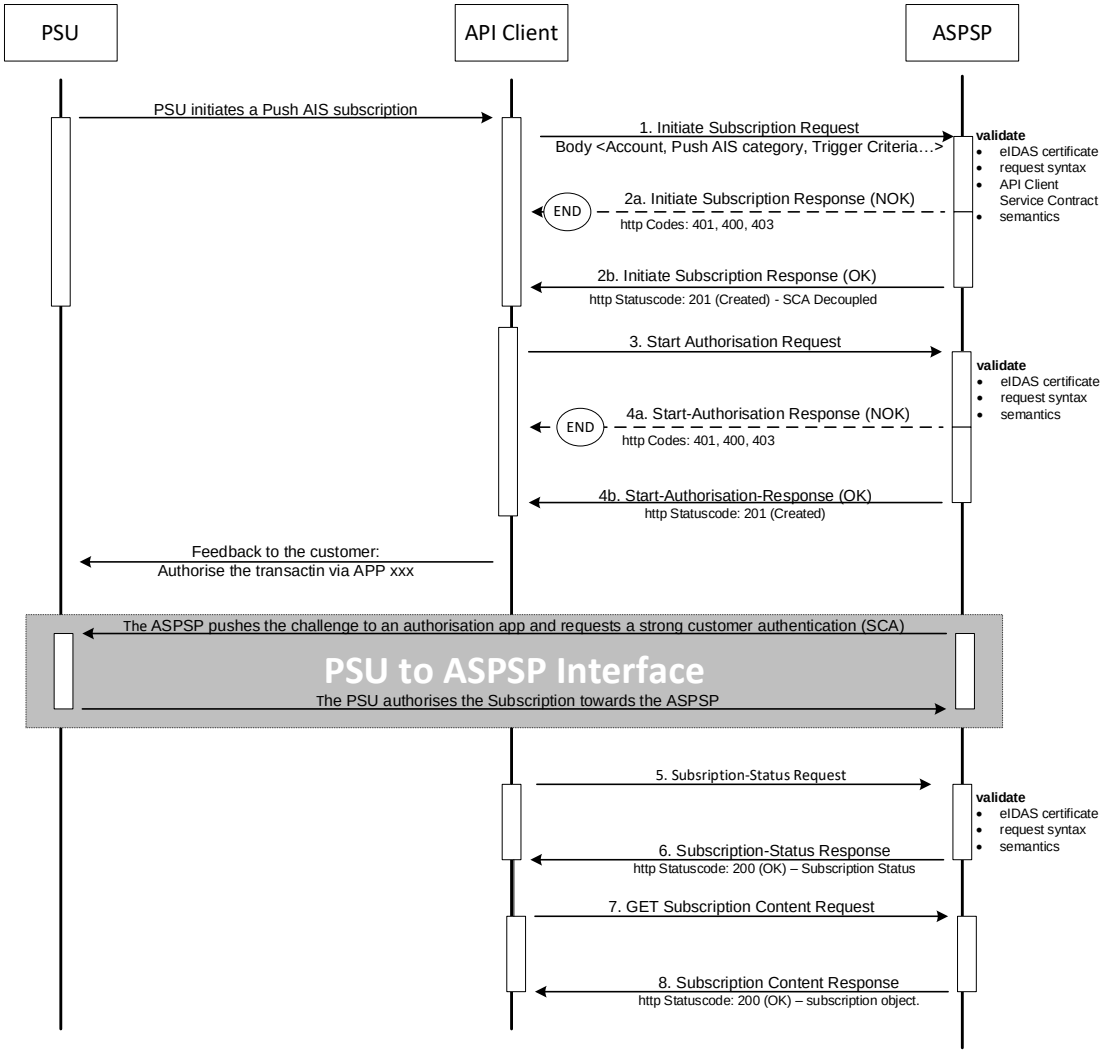
protocol is used for the transaction authorisation process. In the following, a flow is shown, where the Authorisation Process in the opfenFinance API has been implicitly started.



3.1.1.3 Decoupled SCA Approach: Explicit Start of the Authorisation Process

The transaction flow in the Decoupled SCA Approach is similar to the Redirect SCA Approach. The difference is that the ASPSP is asking the PSU to authorise the Push AIS subsription e.g. via a dedicated mobile app. The ASPSP is asking the TPP to inform the PSU about this authentication by sending a corresponding PSU Message like "Please use your xxx App to authorise the Push AIS subscription".

After the SCA between ASPSP and PSU, the API Client then needs to ask for the result of the transaction.



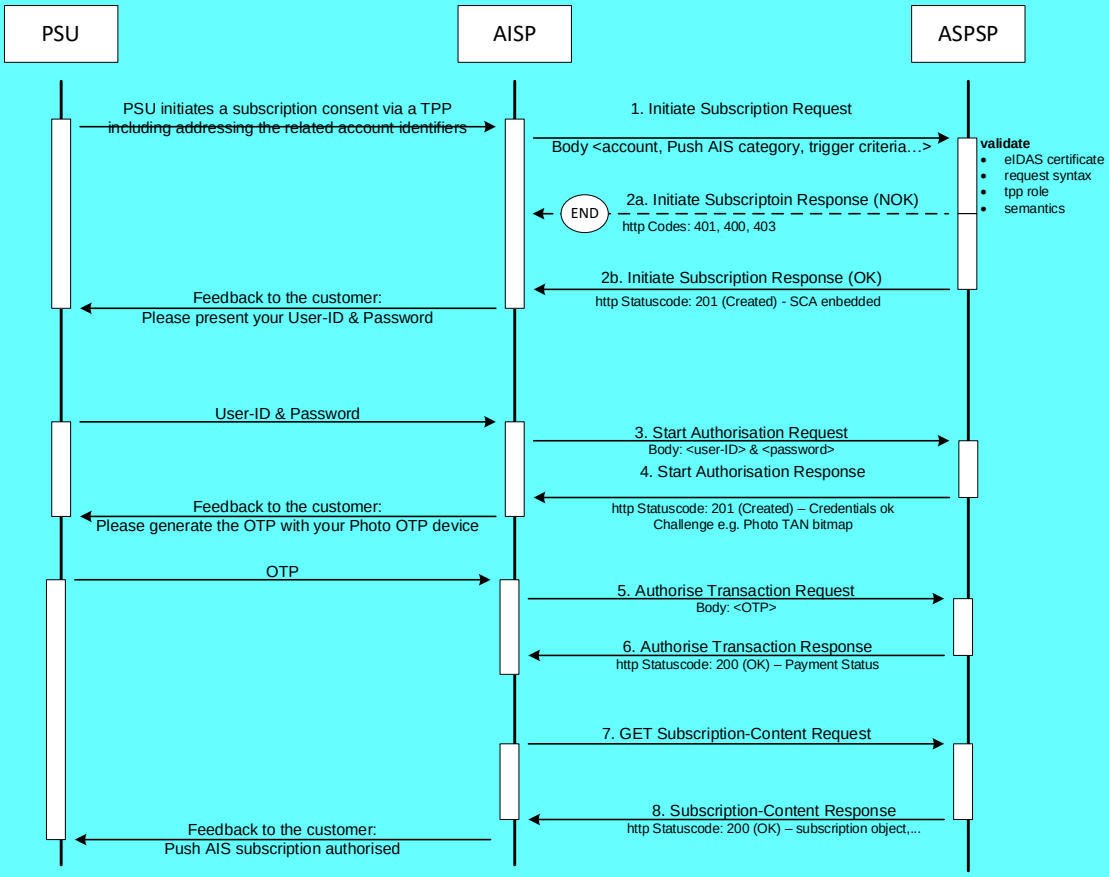
3.1.1.4 Embedded SCA Approach with only one SCA method available

In the following, several exemplary flows are shown, where the ASPSP has chosen to process the SCA methods for the subscription authorisation through the API Client – ASPSP interface.

Remark: In case where OAuth2 is requested by the ASPSP as a pre-step to replace the PSU- and password by an access token, the sequence of the PSU authentication

with the first authentication factor is omitted. This applies for all examples for the Embedded SCA Approach.

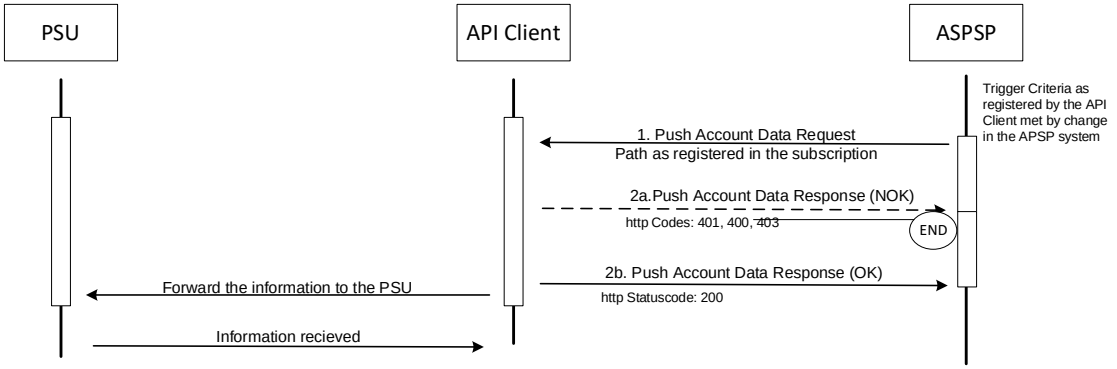
In case where only one SCA method is available, the "Authorise Transaction Request" is added to the flow, where the TPP is transmitting the authentication data of the customer, e.g. an OTP with included dynamic linking to the transaction details.



3.1.2 Push Account Information Data Flow

The Push Account Data flow is independent from the corresponding Subscription Management flow. It is a simple Request/Response process as follows, initiated by the ASPSP

when push trigger criteria, which have been registered during subscription by the API Client, are met, e.g. by receiving new incoming instant payments.



3.2 Data Overview Push Account Information Service

The following table defines the technical description of the abstract data model as defined in [oFA-OR-PAIS] for the subscription services within the Push Account Information Services.

The following methodology applies

- The "Data element" column is using the abstract data elements following [oFA-OR-PAIS] to deliver the connection to rules and role definitions in this document.
- The "Attribute encoding" is giving the actual encoding definition within the openFinance API as defined in this document. Please note that some data structures are dependent on the account information category to be pushed. These data attributes are then given by introducing an abstract place holder accountInformationCategory which needs to be instantiated yet by the actual category.
- The "Location" columns define, where the corresponding data elements are transported as HTTP parameters, resp. are taken from eIDAS certificates.
- The "Usage" column gives an overview on the usage of data elements in the different API Calls. Within [oFA-OR-PAIS], the openFinance API calls are described as abstract API calls. These calls will be technically realised as HTTP POST, PUT, DELETE and GET commands. The calls are divided into the following calls:
    - Initiate Subscription Request, which shall be the first API Call for starting a new subscription for a push account information function.
    - The Read Data Request is the request to retrieve Account Information data, which is addressed to different endpoints with different parameters.
    - The Status Request is used in cases, where the SCA control is taken over by the ASPSP and the TPP needs later information about the outcome.

The following usage of abbreviations in the Location and Usage columns is defined

- x: This data element is transported on the corresponding level.
- m: Mandatory
- o: Optional for the API Client to use
- c: Conditional. The Condition is described in the API Calls, condition defined by the ASPSP

The following table does not only define requirements on request messages but also requirements on data elements for the response messages. These requirements only apply to positive responses (i.e. HTTP response code 2xx).

Remark: The more technical functions like GET …/{consentId} and GET …/{authorisationId} and the Cancellation Request are not covered by this table.

Data element	**Attribute encoding**	**Location**					**Usage**					
		Path	Query Param.	Header	Body	Certificate	Initiate Subscr. Req.	Initiate Subsrc. Resp.	Add Subscr. Req.	Add Subscr. Resp.	Status Req.	Status Resp.
Provider Identification		x					m		m		m	
API Client Identification						x	m		m		m	
API Client Name						x	m		m		m	
API Client Role						x	c		c		c	
API Client National Competent Authority						x	c		c		c	
Request Identification	X-Request-ID			x			m	m	m	m	m	m
Resource ID	subcriptionId				x			m				
Resource ID[2]		x							m		m	

[2] Please note that the Resource ID is transported in the path after the generation of the consent resource. This is then a path parameter without an explicit encoding of the attribute name.

Data element	Attribute encoding	Location					Usage					
		Path	Query Param.	Header	Body	Certificate	Initiate Subscr. Req.	Initiate Subsrc. Resp.	Add Subscr. Req.	Add Subscr. Resp.	Status Req.	Status Resp.
Access Token (from optional OAuth2)	Authorization			x			c		c		c	
API Client Signing Certificate Data	Client-Signature-Certificate			x			c		c		c	
API Client Signing Electronic Signature	Signature			x			c		c		c	
Further signature related data	Digest			x			c		c		c	
ASPSP-SCA-Approach	ASPSP-SCA-Approach			x				c		c		
Transaction Status	subscriptionStatus				x			m		m		m
SCA Status	scaStatus				x							o
PSU Message Information	psuMessage				x			o		o		o
API Client Message Information	apiClientMessages				x			o		o		o
PSU Identification	PSU-ID			x			c		c			
PSU Identification Type	PSU-ID-Type			x			c		c			
Corporate Identification	PSU-Corporate-ID			x			c		c			
Corporate Type	PSU-Corporate-ID-Type						c		c			
Available SCA Methods	scaMethods				x			c		c		
IP Address PSU	PSU-IP-Address			x			m		o		o	
PSU IP Port	PSU-IP-Port			x			o		o		o	

Data element	Attribute encoding	Location					Usage					
		Path	Query Param.	Header	Body	Certificate	Initiate Subscr. Req.	Initiate Subsrc. Resp.	Add Subscr. Req.	Add Subscr. Resp.	Status Req.	Status Resp.
Further PSU related Information	PSU-Accept			x			o		o		o	
	PSU-Accept-Charset			x			o		o		o	
	PSU-Accept-Encoding			x			o		o		o	
	PSU-Accept-Language			x			o		o		o	
	PSU-Http-Method			x			o		o		o	
	PSU-Device-ID			x			o		o		o	
PSU User Agent	PSU-User-Agent			x			o		o		o	
GEO Information	PSU-Geo-Location			x			o		o		o	
Redirect URL ASPSP	_links.scaRedirect				x			c				
Client SCA Preference	TPPSCA-Preference			x			o		o			
Redirect URL TPP	TPP-Redirect-URI			x			c		c			
Authorisation Preference	TPPExplicit-Authorisation-Preferred			x			o		o			
Client Notification URI	Client-Notification-URI			x			o		o			
Client Notification Content Preference	Client-Notification-Content-Preferred			x			o		o			
Client Brand Information	Client-Brand-Logging-Information			x			o		o			
API Contract ID	API-Contract-ID			x			c		c			
Account Information Category		x					m		m		m	
Preference on application layer encryption	encryptionSupported				x		o					

Data element	Attribute encoding	Location					Usage					
		Path	Query Param.	Header	Body	Certificate	Initiate Subscr. Req.	Initiate Subsrc. Resp.	Add Subscr. Req.	Add Subscr. Resp.	Status Req.	Status Resp.
Encryption certificate	encryptionCertificate				x		o					
Subscription Identification	subcriptionId	x							m		m	
Subscription Entries							m		m			
Account Reference	subscriptionEntries.account				x		m		m			
Validity	subscriptionEntries.validUntil				x		o		o			
Subscription Entry Name	subcriptionEntries.subscriptionEntryName				x		o		o			
Account Information Category Parameters	subscriptionEntries.accountInformationCategoryParameters				x		m		m			
Trigger Criteria	subscriptionEntries.accountInformationCategoryParameters.accountInformationCategoryCriteria				x		m		m			
Additional Parameter	all parameters in addition to accountInformationCageroyCriteria within subscriptionEntries.accountInformationCategoryParameters				x		c		c			
Primary Push URI	subsriptionEntry.apiClientPrimaryPushURI				x		m		m			
Secondary Push URI	subsriptionEntry.apiClientSecondaryPushURI				x		o		o			

PSU IP Address/Port and Further PSU related Information

The above table addresses several PSU related context data. These data, its importance and its usage are defined in detail in [oFA-PFSM]. They are not mentioned anymore in the following detailed definitions for matter of better readability, as long as the usage is not mandated.

3.3 Initiate Subscription Request

3.3.1 Generic Request and Response

The Initiate Subscription Request will be on the one hand uniform in the level of data provisioning, which is why this is specified in the following as "Generic Initiate Subscription Request". On the other hand, the structure of the selection and content criteria are quite different. So, some extensive examples are provided in the following sub sections for the different subservices.

This request is a Transaction Initiation Message as introduced in [oFA-PFSM], i.e. all related conditions on (additional) parameters e.g. on PSU identification or on authorisation apply. The related parameters and conditions are not further mentioned below.

Call

```
POST /subscriptions/{push-accountinformation-subservice}
```

Creates a corresponding subscription resource in the ASPSP server to start a Push Account Information Service on the addressed account to a push URI provided by the API Client.

NOTE: The subscription resource may contain several subscription entries. Every subscription entry where the related subscription entry (trigger) criteria are met will lead to the related push activity. If several subscription entries meet the (trigger) criteria following e.g. an account entry, **all** of the related push activities are performed even if this implies potentially a double processing.

NOTE: The request will not be authorised successfully, if there is already a subscription resource authorised successfully for the addressed sub-service by the same PSU (in a retail banking context) or by the same corporate and being still valid or partially authorised. In such a case the related entries need to be added to the related subscription.

Path Parameters

Attribute	Type	Condition	Description
push-accountinformation-subservice	String	Mandatory	The addressed subservice endpoint, e.g. for push-account-entries. The default list of sub services supported in this standard is: • push-account-entries • push-account-statements • push-balances • push-requests-to-pay

Attribute	Type	Condition	Description
			The ASPSP will publish which of the subservices will be supported.

Query Parameters

No specific requirements

Request Header

Attribute	Type	Condition	Description
X-Request-ID	UUID	Mandatory	Unique identification of the request.
PSU-IP-Address	String	Mandatory	The forwarded IP Address header field consists of the corresponding HTTP request IP Address field between PSU and TPP, or API Client IP-Address in case of a direct access. If not available, the TPP shall use the IP Address used by the TPP when submitting this request.
Authorization	String	Conditional	Is contained only if Oauth2 has been used in a pre-step for onboarding the API Client
API-Contract-ID	UUID	Conditional	ID of the underlying service contract between API Client and ASPSP, resulting from API Client onboarding, following [oFA-IG-ADM]. Only applies where a contract is mandated by the ASPSP.

Request Body

Attribute	Type	Condition	Description
subscriptionEntries	Array of Subscription Entry	Mandatory	This is the non-empty array of subscription entries. Each subscriptionEntry will be stored in a sub-resource, which might be referenced later by the API client directly. **Remark:** If the balanceAmount ist provided with a currency that is not supported for this account, the ASPSP will reject the request.

Attribute	Type	Condition	Description
encryptionSupported	Boolean	Optional	If the flag is set to false, no encryption is provided by the ASPSP. If the flag is set to true, the ASPSP can choose to support encryption.
encryptionCertificate	String	Optional	The certificate to be used for encryption by the ASPSP in base64 encoding. If not provided, no encryption will be provided.

RULE: There can be multiple subscription entries for one account-id.

RULE: If the data in one of the contained subscription entries is inconsistent, the whole subscription needs to be rejected. A partial acceptance is not supported.

Response Code

The HTTP response code equals 201.

Response Header

Attribute	Type	Condition	Description
ASPSP-Corporate	Boolean	Optional	This is a Boolean telling the API Client whether the addressed account is a corporate account (true) or not (false). This information might also be provided after the successful authorisation of the addressed subscription in a dedicated resource attribute.
Location	String	Mandatory	Location of the created resource (if created)
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
subscriptionId	Max70Text	Mandatory	Identification of the generated subscription.
subscriptionStatus	Subscription Status	Mandatory	This is the status of the subscription.

Attribute	Type	Condition	Description
_links	Links	Mandatory	A list of hyperlinks to be recognised by the API Client. The actual hyperlinks used in the response depend on the dynamical decisions of the ASPSP when processing the request. The potential links for this response message are generically defined in [oFA-PFSM] for all Transaction Initiation Response messages. These links will also be contained in the related Open API files. **Remark**: All links can be relative or full links, to be decided by the ASPSP.

3.3.2 Example Push Account Entry Service

Example

With the following request, the TPP submits a subscription for the Push Account Entry Service. The subscription contains one subscription entry: With this entry, the TPP asks to be informed via a callback each time a card transaction with amount 100 € or higher appears on the account (with IBAN DE40100100103307118608). The push notification shall contain the text "debit alarm" and should contain the data elements "transactionAmpount" and "creditorName" from the respective card transaction.

Request

```
POST https://api.testbank.com/v2/subscriptions/push-account-entries
X-Request-ID:           99391c7e-ad88-49ec-a2ad-99ddcb1f7719
API-Contract-ID:        99391c7e-ad88-49ec-a2ad-888888aaaaa1
Date:                   Sun, 06 Aug 2017 15:03:46 GMT
{
  "subscriptionEntries":
   [
     {"accountId": {"iban": "DE40100100103307118608"},
      "subscriptionEntryName": "Card Debit-Alarm",
      "apiClientPrimaryPushURI": "www.example-TPP.com/openFinance-
client/v2/ASPSPid/psu-account-id/entries",
      "responseWithLinkPreferred": false,
      "responseWithStaticTextPreferred": true,
      "staticCallbackText": "debit-alarm", /* in case of high value debit
      "pushAccountEntryParameters":
       {
        "accountEntryCriteria": {
            "bankTransactionCodePatterns": ["PMNT-CCRD-????"], /* all card
transactions
```

```
            "minimumAmount": {"currency": "EUR", "amount": "100"} /*would
only forward information about entries starting from 100 Euro
        },
        "acceptedFormat": "application/json",
        "preferredAttributes": ["transactionAmount", "creditorName"],
        "documentsPreferred": true /* forwards also e.g. e-receipts related
to the card transaction if supported by the ASPSP
       }
   }
  ],
  "encryptionSupported": false
}
```

Response

```
HTTP/1.x 201 Ok
ASPSP-Corporate:        false
X-Request-ID:           99391c7e-ad88-49ec-a2ad-99ddcb1f7719
Date:                   Sun, 06 Aug 2017 15:03:47 GMT
Content-Type:           application/json

{
  "subscriptionId": "3d9a81b3-a47d-4130-8765-a9c0ff861103",
  "subscriptionStatus": "received",
  "_links":
   {
    "scaRedirect": {"href": "https://www.testbank.com/asdfasdfasdf"},
    "self": {"href": "/v2/subscriptions/push-account-entries/3d9a81b3-a47d-4130-
8765-a9c0ff861103"},
    "status": {"href": "/v2/subscriptions/push-account-entries/3d9a81b3-a47d-
4130-8765-a9c0ff861103/status"},
    "scaStatus": {"href": "/ v2/subscriptions/push-account-entries/1234-wertiq-
                           983/authorisations/123auth456"}
   }
}
```

3.3.3 Example Push Account Statement Subservice

For this subservice, only the subscription entry is provided as example as part of the request body:

With this entry, the TPP asks to be provided with each account statement of "camt.053" format for the account with IBAN DE40100100103307118608 as soon as it is created within the ASPSP's system. The callback shall contain the data directly (not with a download link). A specific text for the callback is not defined.

```
"subscriptionEntries":
```

```
    [
     {
      "accountId": {"iban": "DE40100100103307118608"},
      "subscriptionEntryName": "Daily Statement Delivery",
      "apiClientPrimaryPushURI": "www.example-TPP.com/xs2a-
client/v2/ASPSPid/psu-account-id/account-statements",
      "responseWithLinkPreferred": false,
      "responseWithStaticTextPreferred": false,
      "pushAccountStatementParameters":
        {
         "accountStatementCriteria": {
             "statements": "camt.053",
             "event": "afterGeneration"
             }
        }
      }
    ]
```

3.3.4 Example Push Balance Information Subservice

For this subservice, only the subscription entry is provided as example as part of the request body:

With this entry, the TPP asks to be informed via a callback, whenever the balance of type "interimAvailble" for the account with IBAN DE40100100103307118608 changes from being 0 € or more to being less than 0 €. The callback shall contain the data directly (not with a download link). A specific text for the callback is not defined.

```
"subscriptionEntries":
    [
     {
      "accountId": {"iban": "DE40100100103307118608"},
      "subscriptionEntryName": "balance alarm",
      "apiClientPrimaryPushURI": "www.example-TPP.com/xs2a-client/v2/ASPSPid/psu-account-id/entries",
      "responseWithLinkPreferred": false,
      "responseWithStaticTextPreferred": false,
      "pushBalanceParameters":
        {
         "balanceCriteria": {
             "balanceAmount": {"currency": "EUR", "amount": "0.0"},
             "balanceOperator": "less",
             "balanceType": "interimAvailable"}
        }
      }
  ]
```

3.3.5 Example Push RTP Information Subservice

For this subservice, only the subscription entry is provided as example as part of the request body:

With this entry, the TPP asks to be informed via a callback, whenever specific Requests for Payment are received for the account with IBAN DE40100100103307118608. Not every Request for Payment shall trigger a callback, only those that have an expiration time of 30 minutes or less. The callback shall be sent directly after the receipt of the corresponding Request to Pay The callback shall contain the data directly (not with a download link). A specific text for the callback is not defined.

```
"subscriptionEntries":
   [
    {
     "accountId": {"iban": "DE40100100103307118608"},
     "subscriptionEntryName": "Forward urgent RTPs",
     "apiClientPrimaryPushURI": "www.example-TPP.com/xs2a-
client/v2/ASPSPid/psu-account-id/rtps",
     "responseWithLinkPreferred": true,
     "responseWithStaticTextPreferred": false,
     "pushRtpParameters":
       {
        "rtpCriteria": {
            "event": "afterReception",
            "timeToExpire": 30 /* only forward incoming rtp with an expiry
timestamp in a period of 30 minutes
         }
        "acceptedFormat": "application/JSON",
        "documentsPreferred": true
       }
   ]
```

3.4 Read Subscriptions

Call

```
GET /v2/subscriptions/{push-accountinformation-subservice}
```

Reads all subscriptions related to a given PSU and a given account information sub-service.

Path Parameters

Attribute	Type		Description
push-accountinformation-subservice	String	Mandatory	The addressed subservice endpoint, e.g. for push-account-entries. The default list of sub services supported in this standard is: • push-account-entries • push-account-statements • push-balances • push-requests-to-pay, The ASPSP will publish which of the subservices will be supported.

Query Parameters

Attribute	Type	Condition	Description
subscriptionStatus	String	Optional	The status of the subscription to be retrieved, eg. subscriptionStatus=valid.

Request Headers

Attribute	Type	Condition	Description
X-Request-ID	UUID	Mandatory	ID of the request, unique to the call, as determined by the initiating party.
Authorization	String	Conditional	Is contained if OAuth2 has been used as SCA channel for submitting the related subscription or if the OAuth2 Prestep has been used for registrating the API Client.
PSU-ID	Max140Text	Conditional	Client ID of the PSU in the ASPSP client interface. Is mandated, • if no related Authorization identifying the PSU or • if the PSU is not identified by the certificates used for Client identification in the corporate direct access case. It might be contained even if an OAuth2 based authentication was performed in a pre-step. In this

Attribute	Type	Condition	Description
			case the ASPSP might check whether PSU-ID and token match, according to ASPSP documentation.
PSU-ID-Type	Max35Text	Conditional	Type of the PSU-ID; needed in scenarios where PSUs have several PSU-IDs as access possibility. In this case, the mean and use are then defined in the ASPSP's documentation.
PSU-Corporate-ID	Max140Text	Conditional	Identification of a Corporate in the Online Channels Might be mandated in the ASPSP's documentation. Only used in a corporate context. It is not used if the corporate identification is provided by the API Client identification via certificates in the corporate direct access case.
PSU-Corporate-ID-Type	Max35Text	Conditional	This is describing the type of the identification needed by the ASPSP to identify the PSU-Corporate-ID content as used in online channels. Typically, this is a proprietary definition. Mean and use is defined in the ASPSP's documentation. Only used in a corporate context.

Request Body

No request 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
subscriptions	Array of Subscription Identification and Link	Mandatory	The subscriptionIds addressed by the related query together with status information.

Example

Request

```
GET https://api.testbank.com/v2/subscriptions/push-account-entries
PSU-ID:           123456
X-Request-ID:     99391c7e-ad88-49ec-a2ad-99ddcb1f7721
Date:             Sun, 06 Aug 2017 15:05:46 GMT
```

Response

```
HTTP/1.x 200 Ok
X-Request-ID:           99391c7e-ad88-49ec-a2ad-99ddcb1f7721
Date:                   Sun, 06 Aug 2017 15:05:47 GMT
Content-Type:           application/json

{"subscriptions": [
      {"subscriptionId": ""3d9a81b3-a47d-4130-8765-a9c0ff861103",
       "subscriptionStatus": "valid",
       "_links":
            {"subscription": {"href": "/v2/subscriptions/push-account-
entries/1234-wertiq-9833d9a81b3-a47d-4130-8765-a9c0ff861103"}
      }
  ]
```

3.5 Read Subscription Details

Call

```
GET /v2/subscriptions/{push-accountinformation-subservice}/{subscriptionId}
```

Reads the details (including subscriptionEntryIds and status) of the subscription resource.

Path Parameters

Attribute	Type		Description
push-accountinformation-subservice	String	Mandatory	The addressed subservice endpoint, e.g. for push-account-entries. The default list of sub services supported in this standard is: • push-account-entries • push-account-statements • push-balances • push-requests-to-pay, The ASPSP will publish which of the subservices will be supported.
subscriptionId	Max70Text	Mandatory	The ID of the subscription object to be retrieved

Query Parameters

No specific query parameter.

Request Headers

Attribute	Type	Condition	Description
X-Request-ID	UUID	Mandatory	ID of the request, unique to the call, as determined by the initiating party.
Authorization	String	Conditional	Is contained if OAuth2 has been used as SCA channel for submitting the related subscription or if the OAuth2 Prestep has been used for registrating the API Client.

Request Body

No request 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
subscriptionStatus	Subscription Status	Mandatory	This is the status of the subscription.
corporateAccountFlag	Boolean	Optional	This is a Boolean telling the API Client whether the addressed accounts are a corporate account (true) or not (false). Should be provided after successful authorisation.
subscriptionEntries	Array of Subscription Entry	Mandatory	This is the collection of subscription entries.
encryptionSupported	Boolean	Optional	If the flag is set to false, no encryption is provided by the ASPSP. If the flag is set to true, the ASPSP can choose to support encryption.
encryptionCertificate	String	Optional	The certificate to be used for encryption by the ASPSP in base64 encoding. If not provided, no encryption will be provided.

Example

Request

```
GET https://api.testbank.com/v2/subscriptions/push-account-
entries/3d9a81b3-a47d-4130-8765-a9c0ff861103
X-Request-ID:           99391c7e-ad88-49ec-a2ad-99ddcb1f7721
Date:                   Sun, 06 Aug 2017 15:05:46 GMT
```

Response

```
HTTP/1.x 200 Ok
X-Request-ID:           99391c7e-ad88-49ec-a2ad-99ddcb1f7721
Date:                   Sun, 06 Aug 2017 15:05:47 GMT
Content-Type:           application/json

{
  "subscriptionEntries": [
    {
     "accountId": {"iban": "DE40100100103307118608"},
```

```
      "subscriptionEntryName": "description 1",
      "subscriptionEntryId": "3D9A81B3-A47D-4130-8765-A9C0FF867777",
      "_links":
        {
        "entry": {"href": " /v2/subscriptions/push-account-entries/1234-
                           wertiq-983/entries/3d9a81b3-a47d-4130-8765-
a9c0ff867777"}
        }
      },
      {
       "accountId": {"iban": "DE40345100103307118789"},
       "subscriptionEntryName": "description 2",
       "subscriptionEntryId": "3D9A81B3-A47D-4130-8765-A9C0FF867778",
       "_links":
        {
        "entry": {"href": "/v2/subscriptions/push-account-entries/3d9a81b3-
a47d-4130-8765-a9c0ff861103/entries/3D9A81B3-A47D-4130-8765-A9C0FF867778"}
        }
      },
  ],
  "encryptionSupported": false,
  "subscriptionStatus": "received"
}
```


3.6 Read Subscription Status

Call

`GET /v2/subscriptions/{push-accountinformation-subservice}/{subcriptionId}/status`

Reads the status of the subscription resource.

Path Parameters

Attribute	Type		Description
push-accountinformation-subservice	String	Mandatory	The addressed sub service endpoint, e.g. for push-account-entries. The default list of sub services supported in this standard is: • push-account-entries • push-account-statements • push-balances • push-requests-to-pay

Attribute	Type		Description
			The ASPSP will publish which of the subservices will be supported.
subscriptionId	Max70Text	Mandatory	The ID of the subscription object to be retrieved

Query Parameters

No specific query parameters defined.

Request Header

Attribute	Type	Condition	Description
X-Request-ID	UUID	Mandatory	ID of the request, unique to the call, as determined by the initiating party.
Authorization	String	Conditional	Is contained if OAuth2 has been used as SCA channel for submitting the related subscription or if the OAuth2 Prestep has been used for registrating the API Client.

Request Body

No request body.

Response Code

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
subscriptionStatus	Subscription Status	Mandatory	This is the status of the subscription.

Attribute	Type	Condition	Description
_links	Links	Optional	This a list of hyperlinks of type "entryRevokedStatus" to the status endpoints of entries where the subscriptionEntryStatus equals revokedByPsu.

Example

Request

```
GET https://api.testbank.com/v2/subscriptions/push-account-
entries/3d9a81b3-a47d-4130-8765-a9c0ff861103/status
X-Request-ID:           99391c7e-ad88-49ec-a2ad-99ddcb1f7721
Date:                   Sun, 06 Aug 2017 15:05:46 GMT
```

Response

```
HTTP/1.x 200 Ok
X-Request-ID:           99391c7e-ad88-49ec-a2ad-99ddcb1f7721
Date:                   Sun, 06 Aug 2017 15:05:47 GMT
Content-Type:           application/json

{
  "subscriptionStatus": "received",
  "_links":
      {"entryStatusRevoked":
            [{"href": "…/status"}]
      }
}
```

3.7 Add a Subscription Entry

Call

```
POST /subscriptions/{push-accountinformation-subservice}/{subscriptionId}/
subscription-entries
```

Creates and adds subscription entry sub-resource to an existing subscription.

Please note that this call is a Transaction Initiation Request in the sense of [oFA-PFSM], and all related http headers as defined there are applicable for request and response messages.

Path Parameters

Attribute	Type	Condition	Description
push-accountinformation-subservice	String	Mandatory	The addressed sub service endpoint, e.g. for push-account-entries. The default list of sub services supported in this standard is: • push-account-entries • push-account-statements • push-balances • push-requests-to-pay, The ASPSP will publish which of the subservices will be supported. **Note:** For this version, this specification only supports the service referred to by "push-account-entries".
subscriptionId	Max70Text	Mandatory	The subscription object to be extended

Query Parameters

No specific requirements

Request Header

Attribute	Type	Condition	Description
X-Request-ID	UUID	Mandatory	Unique identification of the request.
Authorization	String	Conditional	Is contained if OAuth2 has been used as SCA channel for submitting the related subscription or if the OAuth2 Prestep has been used for registrating the API Client.
PSU-IP-Address	String	Mandatory	The forwarded IP Address header field consists of the corresponding HTTP request IP Address field between PSU and TPP, or API Client IP-Address in case of a direct access. If not available in a TPP scenario, the TPP shall use the IP Address used by the TPP when submitting this request.

NOTE: The request does not support the registration of a resource status notification service. The support of this lean push service is inherited from the master subscription resource. This also applies for updating SCA status from the new authorisation resources.

Request Body

Attribute	Type	Condition	Description
accountId	Account Reference	Mandatory	This is the account from which related entries shall be pushed.
validUntil	ISODate	Optional	Requested validity period of the service.
subscriptionEntryName	Max35Text	Optional	Name of the subscription entry
apiClientPrimaryPush URI	Max256Text	Mandatory	The URI where related account information shall be pushed to.
apiClientSecondaryPush URI	Max256Text	[Optional if supported]	The URI where related account information shall be pushed to if the primary push URI is not reachable. If this attribute is used by the API Client and the ASPSP is not supporting this feature, then the request will be rejected with a dedicated additional error information.
callbackWithLinkPreferred	Boolean	Optional	API Client prefers to receive hyperlinks pointing to the related account information element if the related subservice criteria are met.
callbackWithStaticTextPreferred	Boolean	Optional	API Client prefers to get informed by static text if the related subservice criteria are met.
staticCallbackText	Max140Text	Optional	The text to be provided by the ASPSP in case static text is sent if the related subservice criteria are met.
pushAccountEntryParameters	Push Account	{Or	Parameters for a subscription entry for the Push Account Entry Service.

Attribute	Type	Condition	Description
	Entry Parameters		
pushAccount StatementParameters	Push Account Statement Parameters	Or	Parameters for a subscription entry for the Push Account Statement Service.
pushBalanceParameters	Push Balance Parameters	Or	Parameters for a subscription entry for the Push Balance Information Service.
pushRtpParameters	Push RTP Parameters	Or}	Parameters for a subscription entry for the Push RTP Information Service.

Response Code

The HTTP response code equals 201.

Response Header

Attribute	Type	Condition	Description
Location	String	Mandatory	Location of the created resource (if created)
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
subscriptionEntryId	Max70Text	Mandatory	Identification of the created subscription entry within the related subscription.
subscriptionEntry Status	Subscription Entry Status	Mandatory	This is the status of the subscription entry.
_links	Links	Mandatory	A list of hyperlinks to be recognised by the API Client. The actual hyperlinks used in the response depend on the dynamical decisions of the ASPSP when processing the request. The potential links for this response message are generically defined in [oFA-PFSM] for all

Attribute	Type	Condition	Description
			Transaction Initiation Response messages. These links will also be contained in the related Open API files. **Remark**: All links can be relative or full links, to be decided by the ASPSP.

Example

Request

```
POST https://api.testbank.com/v2/subscriptions/push-account-
entries/3d9a81b3-a47d-4130-8765-a9c0ff861103/subcription-entries
X-Request-ID:           99391c7e-ad88-49ec-a2ad-99ddcb1f7719
Date:                   Sun, 06 Aug 2017 15:03:46 GMT
{

     "accountId": {"iban": "DE40100100103307118608"},
     "subscriptionEntryName": "description 1",
      "apiClientPrimaryPushURI": "www.example-TPP.com/openFinance-
client/v2/ASPSPid/psu-account-id/entries",
      "responseWithLinkPreferred": false,
      "responseWithStaticTextPreferred": true,
      "staticCallbackText": "high value alarm", /* in case of high value
transactions (debit and credit)
      "pushAccountEntryParameters":
       {
        "accountEntryCriteria": {
            "minimumAmount": {"currency": "EUR", "amount": "300"} /would
only forward information about entries starting with 300 Euro
        },
        "acceptedFormat": "application/json",
        "preferredAttributes": ["transactionAmount", "creditorName"],
        "documentsPreferred": true /* forwards also e.g. e-receipts related
to the card transaction if supported by the ASPSP
   }

}
```

Response

```
HTTP/1.x 201 Ok
X-Request-ID:           99391c7e-ad88-49ec-a2ad-99ddcb1f7719
Date:                   Sun, 06 Aug 2017 15:03:47 GMT
Content-Type:           application/json
```

```
{
  "subscriptionId": "3d9a81b3-a47d-4130-8765-a9c0ff861103",
  "subscriptionEntryId": "3D9A81B3-A47D-4130-8765-A9C0FF867778",
  "subscriptionEntryStatus": "received",
  "_links":
   {
    "scaRedirect": {"href": "https://www.testbank.com/asdfasdfasdf"},
    "self": {"href": "/v2/subscriptions/push-account-entries/
3d9a81b3-a47d-4130-8765-a9c0ff861103/subscription-entries/3D9A81B3-A47D-4130-8765-
A9C0FF867778"},
    "status": {"href": "/v2/subscriptions/push-account-entries/
3d9a81b3-a47d-4130-8765-a9c0ff861103/subscription-entries/3D9A81B3-A47D-4130-8765-
A9C0FF867778/status"},
    "scaStatus": {"href": "/v2/subscriptions/push-account-entries/
3d9a81b3-a47d-4130-8765-a9c0ff861103/subscription-entries/3D9A81B3-A47D-4130-8765-
A9C0FF867778/authorisations/123auth456"}
   }
}
```


3.8 Read Subscription Entry Details

Call

```
GET /v2/ subscriptions/{push-accountinformation-
subservice}/{subscriptionId}/subscription-entries/{subscriptionEntryId}
```

Path Parameters

Attribute	Type	Condition	Description
push-accountinformation-subservice	String	Mandatory	The addressed sub service endpoint, e.g. for push-account-entries. The default list of sub services supported in this standard is: • push-account-entries • push-account-statements • push-balances • push-requests-to-pay, The ASPSP will publish which of the subservices will be supported. **Note:** For this version, this specification only supports the service referred to by push-account-entries.

Attribute	Type	Condition	Description
subscriptionId	Max70Text	Mandatory	Identification of the addressed subscription.
subscriptionEntryId	Max70Text	Mandatory	Identification of the created subscription entry itself.

Query Parameters

No specific query parameter.

Request Headers

Attribute	Type	Condition	Description
X-Request-ID	UUID	Mandatory	ID of the request, unique to the call, as determined by the initiating party.
Authorization	String	Conditional	Is contained if OAuth2 has been used as SCA channel for submitting the related subscription or if the OAuth2 Prestep has been used for registrating the API Client.

Request Body

No request 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
subscriptionEntryStatus	Subscription Entry Status	Mandatory	The status of the subscription entry resource itself.
accountId	Account Reference	Mandatory	This is the account from which related entries shall be pushed.

Attribute	Type	Condition	Description
validUntil	ISODate	Optional	Requested validity period of the service.
subscriptionEntryName	Max35Text	Optional	Name of the subscription entry
subscriptionEntryId	Max70Text	Mandatory	Identification of the subscription as provided by the ASPSP
apiClientPrimaryPush URI	Max256Text	Mandatory	The URI where related account information shall be pushed to.
apiClientSecondaryPush URI	Max256Text	[Optional if supported]	The URI where related account information shall be pushed to if the primary push URI is not reachable. If this attribute is used by the API Client and the ASPSP is not supporting this feature, then the request will be rejected with a dedicated additional error information.
callbackWithLinkPreferred	Boolean	Optional	API Client prefers to receive hyperlinks pointing to the related account information element if the related subservice criteria are met.
callbackWithStaticTextPreferred	Boolean	Optional	API Client prefers to get informed by static text if the related subservice criteria are met.
staticCallbackText	Max140Text	Optional	The text to be provided by the ASPSP in case static text is sent if the related subservice criteria are met.
pushAccountEntryParameters	Push Account Entry Parameters	{Or	Parameters for a subscription entry for the Push Account Entry Service.
pushAccount StatementParameters	Push Account Statement Parameters	Or	Parameters for a subscription entry for the Push Account Statement Service.

Attribute	Type	Condition	Description
pushBalanceParameters	Push Balance Parameters	Or	Parameters for a subscription entry for the Push Balance Information Service.
pushRtpParameters	Push RTP Parameters	Or}	Parameters for a subscription entry for the Push RTP Information Service.

Example

Request

```
GET https://api.testbank.com/v2/subscriptions/push-account-
entries/3d9a81b3-a47d-4130-8765-a9c0ff861103/entries/3d9a81b3-a47d-4130-
8765-a9c0ff867777
X-Request-ID:           99391c7e-ad88-49ec-a2ad-99ddcb1f7721
Date:                   Sun, 06 Aug 2017 15:05:46 GMT
```

Response

```
HTTP/1.x 200 Ok
X-Request-ID:           99391c7e-ad88-49ec-a2ad-99ddcb1f7721
Date:                   Sun, 06 Aug 2017 15:05:47 GMT
Content-Type:           application/json
```

```
{
  "subscriptionEntryStatus": "received",
  "accountId": {"iban": "DE40100100103307118608"},
  "subscriptionEntryId": "3D9A81B3-A47D-4130-8765-A9C0FF867777",
  "subscriptionEntryName": "Card Debit-Alarm",
  "apiClientPrimaryPushURI": "www.example-TPP.com/openFinance-
client/v2/ASPSPid/psu-account-id/entries",
  "responseWithLinkPreferred": false,
  "responseWithStaticTextPreferred": true,
  "staticCallbackText": "debit-alarm",
  "pushAccountEntryParameters":
    {
     "accountEntryCriteria": {
       "bankTransactionCodePatterns": ["PMNT-CCRD-*"],
       "minimumAmount": {"currency": "EUR", "amount": "100"}
    },
  "acceptedFormat": "application/json",
  "preferredAttributes": ["transactionAmount", "creditorName"],
  "documentsPreferred": true
```

```
}
```

3.9 Read Subscription Entry Status

Call

```
GET /v2/subscriptions/{push-accountinformation-
subservice}/{subscriptionId}/subscription-
entries/{subscriptionEntryId}/status
```

Path Parameters

Attribute	Type	Condition	Description
push-accountinformation-subservice	String	Mandatory	The addressed sub service endpoint, e.g. for push-account-entries. The default list of sub services supported in this standard is: • push-account-entries • push-account-statements • push-balances • push-requests-to-pay The ASPSP will publish which of the subservices will be supported.
subscriptionId	Max70Text	Mandatory	Identification of the addressed subscription.
subscriptionEntryId	Max70Text	Mandatory	Identification of the subscription entry itself.

Query Parameters

No specific query parameter.

Request Headers

Attribute	Type	Condition	Description
X-Request-ID	UUID	Mandatory	ID of the request, unique to the call, as determined by the initiating party.
Authorization	String	Conditional	Is contained if OAuth2 has been used as SCA channel for submitting the related subscription or if the OAuth2 Prestep has been used for registrating the API Client.

Request Body

No request 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
subscriptionEntryStatus	Subscription Status	Mandatory	The status of the subscription entry resource itself.

Example

Request

```
GET https://api.testbank.com/v2/subscriptions/push-account-
entries/3d9a81b3-a47d-4130-8765-a9c0ff861103/entries/3d9a81b3-a47d-4130-
8765-a9c0ff867777
X-Request-ID:           99391c7e-ad88-49ec-a2ad-99ddcb1f7721
Date:                   Sun, 06 Aug 2017 15:05:46 GMT
```

Response

```
HTTP/1.x 200 Ok
X-Request-ID:           99391c7e-ad88-49ec-a2ad-99ddcb1f7721
Date:                   Sun, 06 Aug 2017 15:05:47 GMT
Content-Type:           application/json

{
  "subscriptionEntryStatus": "received"
}
```

3.10 Cancel Subscription Request

The TPP can cancel a push account information subscription if needed with the following call:

Call

```
DELETE /v2/subscriptions/{push-accountinformation-
subservice}/{subscriptionId}
```

Deletes a given subscription resource.

Path Parameters

Attribute	Type	Condition	Description
push-accountinformation-subservice	String	Mandatory	The addressed sub service endpoint, e.g. for push-account-enties. The default list of sub services supported in this standard is: • push-account-entries • push-account-statements • push-balances • push-requests-to-pay, The ASPSP will publish which of the subservices will be supported.
subscriptionId	Max70Text	Mandatory	The ID of the subscription to be cancelled.

Query Parameters

No specific query parameters.

Request Header

Attribute	Type	Condition	Description
X-Request-ID	UUID	Mandatory	ID of the request, unique to the call, as determined by the initiating party.
Authorization	String	Conditional	Is contained if OAuth2 has been used as SCA channel for submitting the related subscription or if the OAuth2 Prestep has been used for registrating the API Client.

Request Body

No Request Body.

Response Code

The HTTP response code 204 is used if a subscription was deleted successfully.

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

No Response Body

Example

Request

```
DELETE https://api.testbank.com/v2/subscriptions/push-account-
entries/3d9a81b3-a47d-4130-8765-a9c0ff861103
X-Request-ID            99391c7e-ad88-49ec-a2ad-99ddcb1f7757
Date                    Sun, 13 Aug 2017 17:05:37 GMT
```

Response

```
HTTP/1.x 204 No Content
X-Request-ID:           99391c7e-ad88-49ec-a2ad-99ddcb1f7757
Date:                   Sun, 13 Aug 2017 17:05:47 GMT
```

3.11 De-activate Subscription Entry Request

The TPP can delete a push account information entry subscription if needed with the following call:

Call

```
DELETE /v2/subscriptions/push-account-entries/{subscriptionId}/
subscription-entries/{subcriptionEntryId}
```

Deactivates a given subscription entry resource.

Path Parameters

Attribute	Type	Condition	Description
push-accountinformation-subservice	String	Mandatory	The addressed sub service endpoint, e.g. for push-account-entries. The default list of sub services supported in this standard is: • push-account-entries • push-account-statements • push-balances • push-requests-to-pay The ASPSP will publish which of the subservices will be supported.
subscriptionId	Max70Text	Mandatory	Identification of the addressed subscription.
subscriptionEntryId	Max70Text	Mandatory	Identification of the created subscription entry itself.

Query Parameters

No specific query parameters.

Request Header

Attribute	Type	Condition	Description
X-Request-ID	UUID	Mandatory	ID of the request, unique to the call, as determined by the initiating party.
Authorization	String	Conditional	Is contained if OAuth2 has been used as SCA channel for submitting the related subscription or if the OAuth2 Prestep has been used for registrating the API Client.

Request Body

No Request Body.

Response Code

The HTTP response code 204 is used if a subscription was deleted successfully.

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

No Response Body

Example

Request

```
DELETE https://api.testbank.com/v2/subscriptions/push-account-
entries/3d9a81b3-a47d-4130-8765-a9c0ff861103
X-Request-ID            99391c7e-ad88-49ec-a2ad-99ddcb1f7757
Date                    Sun, 13 Aug 2017 17:05:37 GMT
```

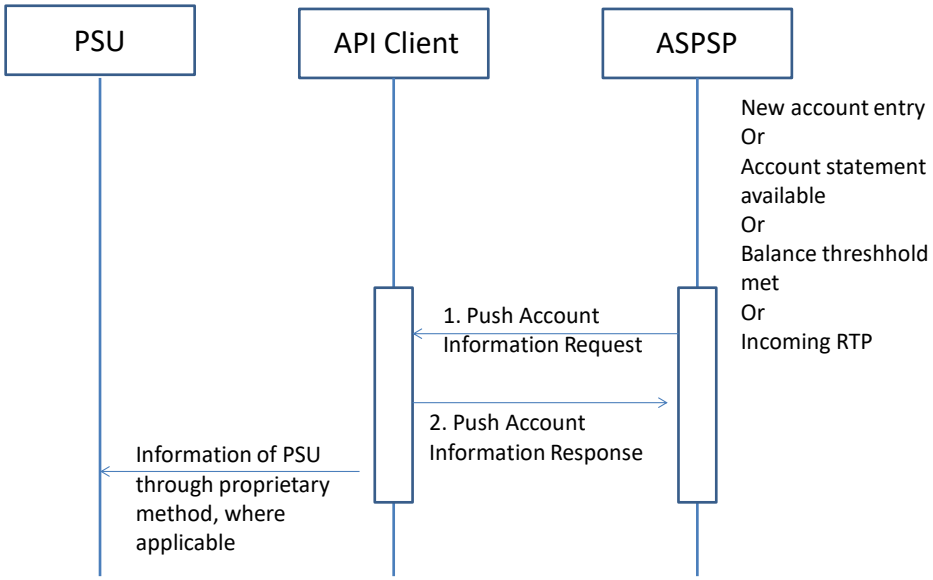
Response

```
HTTP/1.x 204 No Content
X-Request-ID:           99391c7e-ad88-49ec-a2ad-99ddcb1f7757
Date:                   Sun, 13 Aug 2017 17:05:47 GMT
```

4 Push Account Information

4.1 Push Account Information Message Flows

The following flow shows the simple request and response flow for an account information push service:



.

4.2 Push Account Information Messages

Call

```
POST <apiClientPrimaryPushURI>
```

Posts account information in JSON encoding to the primary URI of the API Client.

Path Parameters

No Path Parameter

Query Parameters

No Query Parameter

Request Header

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 for Push Account Entries (JSON)

Attribute	Type	Condition	Description
account	Account Reference	Mandatory	
debitAccounting	Boolean	Optional	If true, the amounts of debits on the reports are quoted positive with the related consequence for balances. If false, the amount of debits on the reports are quoted negative.
dateTimeLastPush	ISODateTime	Optional	
staticCallbackText	Max140	Optional	
_links	Links	Optional	
transactions	Lean Account Report	Optional	

Note: This service does not support the pushing of the technical cardTransactions formats. Card transactions can still be reported as specific entries on a regular payment account (e.g. in case of debit cards).

Request Body for Push Account Entries (XML)

This is transported via camt.05x messages, as defined by the ASPSP.

Request Body for Push Balance Information

Attribute	Type	Condition	Description
account	Account Reference	Mandatory	
balance	Balance	Optional	
staticCallbackText	Max140	Optional	
_links	Links	Optional	
dateTimeLastPush	ISODateTime	Optional	

Request Body for Account Statements (JSON)

Account Statements as defined in [oFA-AS] will be pushed:

Attribute	Type	Condition	Description
messageId	Max35Text	Mandatory	Point to point reference, as assigned by the sending party, to unambiguously identify the batch of transactions.
creationDateTime	ISODateTIme	Mandatory	Date and time at which the message was created.
messageRecipient	Party Identification	Optional	Party authorised by the account owner to receive information about movements on the account.
pageNumber	Integer	Optional	Page number.
lastPageIndicator	Boolean	Optional	Indicates the last page.
statement	Array of Statement	Mandatory	**Usage Rule:** Only the statement for one account can be pushed, so only one statement element will be contained in this array.

Request Body for Account Statements (XML)

This is transported via camt.05x messages, as defined by the ASPSP.

Request Body for Account Statements (MT version)

This is transported via MT94x messages, as defined by the ASPSP.

Request Body for RTP Information (JSON)

Attribute	Type	Condition	Description
dateTimeLastPush	ISODateTime	Optional	
account	Account Reference	Mandatory	
staticCallbackText	Max140	Optional	
_links	Links	Optional	

Attribute	Type	Condition	Description
transactions	Lean Account Report	Optional	Incoming RTPs are reported as transactions of type "information" and with a mandatory subfield "additionalInformationStructured" as well as either a related Bank Transaction Code or proprietary Bank Transaction Code.

Request Body for RTP-information (XML)

in pain.013 formats.

HTTP Response Code

204

Note: If the response code does not equal 204, then the ASPSP will push the related information to the secondary URI if registered during subscription. If a secondary URI has not been registered by the API Client, then the notification will not be repeated. The ASPSP may repeat each of these notifications.

Response Header

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

Response Body

No Response Body

Example for a Push Account Entries Request (JSON)

Request

```
POST www.example-TPP.com/openFinance-client/v2/ASPSPid/psu-account-id/entries
Content-Type:           application/json
X-Request-ID:           99391c7e-ad88-49ec-a2ad-99ddcb1f7721
Date:                   Sun, 29 Oct 2017 15:02:37 GMT

{
  "account": {"iban": "DE40100100103307118608" },
  "dateTimeLastPush": "2017-10-01T00:00:00Z",
  "staticCallbackText": "high value alarm",
  "transactions":
     [{
       "transactionId": "1234567",
       "creditorName": "John Miles",
       "creditorAccount": {"iban": "DE67100100101306118605"},
```

```
        "transactionAmount": {"currency": "EUR", "amount": "356.67"},
        "bookingDate": "2017-10-25",
        "valueDate": "2017-10-26",
        "remittanceInformationUnstructured": "Example 1"
       },{
        "transactionId": "1234568",
        "debtorName": "Paul Simpson",
        "debtorAccount": {"iban": "NL76RABO0359400371"},
        "transactionAmount": {"currency": "EUR", "amount": "343.01"},
        "bookingDate": "2017-10-25",
        "valueDate": "2017-10-26",
        "remittanceInformationUnstructured": "Example 2"
     }]

}
```

Response

```
HTTP/1.x 204
Content-Type:           application/json
X-Request-ID:           99391c7e-ad88-49ec-a2ad-99ddcb1f7721
Date:                   Sun, 29 Oct 2017 15:04:08 GMT
```

5 References

[XS2A-OR] NextGenPSD2 XS2A Framework, Operational Rules, The Berlin Group Joint Initiative on a PSD2 Compliant XS2A Interface, version 1.3., published December 2019

[oFA-OR-PAIS] openFinance API Framework, Operational Rules, Push Account Information Services, Version 2.0, 13 February 2024

[oFA-PFSM] openFinance API Framework, Protocol Functions and Security Measures, Version 2.0, 5 October 2023

[oFA-DaD] openFinance API Framework, Data Dictionary, Version 2.1,08 February 2024

[oFA-Co] openFinance API Framework, Consent API, Version 2, 05 October 2023

[oFA-AS] Transactions Data Model and Account Statements for Version 2.0 of the openFinance API Framework, Version 1.0, 16 November 2022

[oFA-OR-ADM] openFinance API Framework, Operational Rules for Administrative Service, Version 0.9, published for market consultation 26 May 2021

[oFA-IG-ADM] openFinance API Framework, Implementation Guidelines, Administrative Services, Version 1.0, 08 February 2024

[EBA-RTS] Commission Delegated Regulation (EU) 2018/389 of 27 November 2017 supplementing Directive 2015/2366 of the European Parliament and of the Council with regard to Regulatory Technical Standards for Strong Customer Authentication and Common and Secure Open Standards of Communication, C(2017) 7782 final, published 13 March 2018

[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

[PSD2] Directive (EU) 2015/2366 of the European Parliament and of the Council on payment services in the internal market, published 23 December 2015