



openFinance API Framework
Implementation Guidelines for Extended Services

Push Account Information Services

Version 1.1

03 March 2022

License Notice

This Specification has been prepared by the Participants of the openFinance Taskforce*. 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
2	Character Sets and Notations.....	6
2.1	Additional Notations.....	6
3	Transport Layer.....	7
4	Application Layer: Guiding Principles.....	7
4.1	Signing Messages at Application Layer.....	7
4.2	Additional Error Information.....	7
4.3	Requirements on API Client URIs.....	8
4.4	openFinance API Structure.....	9
4.5	API Access Methods.....	9
4.5.1	Subscription Endpoint.....	10
4.5.2	Push Account Information Endpoint.....	13
4.6	HTTP Response Codes.....	14
4.7	SCA related Hyperlinks.....	15
4.8	Subscription and Subscription Entry Status.....	16
4.9	Resource Availability.....	16
4.10	Model for direct access for Corporates.....	16
5	Subscription for Account Information Service.....	17
5.1	Push Account Information Service Flows.....	18
5.2	Data Overview Push Account Information Service.....	22
5.3	Generic Request.....	27
5.3.1	Example Push Account Entry Service.....	34
5.3.2	Example Push Account Statement Subservice.....	35
5.3.3	Example Push Balance Information Subservice.....	36
5.3.4	Example Push RTP Information Subservice.....	36
5.4	Read Subscription Details.....	37
5.5	Read Subscription Status.....	40
5.6	Add a Subscription Entry.....	42
5.7	Read Subscription Entry Details.....	47
5.8	Read Subscription Entry Status.....	50
5.9	Cancel Subscription Request.....	52



5.10	De-activate Subscription Entry Request	54
6	Push Account Information	56
6.1	Push Account Information Message Flows	56
6.2	Push Account Information Messages.....	56
7	Complex Data Types.....	60
7.1	Subscription Entry.....	60
7.2	Push Account Entry Parameters	61
7.3	Account Entry Trigger Criteria	62
7.4	Push Account Statements Parameters	64
7.5	Account Statement Trigger Criteria	64
7.6	Push Balance Parameters.....	65
7.7	Balance Trigger Criteria.....	65
7.8	pushRtpParameters.....	66
7.9	RTP Trigger Criteria.....	67
7.10	Transactions	67
7.11	Additional Party Information	71
7.14	Payment Condition	72
7.15	Financial Institution Identification	72
7.16	Subscription Status.....	72
7.17	Subscription Entry Status	73
7.18	Credit Debit Code	73
7.19	Account Entry Status	74
7.20	Mime Type Code	74
7.21	Links.....	74
7.22	Lean Account Report.....	77
8	References	79



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 extended 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:

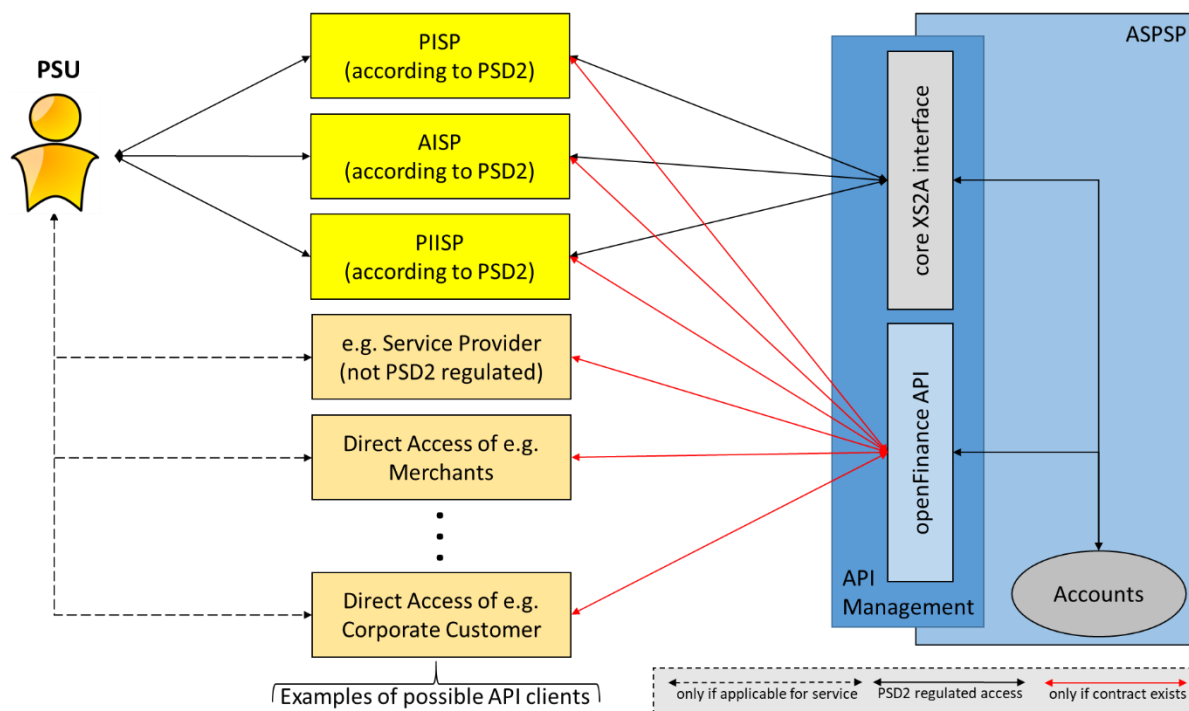


Figure 1: Core XS2A interface and openFinance API

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.

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 and Section 3 give generic information on character sets and the transportation layer. Section 4 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 5 defines the API for subscribing to Push Account Information Services in detail. Section 6 shortly defines the push messages to be used towards the webhooks registered

within the subscription of the related Push Account Information Service. Section 7 then provides the complex data types needed for this 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	<p>Section 4.1: Added Clarifications on Signature Headers</p> <p>Section 4.6: Added clarifications on HTTP response code usage</p> <p>Section 5: Authorization header added to all access methods.</p> <p>New Section 5.1 for description of exemplary flows.</p> <p>New Section 5.2 for a mapping of the abstract data model to API attributes.</p> <p>Section 0: Added a remark on PSU related header fields</p> <p>Section 5.3 as well as 5.6: Several authorisation and status related header fields corrected/added for the request and the related response</p> <p>Section 5.4 GET response: double attribute subscriptionStatus removed.</p> <p>Section 5.7 Corrected Data Type Subscription Entry Status</p> <p>Section 6.2: Added a note that card transactions are not dedicatedly supported</p> <p>Section 7.3: debtorAccount and creditorAccount corrected to singular, Purpose to purpose</p> <p>Section 7.9: Data type of expiryReminder corrected to "Boolean"</p>	15 February 2022

Version	Change/Note	Approved
	Sections 7.11, 7.14, 7.15 Reference to several Data Type from Payment data model added	



2 Character Sets and Notations

For definition on character Sets and Notations as well as for request and response notations refer to Chapter 2 of [XS2A-IG].

2.1 Additional Notations

As an extension of the notations in Chapter 3 in [XS2A-IG], the following conditions may be used. The additional conditions apply to both, requests from the client to the server as well as responses from the server to the client:

Attribute	Type	Condition	Description
		{OR	
		OR	
		OR}	
		{Or – Optional	
		Or – Optional	
		Or – Optional}	

- {Or: The **first** element in a sequence of elements of which **exactly one** has to be included.
- Or: An element in a sequence of elements of which **exactly one** has to be included. The element is **neither the first nor the last** within this sequence.
- Or}: The **last** element in a sequence of elements of which **exactly one** has to be included.
- {Or – Optional: The **first** element in a sequence of elements of which **at most one** may be included.
- Or – Optional: An element in a sequence of elements of which **at most one** may be included. The element is **neither the first nor the last** within this sequence.
- Or – Optional}: The **last** element in a sequence of elements of which **at most one** may be included.

3 Transport Layer

For details on the transport Layer, please refer to Chapter 3 in [XS2A-IG].

Note: The ASPSP is required to use the same web site certificate as client certificate towards the API Client as used as web site certificate in the corresponding API Client – ASPSP XS2A interface as defined in [oFA-OR-ADM].

4 Application Layer: Guiding Principles

4.1 Signing Messages at Application Layer

The same conditions on signing messages by the TPP as defined in [XS2A-IG] apply to the subscription process, cp. Section 5. These conditions also imply the signature related http headers.

4.2 Additional Error Information

Push Account Information service specific error codes are as follows:

Message Code	HTTP Response Code	Description
PRIOR_SUBSCRIPTION_AVAILABLE	409	The PSU/Corporate initiating the Push Account Information Service subscription already holds a subscription in the quadruple API Client/PSU/ASPSP/subservice or API Client/Corporate/ASPSP/subservice respectively or in the triple API Client/ASPSP/subservice in the case of direct access.
SECONDARY_URI_NOT_SUPPORTED	400	An API Client is trying to register a secondary Push URI within a Push Account Information subscription, which is not supported by the ASPSP.
MIME_TYPE_NOT_SUPPORTED	400	An API Client is trying to define a Push Account Information subservice with one or several mime types, where none of the mime types is supported by the ASPSP for this subservice.
LIMIT_FOR_BALANCE_NOT_SUPPORTED	400	The subscription for a Push Balance Information Subservice is asking to involve a credit limit into the balance selection criteria, where the related

Message Code	HTTP Response Code	Description
		balance does not support to include a related credit limit of the account.
ACCOUNT_CURRENCY_NOT_MATCHING	400	The API Client initiated a subscription where an amount is involved which is not matching the currency of the addressed account.
BALANCE_TYPE_NOT_SUPPORTED	400	The API Client initiated a subscription for a balance type that is not supported for the addressed account by the ASPSP.

4.3 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. The only restriction is, that the domain within the URI equals the domain as contained in the API Client eIDAS web site certificate used for identification towards the ASPSP, cp. Section 3. 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-OR-ADM]. 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/v1/ASPSPidentification/psu-account-id/entries
- `pushentries.example-TPP.com/xs2a-client/v1/ASPSPidentification/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.

4.4 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 4.6 They are not repeated for every API call definition.

4.5 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 /v1/subscriptions/{subscriptionId} applies only if the endpoint /v1/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.

4.5.1 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,
- push-account-statements for subscribing to the Push Account Statement Service,
- push-balances for subscribing to the Push Balance Information Service, and
- push-requests-to-pay for subscribing to the Push RTP Information Service.

Endpoint	Method	Condition	Description
subscriptions/{push-accountinformation-subservice}	POST	Mandatory	<p>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.</p> <p>This access method requires an SCA.</p> <p>Cp. Section 0</p>
subscriptions/{push-accountinformation-subservice}/{subscriptionId}	GET	Mandatory	<p>Show information about the activated subscription and the related subscription entries, specifically delivering the entryId of every subscription entry.</p> <p>Cp. Section 5.4</p>
subscriptions/{push-accountinformation-subservice}/{subscriptionId}/status	GET	Mandatory	<p>Show the status of a subscription.</p> <p>Cp. Section 5.5</p>

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 5.9
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 5.6
subscriptions/{push-accountinformation-subservice}/{subscriptionId}/subscription-entries/{subscriptionEntryId}	GET	Mandatory	Show information about the activated subscription entry See Section 5.7
subscriptions/{push-accountinformation-subservice}/{subscriptionId}/subscription-entries/{subscriptionEntryId}/status	GET	Mandatory	Shows the status of a subscription entry. See Section 5.8
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 5.10
subscriptions/{push-accountinformation-subservice}/{subscriptionId}/authorisations	POST	Conditional	Starts an authorisation of the subscription through the PSU. The support of this access method is mandatory in case of the support of the Embedded or Decoupled SCA approach. Cp. [XS2A-IG]
subscriptions/{push-accountinformation-subservice}/{subscriptionId}/authorisations/{authorisationId}	PUT	Mandatory for Embedded SCA Approach, Conditional for other approaches	Update data on the authorisation resource if needed. It may authorise a subscription within the Embedded SCA Approach where needed. Independently from the SCA Approach it supports e.g. the selection of the authentication



Endpoint	Method	Condition	Description
			method and a non-SCA PSU authentication. Cp. [XS2A-IG]
subscriptions/{push-accountinformation-subservice}/{subscriptionId}/authorisations/{authorisationId}	GET	Mandatory	Read the SCA status of the authorisation. Cp. [XS2A-IG]
subscriptions/{push-accountinformation-subservice}/{subscriptionId}/subscription-entries/{subscriptionEntryId}/authorisations	POST	Conditional	Starts an authorisation of adding the related subscription entry through the PSU. The support of this access method is mandatory in case of the support of the Embedded or Decoupled SCA approach. Cp. [XS2A-IG]
subscriptions/{push-accountinformation-subservice}/{subscriptionId}/subscription-entries/{subscriptionEntryId}/authorisations/{authorisationId}	PUT	Mandatory for Embedded SCA Approach, Conditional for other approaches	Update data on the authorisation resource if needed. It may authorise adding the related subscription entry within the Embedded SCA Approach where needed. Independently from the SCA Approach it supports e.g. the selection of the authentication method and a non-SCA PSU authentication. Cp. [XS2A-IG]
subscriptions/{push-accountinformation-subservice}/{subscriptionId}/subscription-entries/{subscriptionEntryId}/authorisations/{authorisationId}	GET	Mandatory	Read the SCA status of the authorisation. Cp. [XS2A-IG]



4.5.2 Push Account Information Endpoint

Endpoint	Method	Condition	Description
<apiClientPrimaryPushURI>	POST	Mandatory	<p>This command is pushing account information of a dedicated Push Account Information subservice to the addressed URI.</p> <p>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 previous call to the openFinance API of the ASPSP.</p> <p>See Section 6.2</p>
<apiClientSecondaryPushURI>	POST	Optional	<p>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.</p> <p>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.</p>



4.6 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 in Section 4.2 for service specific errors as well within [XS2A-IG] for service unspecific errors

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.

4.7 SCA related Hyperlinks

As described in [XS2A-IG], hyperlinks are used in the interface for steering the communication process. Specifically, the process of strong customer authorisation (SCA) is strongly dependent on the ASPSP's implementation and therefore, any response from the ASPSP to a request from the TPP will include one or more SCA related links if the request creates a resource at the TPP that requires further authorisation. For this context, the following links are already defined in [XS2A-IG]:

Link	Description
"scaRedirect"	In case of an SCA Redirect Approach, the ASPSP is transmitting the link to which to redirect the PSU browser.
"scaOAuth"	In case of an OAuth2 based Redirect Approach, the ASPSP is transmitting the link where the configuration of the OAuth2 Server is defined. The configuration follows the OAuth 2.0 Authorisation Server Metadata specification.
"confirmation"	Might be added by the ASPSP if either the "scaRedirect" or "scaOAuth" hyperlink is returned in the same response message. This hyperlink defines the URL to the resource which needs to be updated with <ul style="list-style-type: none"> • a confirmation code as retrieved after the plain redirect authentication process with the ASPSP authentication server or • an access token as retrieved by submitting an authorization code after the integrated OAuth based authentication process with the ASPSP authentication server.
"startAuthorisation"	In case, where an explicit start of the authorisation is needed, but no more data needs to be updated (no authentication method to be selected, no PSU identification nor PSU authentication data to be uploaded)
"startAuthorisation WithPsuIdentification"	The link to the authorisation end-point, where the authorisation sub-resource has to be generated while uploading the PSU identification data.
"startAuthorisation WithPsuAuthentication"	The link to the authorisation end-point, where the authorisation sub-resource has to be generated while uploading the PSU authentication data.
"startAuthorisation WithEncryptedPsuAuthentication"	The link to the authorisation end-point, where the authorisation sub-resource has to be generated while uploading the encrypted PSU authentication data.

Link	Description
"startAuthorisation WithAuthenticationMethodSelection"	The link to the authorisation end-point, where the authorisation sub-resource has to be generated while selecting the authentication method. This link is contained under exactly the same conditions as the data element "scaMethods"
"startAuthorisationWith TransactionAuthorisation"	The link to the authorisation end-point, where the authorisation sub-resource has to be generated while authorising the subscription e.g. by uploading an OTP received by SMS.
"scaStatus"	The link to retrieve the scaStatus of the corresponding authorisation sub-resource. This link is only contained if an authorisation sub-resource has been already created.

4.8 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.

4.9 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.

4.10 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 corporate signatures for proving SCA, performed by signature media issued by the ASPSP to the corporate customer. The related requirements will be defined in a more general definition yet and then also applied to this service in future.

5 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.



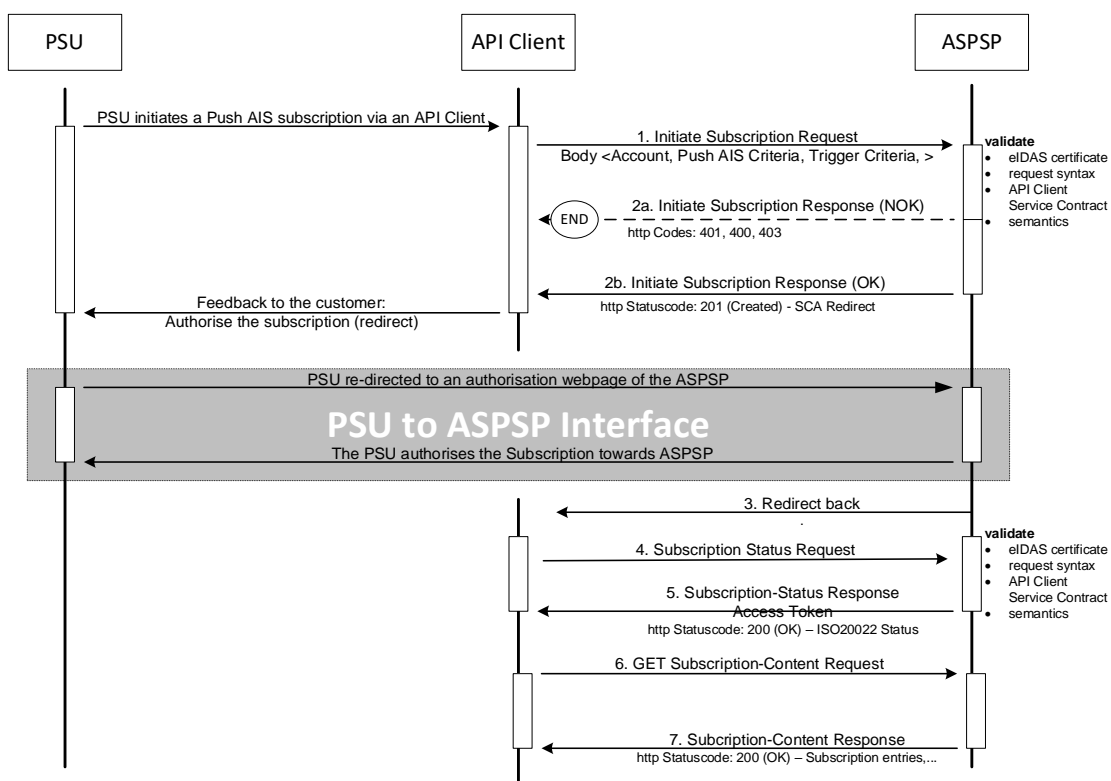
5.1 Push Account Information Service Flows

Please note that the following flows do not cover all possible variances and are exemplary flows, cp. related AIS consent flows in [XS2A-IG].

5.1.1 Push Account Information Subscription Flow

5.1.1.1 Redirect SCA Approach: Implicit Start of the Authorisation Process

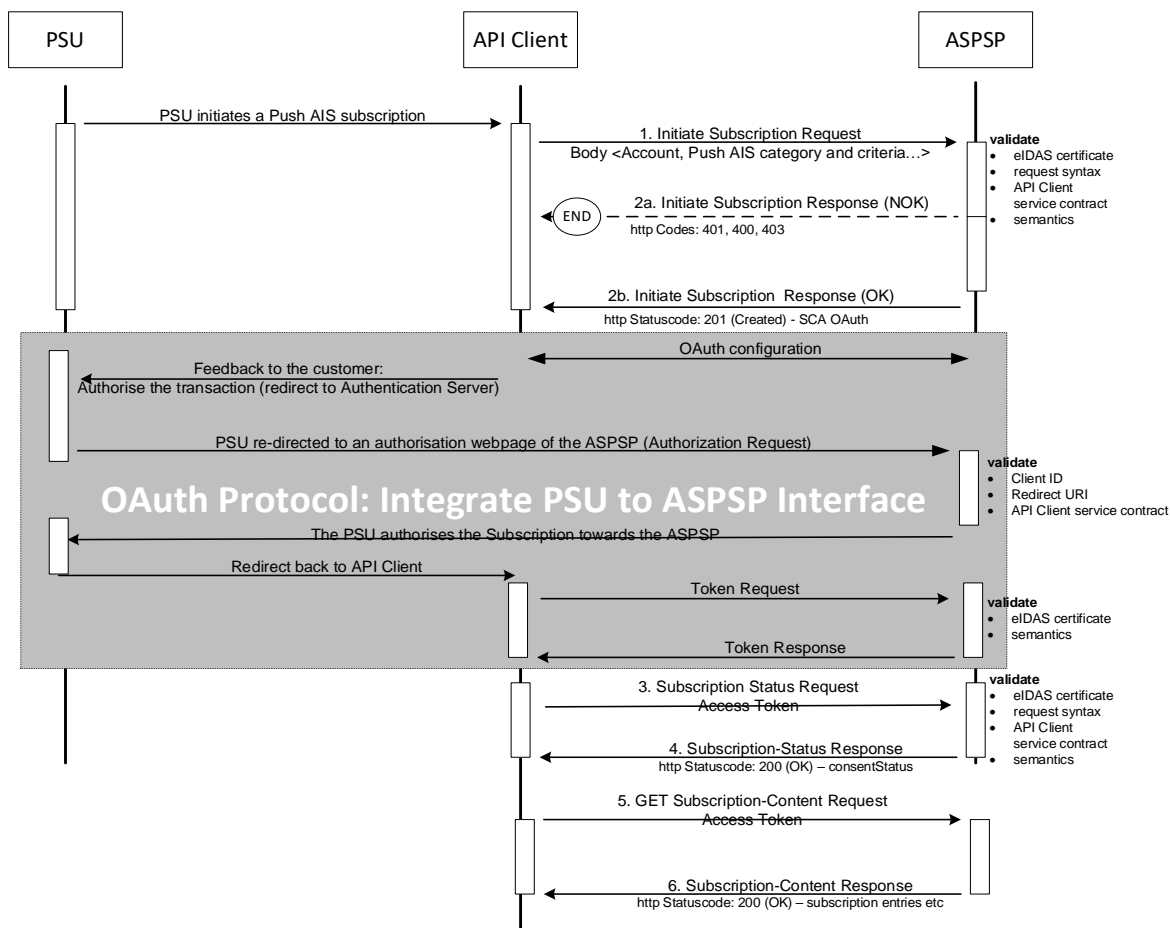
If the ASPSP supports the Redirect SCA Approach, the message flow within the Push Account Information Subscription sub-service is simple. The Push Account Information Subscription Request is followed by a redirection to the ASPSP SCA authorisation site. A status or content request on the created subscription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



5.1.1.2 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.

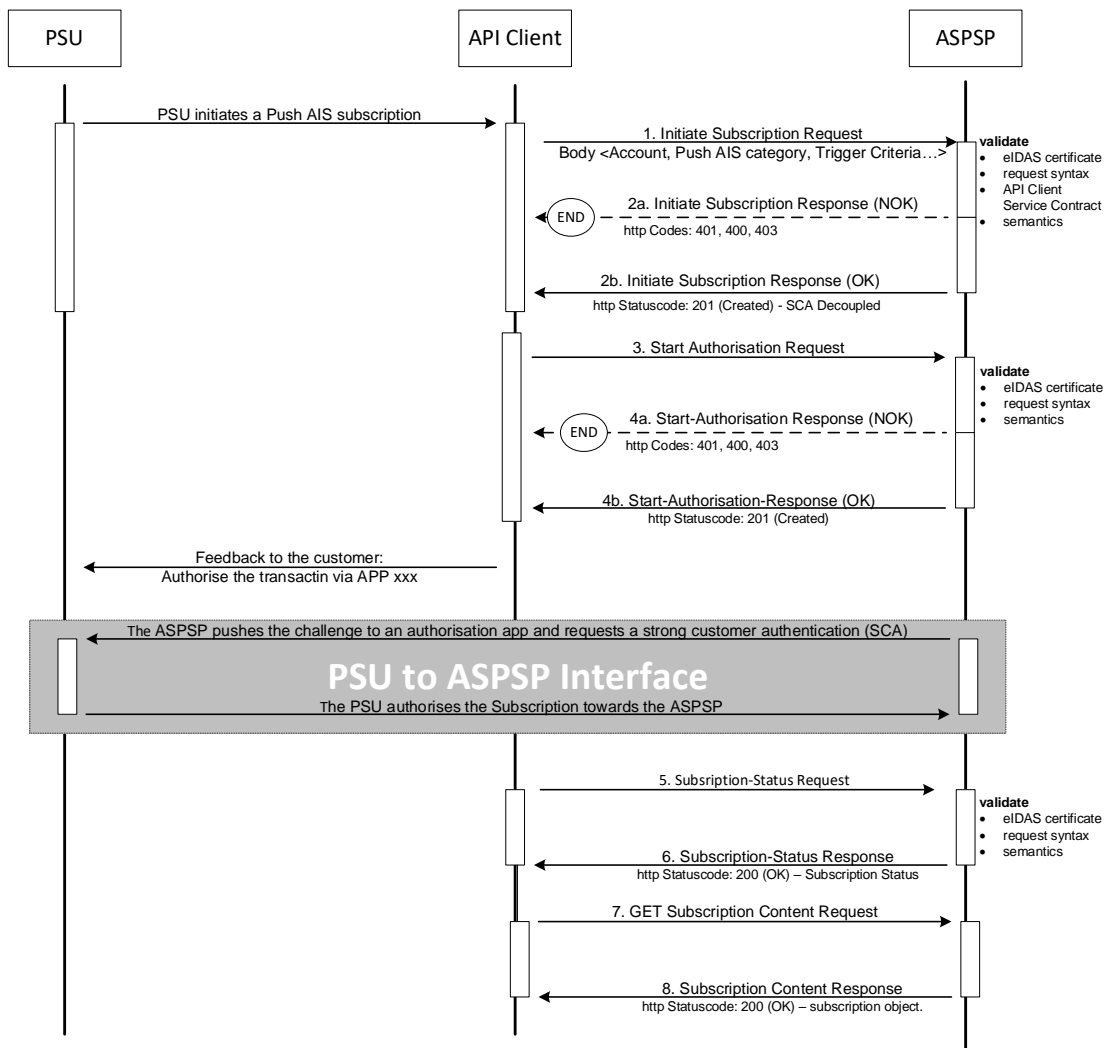


5.1.1.3 Decoupled SCA Approach: Explicit Start of the Authorisation Process

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 subscription 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.



5.1.1.4 Embedded SCA Approach with only one SCA method available

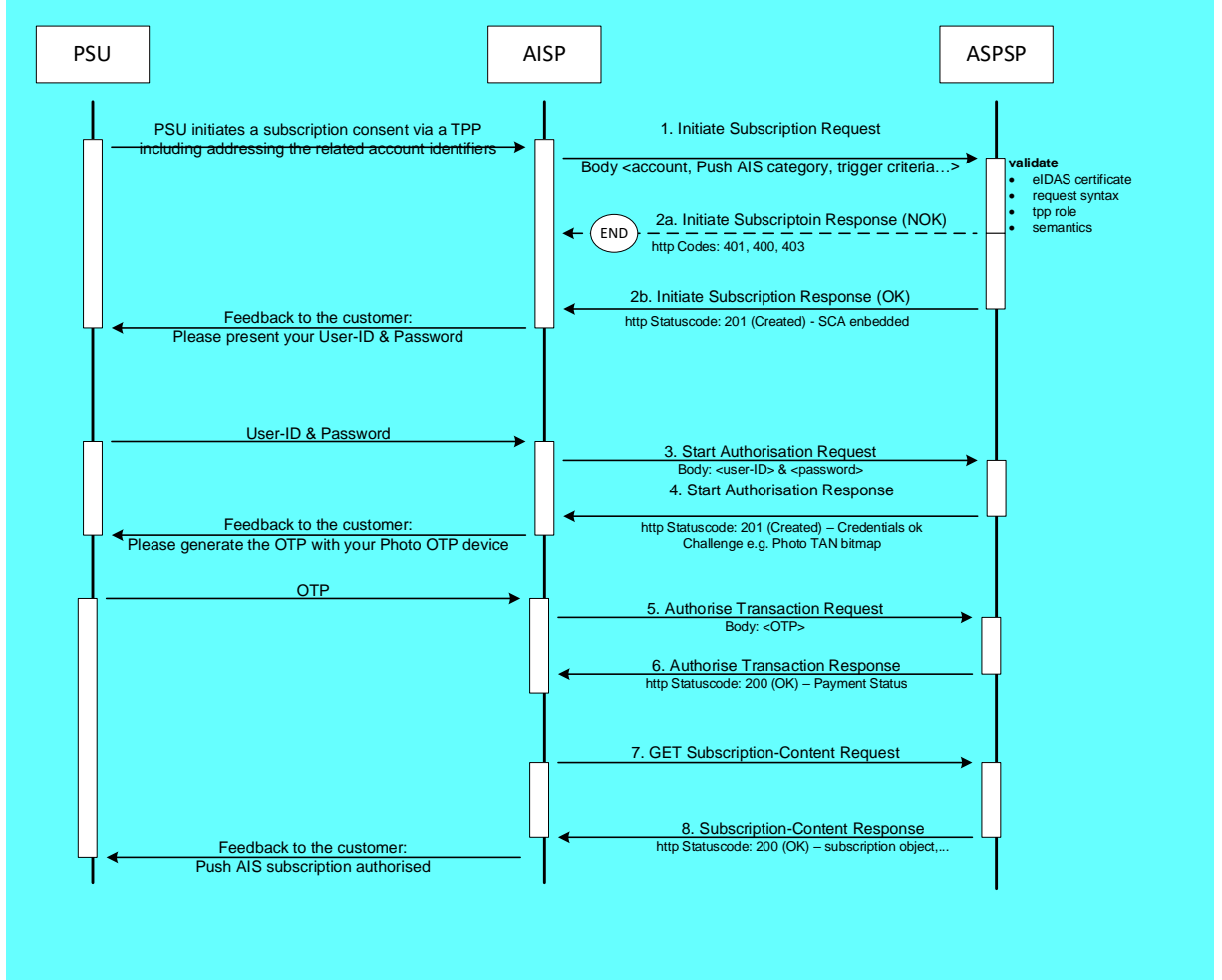
In the following, several exemplary flows are shown, where the ASPSP has chosen to process the SCA methods for the subscription authorisation through the API Client – ASPSP interface.

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.

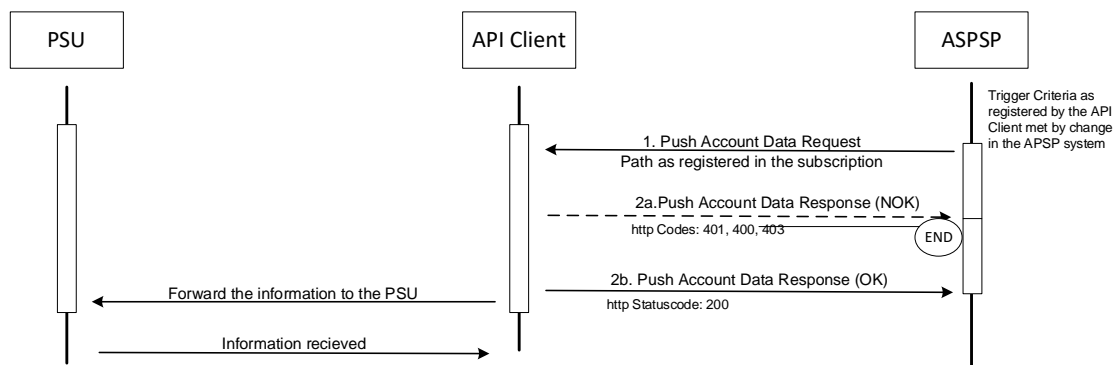


5.1.2 Push Account Information Data Flow

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.



5.2 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 Subscr. 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	subscriptionId				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 Subscr. 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	TPP-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
TPP Message Information	tppMessages				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	
Further PSU related Information	PSU-Accept			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.
	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				
Redirect Preference	Client-Redirect-Preferred			x			o		o			
Decoupled Preference	Client-Decoupled-Preferred			x			o		o			
Redirect URL TPP	TPP-Redirect-URI			x			c		c			
Authorisation Preference	Client-Explicit-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			
Account Information Category		x					m		m		m	
Preference on application layer encryption	encryptionSupported				x		o					
Encryption certificate	encryptionCertificate				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.
Subscription Identification	subscriptionId	x							m		m	
Subscription Entries							m		m			
Account Reference	subscriptionEntries.account				x		m		m			
Validity	subscriptionEntries.validUntil				x		o		o			
Subscription Entry Name	subscriptionEntries.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 accountInformationCategoryCriteria within subscriptionEntries.accountInformationCategoryParameters				x		c		c			
Primary Push URI	subscriptionEntry.apiClientPrimaryPushURI				x		m		m			
Secondary Push URI	subscriptionEntry.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 [XS2A-IG] They are not mentioned anymore in the following

detailed definitions for matter of better readability, as long as the usage is not mandated. Initiate Subscription Request

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 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.

5.3 Generic Request

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.

Path Parameters

Attribute	Type	Condition	Description
push-accountinformation-subservice	String	Mandatory	<p>The addressed subservice endpoint, e.g. for push-account-entries. The default list of sub services supported in this standard is:</p> <ul style="list-style-type: none"> • push-account-entries • push-account-statements • push-balances • push-requests-to-pay <p>The ASPSP will publish which of the subservices will be supported.</p>

Query Parameters

No specific requirements

Request Header

Attribute	Type	Condition	Description
X-Request-ID	UUID	Mandatory	Unique identification of the request.

Attribute	Type	Condition	Description
PSU-ID	String	Conditional	<p>Client ID of the PSU in the ASPSP client interface. Might be mandated in the ASPSP's documentation.</p> <p>It might be contained even if an OAuth2 based authentication was performed in a pre-step. In this case the ASPSP might check whether PSU-ID and token match, according to ASPSP documentation.</p>
PSU-ID-Type	String	Conditional	<p>Type of the PSU-ID; needed in scenarios where PSUs have several PSU-IDs as access possibility.</p> <p>In this case, the mean and use are then defined in the ASPSP's documentation.</p>
PSU-Corporate-ID	String	Conditional	<p>Identification of a Corporate in the Online Channels</p> <p>Might be mandated in the ASPSP's documentation. Only used in a corporate context.</p>
PSU-Corporate-ID-Type	String	Conditional	<p>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.</p> <p>Mean and use is defined in the ASPSP's documentation. Only used in a corporate context.</p>
PSU-IP-Address	String	Mandatory	<p>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.</p> <p>If not available, the TPP shall use the IP Address used by the TPP when submitting this request.</p>
Authorization	String	Conditional	Is contained only if OAuth2 has been used in a pre-step for onboarding the API Client
TPP-Redirect-Preferred	Boolean	Optional	<p>If it equals "true", the TPP prefers a redirect over an embedded SCA approach.</p> <p>If it equals "false", the TPP prefers not to be redirected for SCA. The ASPSP will then choose between the Embedded or the Decoupled SCA approach, depending on the parameter TPP-</p>

Attribute	Type	Condition	Description
			<p>Decoupled-Preferred and the choice of the SCA procedure by the TPP/PSU.</p> <p>If the parameter is not used, the ASPSP will choose the SCA approach to be applied depending on the SCA method chosen by the TPP/PSU.</p>
TPP-Decoupled-Preferred	Boolean	Optional	<p>If it equals "true", the TPP prefers a decoupled SCA approach.</p> <p>If it equals "false", the TPP prefers not to use the decoupled approach for SCA. The ASPSP will then choose between the embedded or the redirect SCA approach, depending on the choice of the SCA procedure by the TPP/PSU.</p> <p>If the parameter is not used, the ASPSP will choose the SCA approach to be applied depending on the parameter TPP-Redirect-Preferred and the SCA method chosen by the TPP/PSU.</p> <p>The parameter might be ignored by the ASPSP.</p> <p>If both parameters TPP-Redirect-Preferred and TPP-Decoupled-Preferred are present and true, the request is still not rejected, but it is up to the ASPSP, which approach will actually be used.</p> <p>RFU: TPP-Redirect-Preferred and TPP-Decoupled-Preferred will be revised in future versions, maybe merged. Currently kept separate for downward compatibility.</p>
TPP-Redirect-URI	String	Conditional	<p>URI of the TPP, where the transaction flow shall be redirected to after a Redirect. Mandated for the Redirect SCA Approach, specifically when TPP-Redirect-Preferred equals "true". See Section 4.3 for further requirements on this header.</p> <p>It is recommended to always use this header field.</p>
TPP-Nok-Redirect-URI	String	Optional	<p>If this URI is contained, the TPP is asking to redirect the transaction flow to this address instead of the TPP-Redirect-URI in case of a negative result of the redirect SCA method. This might be ignored by the ASPSP.</p>

Attribute	Type	Condition	Description
Client-Explicit-Authorisation-Preferred	Boolean	Optional	<p>If it equals "true", the API Client prefers to start the authorisation process separately, e.g. because of the usage of a signing basket. This preference might be ignored by the ASPSP, if a signing basket is not supported as functionality.</p> <p>If it equals "false" or if the parameter is not used, there is no preference of the API Client. This especially indicates that the API Client assumes a direct authorisation of the transaction in the next step, without using a signing basket.</p>
Client-Notification-URI	String	Optional	<p>URI for the Endpoint of the Client-API to which the status of the related resource should be sent.</p> <p>This header field may by ignored by the ASPSP, cp. also the extended service definition in [oFA-RSNS].</p>
Client-Notification-Content-Preferred	String	Optional	<p>The string has the form</p> <p>status=X1, ..., Xn</p> <p>where Xi is one of the constants SCA, PROCESS, LAST and where constants are not repeated.</p> <p>The usage of the constants supports the following semantics:</p> <p>SCA: A notification on every change of the scaStatus attribute for all related authorisation processes is preferred by the API Client.</p> <p>PROCESS: A notification on all changes of subscriptionStatus attributes is preferred by the API Client.</p> <p>LAST: Only a notification on the last subscriptionStatus as available in the openFinance API is preferred by the API Client.</p> <p>This header field may be ignored if the ASPSP does not support resource notification services for the related API Client.</p>
Client-Brand-Logging-Information	String	Optional	<p>This header might be used by the API Client to inform the ASPSP about the brand used by the API Client towards the PSU. This information is meant for logging entries to enhance</p>

Attribute	Type	Condition	Description
			communication between ASPSP and PSU or ASPSP and API Client. This header might be ignored by the ASPSP.
Contract-ID	String	Conditional	ID of the underlying service contract between API Client and ASPSP, resulting from API Client onboarding, following [oFA-OR-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 is provided with a currency that is not supported for this account, the ASPSP will reject the request.
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.

Response Code

The HTTP response code equals 201.

Response Header

Attribute	Type	Condition	Description
ASPSP-Corporate	Boolean	Mandatory	This is a Boolean telling the API Client whether the addressed account is a corporate account (true) or not (false).
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.
ASPSP-SCA-Approach	String	Conditional	<p>This data element must be contained, if the SCA Approach is already fixed. Possible values are:</p> <ul style="list-style-type: none"> • EMBEDDED • DECOUPLED • REDIRECT <p>The OAuth SCA approach will be subsumed by REDIRECT.</p>
ASPSP-Notification-Support	Boolean	Conditional	<p>true if the ASPSP supports resource status notification services.</p> <p>false if the ASPSP supports resource status notification in general, but not for the current request.</p> <p>Not used, if resource status notification services are generally not supported by the ASPSP.</p> <p>Shall be supported if the ASPSP supports resource status notification services, see more details in the extended service definition [oFA-RSNS].</p>
ASPSP-Notification-Content	String	Conditional	<p>The string has the form</p> <p>status=X1, ..., Xn</p> <p>where Xi is one of the constants SCA, PROCESS, LAST and where constants are not repeated.</p> <p>The usage of the constants supports the following semantics:</p> <p>SCA: Notification on every change of the scaStatus attribute for all related authorisation</p>

Attribute	Type	Condition	Description
			<p>processes is provided by the ASPSP for the related resource.</p> <p>PROCESS: Notification on all changes of consentStatus or transactionStatus attributes is provided by the ASPSP for the related resource.</p> <p>LAST: Notification on the last consentStatus or transactionStatus as available in the XS2A interface is provided by the ASPSP for the related resource.</p> <p>This field must be provided if the ASPSP-Notification-Support =true. The ASPSP might consider the notification content as preferred by the API Client, but can also respond independently of the preferred request.</p>

Response Body

Attribute	Type	Condition	Description
subscriptionId	String	Mandatory	Identification of the generated subscription.
subscriptionStatus	Subscription Status	Mandatory	This is the status of the subscription.
_links	Links	Mandatory	<p>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.</p> <p>Remark: All links can be relative or full links, to be decided by the ASPSP.</p> <p>Type of links admitted in this response, (further links might be added for ASPSP defined extensions):</p> <p>"self": The link to the subscription resource created by this request. This link can be used to retrieve the resource data.</p> <p>"status": The link to retrieve the subscription status of the subscription.</p>

Attribute	Type	Condition	Description
			Also, any of the SCA related hyperlinks (cp. 4.7) may occur.

5.3.1 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 "transactionAmount" and "creditorName" from the respective card transaction.

Request

```
POST https://api.testbank.com/v1/subscriptions/push-account-entries
X-Request-ID:          99391c7e-ad88-49ec-a2ad-99ddcb1f7719
Date:                  Sun, 06 Aug 2017 15:03:46 GMT
{
  "subscriptionEntries":
  [
    {"accountId": {"iban": "DE40100100103307118608"},
      "subscriptionEntryName": "Card Debit-Alarm",
      "apiClientPrimaryPushURI": "www.example-TPP.com/openFinance-client/v1/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": "1234-wertiq-983",
  "subscriptionStatus": "received",
  "_links":
  {
    "scaRedirect": {"href": "https://www.testbank.com/asdfasdf"},
    "self": {"href": "/v2/subscriptions/push-account-entries/1234-wertiq-983"},
    "status": {"href": "/v2/subscriptions/push-account-entries/1234-wertiq-983/status"},
    "scaStatus": {"href": "/v2/subscriptions/push-account-entries/1234-wertiq-983/authorisations/123auth456"}
  }
}
```

5.3.2 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/v1/ASPSPid/psu-account-id/account-statements",
    "responseWithLinkPreferred": false,
    "responseWithStaticTextPreferred": false,
    "pushAccountStatementParameters":
    {
      "accountStatementCriteria": {
        "statements": "camt.053",
        "event": "afterGeneration"
      }
    }
  }
]
```

```

    }
  }
}
]

```

5.3.3 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 "interimAvailable" 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/v1/ASPSpid/psu-account-id/entries",
    "responseWithLinkPreferred": false,
    "responseWithStaticTextPreferred": false,
    "pushBalanceParameters":
    {
      "balanceCriteria": {
        "balanceAmount": {"currency": "EUR", "amount": "0.0"},
        "balanceOperator": "less",
        "balanceType": "interimAvailable"}
    }
  }
]

```

5.3.4 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/v1/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
  }
}

```

5.4 Read Subscription Details

Call

GET /v1/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	<p>The addressed subservice endpoint, e.g. for push-account-entries. The default list of sub services supported in this standard is:</p> <ul style="list-style-type: none"> • push-account-entries • push-account-statements • push-balances • push-requests-to-pay, <p>The ASPSP will publish which of the subservices will be supported.</p>
subscriptionId	String	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 registering 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.
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/v1/subscriptions/push-account-entries/qwer3456tzui7890
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": "ID11111111",
      "_links": {
        "entry": {"href": " /v2/subscriptions/push-account-entries/1234-wertiq-983/entries/ID11111111"}
      }
    },
    {
      "accountId": {"iban": "DE40345100103307118789"},
      "subscriptionEntryName": "description 2",
      "subscriptionEntryId": "ID22222222",
      "_links": {
        "entry": {"href": " /v2/subscriptions/push-account-entries/1234-wertq-983/entries/ID22222222"}
      }
    }
  ],
  "encryptionSupported": false,
  "subscriptionStatus": "received"
}
```



5.5 Read Subscription Status

Call

GET /v1/[subscriptions/{push-accountinformation-subservice}/{subscriptionId}/status](#)

Reads the status of the subscription resource.

Path Parameters

Attribute	Type		Description
push-accountinformation-subservice	String	Mandatory	<p>The addressed sub service endpoint, e.g. for push-account-entries. The default list of sub services supported in this standard is:</p> <ul style="list-style-type: none"> • push-account-entries • push-account-statements • push-balances • push-requests-to-pay <p>The ASPSP will publish which of the subservices will be supported.</p>
subscriptionId	String	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 registering 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.
_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/v1/subscriptions/push-account-entries/qwer3456tzui7890/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"}]}
}
```

5.6 Add a Subscription Entry

Call

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

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

Path Parameters

Attribute	Type	Condition	Description
push-accountinformation-subservice	String	Mandatory	<p>The addressed sub service endpoint, e.g. for push-account-entries. The default list of sub services supported in this standard is:</p> <ul style="list-style-type: none"> • push-account-entries • push-account-statements • push-balances • push-requests-to-pay, <p>The ASPSP will publish which of the subservices will be supported.</p> <p>Note: For this version, this specification only supports the service referred to by "push-account-entries".</p>
subscriptionId	String	Mandatory	The subscription object to be retrieved

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 registering the API Client.
PSU-ID	String	Conditional	Client ID of the PSU in the ASPSP client interface. Might be mandated in the ASPSP's documentation.

Attribute	Type	Condition	Description
			It might be contained even if an OAuth2 based authentication was performed in a pre-step. In this case the ASPSP might check whether PSU-ID and token match, according to ASPSP documentation.
PSU-ID-Type	String	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	String	Conditional	Identification of a Corporate in the Online Channels Might be mandated in the ASPSP's documentation. Only used in a corporate context.
PSU-Corporate-ID-Type	String	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.
TPP-Redirect-Preferred	Boolean	Optional	If it equals "true", the TPP prefers a redirect over an embedded SCA approach. If it equals "false", the TPP prefers not to be redirected for SCA. The ASPSP will then choose between the Embedded or the Decoupled SCA approach, depending on the parameter TPP-Decoupled-Preferred and the choice of the SCA procedure by the TPP/PSU. If the parameter is not used, the ASPSP will choose the SCA approach to be applied depending on the SCA method chosen by the TPP/PSU.
TPP-Decoupled-Preferred	Boolean	Optional	If it equals "true", the TPP prefers a decoupled SCA approach. If it equals "false", the TPP prefers not to use the decoupled approach for SCA. The ASPSP will then choose between the embedded or the

Attribute	Type	Condition	Description
			<p>redirect SCA approach, depending on the choice of the SCA procedure by the TPP/PSU.</p> <p>If the parameter is not used, the ASPSP will choose the SCA approach to be applied depending on the parameter TPP-Redirect-Preferred and the SCA method chosen by the TPP/PSU.</p> <p>The parameter might be ignored by the ASPSP.</p> <p>If both parameters TPP-Redirect-Preferred and TPP-Decoupled-Preferred are present and true, the request is still not rejected, but it is up to the ASPSP, which approach will actually be used.</p> <p>RFU: TPP-Redirect-Preferred and TPP-Decoupled-Preferred will be revised in future versions, maybe merged. Currently kept separate for downward compatibility.</p>
TPP-Redirect-URI	String	Conditional	<p>URI of the TPP, where the transaction flow shall be redirected to after a Redirect. Mandated for the Redirect SCA Approach, specifically when TPP-Redirect-Preferred equals "true".</p> <p>It is recommended to always use this header field.</p>
TPP-Nok-Redirect-URI	String	Optional	<p>If this URI is contained, the TPP is asking to redirect the transaction flow to this address instead of the TPP-Redirect-URI in case of a negative result of the redirect SCA method. This might be ignored by the ASPSP.</p>
Client-Explicit-Authorisation-Preferred	Boolean	Optional	<p>If it equals "true", the API Client prefers to start the authorisation process separately, e.g. because of the usage of a signing basket. This preference might be ignored by the ASPSP, if a signing basket is not supported as functionality.</p> <p>If it equals "false" or if the parameter is not used, there is no preference of the API Client. This especially indicates that the API Client assumes a direct authorisation of the transaction in the next step, without using a signing basket.</p>

Attribute	Type	Condition	Description
Client-Brand-Logging-Information	String	Optional	<p>This header might be used by the API Client to inform the ASPSP about the brand used by the API Client towards the PSU. This information is meant for logging entries to enhance communication between ASPSP and PSU or ASPSP and API Client.</p> <p>This header might be ignored by the ASPSP.</p>

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
subscriptionEntry	Subscription Entry	Mandatory	This is a subscription entry.

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.
ASPSP-SCA-Approach	String	Conditional	<p>This data element must be contained, if the SCA Approach is already fixed. Possible values are:</p> <ul style="list-style-type: none"> • EMBEDDED • DECOUPLED • REDIRECT <p>The OAuth SCA approach will be subsumed by REDIRECT.</p>

Response Body

Attribute	Type	Condition	Description
subscriptionEntryId	String	Mandatory	Identification of the created subscription entry within the related subscription.
subscriptionEntryStatus	Subscription Entry Status	Mandatory	This is the status of the subscription entry.
_links	Links	Mandatory	<p>A list of hyperlinks to be recognised by the TPP. The actual hyperlinks used in the response depend on the dynamical decisions of the ASPSP when processing the request.</p> <p>Remark: All links can be relative or full links, to be decided by the ASPSP.</p> <p>Type of links admitted in this response, (further links might be added for ASPSP defined extensions):</p> <p>"self": The link to the subscription entry resource created by this request. This link can be used to retrieve the resource data.</p> <p>"status": The link to retrieve the subscription entry status.</p> <p>Also, any of the SCA related hyperlinks (cp. 4.7) may occur.</p>

Example

Request

POST <https://api.testbank.com/v1/subscriptions/push-account-entries/1234-wertiq-983/subscription-entries>

X-Request-ID: 99391c7e-ad88-49ec-a2ad-99ddcb1f7719

Date: Sun, 06 Aug 2017 15:03:46 GMT

```
{
  "subscriptionEntry":
  {
    "accountId": {"iban": "DE40100100103307118608"},
    "subscriptionEntryName": "description 1",
    "apiClientPrimaryPushURI": "www.example-TPP.com/openFinance-client/v1/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": "1234-wertiq-983",
  "subscriptionEntryId": "ID22222222",
  "subscriptionEntryStatus": "received",
  "_links":
  {
    "scaRedirect": {"href": "https://www.testbank.com/asdfasdfasdf"},
    "self": {"href": "/v2/subscriptions/push-account-entries/
1234-wertiq-983/subscription-entries/ID22222222"},
    "status": {"href": "/v2/subscriptions/push-account-entries/
1234-wertiq-983/subscription-entries/ID22222222/status"},
    "scaStatus": {"href": "/v2/subscriptions/push-account-entries/
1234-wertiq-983/subscription-entries/ID22222222/authorisations/123auth456"}
  }
}
```

5.7 Read Subscription Entry Details

Call

```
GET /v1/subscriptions/{push-accountinformation-
subservice}/{subscriptionId}/subscription-entries/{subscriptionEntryId}
```



Path Parameters

Attribute	Type	Condition	Description
push-accountinformation-subservice	String	Mandatory	<p>The addressed sub service endpoint, e.g. for push-account-entries. The default list of sub services supported in this standard is:</p> <ul style="list-style-type: none"> • push-account-entries • push-account-statements • push-balances • push-requests-to-pay, <p>The ASPSP will publish which of the subservices will be supported.</p> <p>Note: For this version, this specification only supports the service referred to by push-account-entries.</p>
subscriptionId	String	Mandatory	Identification of the addressed subscription.
subscriptionEntryId	String	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 registering 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.
subscriptionEntry	Subscription Entry	Mandatory	Content of the addressed subscription entry: <ul style="list-style-type: none">

Example

Request

```
GET https://api.testbank.com/v1/subscriptions/push-account-entries/qwer3456tzui7890/entries/ID11111111
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",
  "subscriptionEntry": {
    {
      "accountId": {"iban": "DE40100100103307118608"},
      "subscriptionEntryId": "ID11111111",
      "subscriptionEntryName": "Card Debit-Alarm",
      "apiClientPrimaryPushURI": "www.example-TPP.com/openFinance-client/v1/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
}
}
}

```

5.8 Read Subscription Entry Status

Call

```

GET /v1/ subscriptions/{push-accountinformation-
subservice}/{subscriptionId}/subscription-
entries/{subscriptionEntryId}/status

```

Path Parameters

Attribute	Type	Condition	Description
push-accountinformation-subservice	String	Mandatory	<p>The addressed sub service endpoint, e.g. for push-account-entries. The default list of sub services supported in this standard is:</p> <ul style="list-style-type: none"> • push-account-entries • push-account-statements • push-balances • push-requests-to-pay <p>The ASPSP will publish which of the subservices will be supported.</p>
subscriptionId	String	Mandatory	Identification of the addressed subscription.
subscriptionEntryId	String	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 registering 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/v1/subscriptions/push-account-entries/qwer3456tzui7890/entries/ID11111111
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"
}
```

5.9 Cancel Subscription Request

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

Call

```
DELETE /v1/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-entries. The default list of sub services supported in this standard is: <ul style="list-style-type: none"> • push-account-entries • push-account-statements • push-balances • push-requests-to-pay, The ASPSP will publish which of the subservices will be supported.
subscriptionId	String	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.

Attribute	Type	Condition	Description
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 registering 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/v1/subscriptions/push-account-entries/qwer3456tzui7890>

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



5.10 De-activate Subscription Entry Request

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

Call

```
DELETE /v1/subscriptions/push-account-entries/{subscriptionId}/
subscription-entries/{subscriptionEntryId}
```

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: <ul style="list-style-type: none"> • push-account-entries • push-account-statements • push-balances • push-requests-to-pay The ASPSP will publish which of the subservices will be supported.
subscriptionId	String	Mandatory	Identification of the addressed subscription.
subscriptionEntryId	String	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 registering 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/v1/subscriptions/push-account-entries/qwer3456tzui7890>

X-Request-ID 99391c7e-ad88-49ec-a2ad-99ddcb1f7757
Date Sun, 13 Aug 2017 17:05:37 GMT

Response

HTTP/1.x 204 No Content

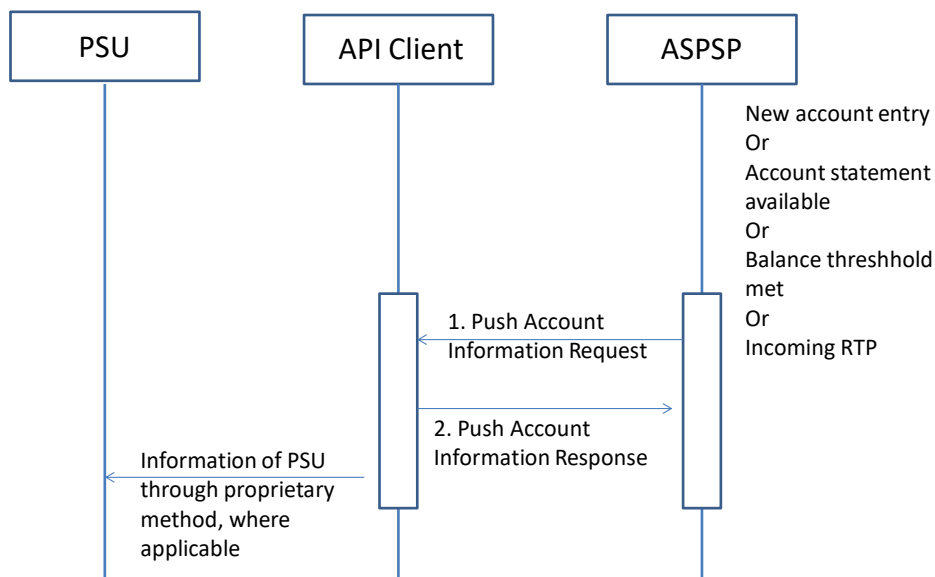
X-Request-ID: 99391c7e-ad88-49ec-a2ad-99ddcb1f7757
Date: Sun, 13 Aug 2017 17:05:47 GMT



6 Push Account Information

6.1 Push Account Information Message Flows

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



6.2 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)

This is not defined yet.

RFU: A format of a JSON based account statement might be defined in future.

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.

RFU: A JSON based statement format will follow in near future.

Request Body for RTP Information (JSON)

Attribute	Type	Condition	Description
dateTimeLastPush	ISODateTime	Optional	
account	Account Reference	Mandatory	
staticCallbackText	Max140	Optional	
_links	Links	Optional	
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/v1/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



7 Complex Data Types

7.1 Subscription Entry

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
subscriptionEntryId	String	Conditional	Forbidden in an <i>Initiate Subscription Request</i> (cp. 0) and in an <i>Add a Subscription Entry Request</i> (cp. 5.6) by the TPP (as this is assigned by the ASPSP). Mandatory in each entry of a response from an 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

Attribute	Type	Condition	Description
			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.
pushAccountStatementParameters	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.

7.2 Push Account Entry Parameters

Attribute	Type	Condition	Description
accountEntryCriteria	Account Entry Trigger Criteria	Optional	This attribute is defining the account entries to be pushed to the API Client push URI: All entries which are fulfilling all the criteria listed in this attribute will be pushed. If the attribute is not provided, all account entries will be pushed.
acceptedFormat	Mime Type Code	Mandatory	This attribute describes the mime type which is supported for pushing the account entries by the API Client. If there is no match with the mime types offered by the ASPSP,

Attribute	Type	Condition	Description
			the subscription is rejected with a dedicated error code.
preferredAttributes	Array of Max35Text	Optional	These attributes are the preferred ones for every account entry to be pushed. The ASPSP might ignore this preference. Only attributes of the first level of Type Transactions as defined in [XS2A-IG] might be used.
documentsPreferred	Boolean	Optional	RFU: This data element indicates that the API Client prefers to also receive documents attached to an account entry. This data attribute might be ignored by the ASPSP.

7.3 Account Entry Trigger Criteria

Attribute	Type	Condition	Description
bankTransactionCodePatterns	Array of Bank Transaction Code Pattern	optional	<p>The criterion is fulfilled if the bankTransactionCode of the related account entry matches one of the bank transaction codes or code patterns (see the following paragraph) defined by this list.</p> <p>Any particle of the Bank Transaction Code triple might be replaced by the "???" character, indicating that all sub family codes apply. However, at least one particle of the triple must be stated explicitly (e.g. "????-CCRD-????").</p>
proprietaryBankCodes	Array of Max35Text	Optional	The criterion is fulfilled if the proprietaryBankTransactionCode of the related account entry equals one of the proprietary bank transaction codes defined by this list.
accountEntryStatus	Account Entry Status	Optional	The criterion is fulfilled if the account entry has the related



Attribute	Type	Condition	Description
			status booked or pending respectively. Note: Entries might be pushed twice in case where this criterion is not provided if the transactionStatus is changed from pending to booked in the account system.
maximumAmount	Amount	Optional	The criterion is fulfilled if the (absolute value of the) transactionAmount of the related account entry is provided in the same currency and if the transactionAmount is less or equal to the amount provided in this attribute.
minimumAmount	Amount	Optional	The criterion is fulfilled if the (absolute value of the) transactionAmount of the related account entry is provided in the same currency and if the transactionAmount is greater or equal to the amount provided in this attribute.
debtorAccount	Account Reference	Optional	The criterion is fulfilled if the counterparty account of the related account entry is a debtor account and equals the account defined in this attribute.
creditorAccount	Account Reference	Optional	The criterion is fulfilled if the counterparty account of the related account entry is a creditor account and equals the account defined in this attribute.
creditDebitIndicator	Credit Debit Code	Optional	The criterion is fulfilled if the related account entry is a credit as defined in this attribute, or a debit respectively.
purpose	Purpose Code	Optional	The criterion is fulfilled if the related account entry has a purpose code



Attribute	Type	Condition	Description
			equal to the code provided in this attribute.
remittanceInformation Unstructured	Max140Text	Optional	The criterion is fulfilled if the related account entry has an unstructured remittance information where the string provided in this attribute is contained as a substring.

7.4 Push Account Statements Parameters

Attribute	Type	Condition	Description
accountStatementCriteria	Account Statement Trigger Criteria	Mandatory	This attribute is defining the account statements to be pushed to the API Client push URI: Every time, the criteria listed in this attribute are all fulfilled, the current account statement of the related account is pushed.

7.5 Account Statement Trigger Criteria

Attribute	Type	Condition	Description
reports	Max35Text	{OR	<p>One of the values ""MT942", "camt.052", shall be used. The criterion is fulfilled if a related report is available.</p> <p>If the related report format is not supported, then the ASPSP is rejecting the subscription with a dedicated error code.</p>
statements	Max35Text	OR}	<p>One of the values "MT940", and "camt.053" shall be used. The criterion is fulfilled if a related statement is available.</p> <p>If the related statement format is not supported, then the ASPSP is rejecting the subscription with a dedicated error code.</p>

Attribute	Type	Condition	Description
event	Max35Text	Optional	Only the value "afterGeneration" may be used. The criterion is fulfilled if the related report is generated and made available to the PSU in the online channel.
times	Array of ISO Time	Optional	The value is provided in format "hh:mm:ss". The criterion is fulfilled if the current local ASPSP time is matching one of the defined times in the list of times. The list shall not be empty.

7.6 Push Balance Parameters

Attribute	Type	Condition	Description
balanceCriteria	Balance Trigger Criteria	Mandatory	This attribute is defining the balance information to be pushed to the API Client push URI: Every time, a balance is fulfilling all the criteria listed in this attribute it is pushed.

7.7 Balance Trigger Criteria

Attribute	Type	Condition	Description
balanceAmount	Amount	Optional	The criterion is fulfilled if the current balance of type balanceType provided below fulfills the balanceOperator requirement provided below together with the provided balanceAmount in this attribute and where the last provided balance did not fulfill it.
balanceOperator	Max35Text	Optional	Must be one of the following codes: <ul style="list-style-type: none"> • less, • lessOrEqual, • equal, • greaterOrEqual,

Attribute	Type	Condition	Description
			<ul style="list-style-type: none"> greater
balanceType	Balance Type	Optional	<p>This criterion is fulfilled if the balance is of this type.</p> <p>If the related balance type is not supported for the addressed account, then the ASPSP is rejecting the subscription with a dedicated error code.</p>
creditLimitIncluded	Boolean	Optional	<p>This criterion is fulfilled if the related current balance is provided with the credit limit included (if this attribute is true) or not (if this attribute is false).</p> <p>If the attribute is true and if the related balance type defined above is not applicable to the condition whether a credit limit is included then the subscription is rejected with a dedicated error code.</p>
times	Array of ISO Time	Optional	<p>The value is provided in format "hh:mm:ss". The criterion is fulfilled if the current local ASPSP time is matching one of the defined times in the list of times.</p> <p>The list shall not be empty.</p>

7.8 pushRtpParameters

Attribute	Type	Condition	Description
rtpCriteria	RTP Trigger Criteria	Optional	<p>This attribute is defining the incoming RTP requests to be pushed to the API Client push URI: All RTPs which are fulfilling all the criteria listed in this attribute will be pushed.</p> <p>If the attribute is not provided, all RTPs will be pushed as soon as</p>

Attribute	Type	Condition	Description
			they are arriving at the ASPSP's system.
acceptedFormat	Mime Type Code	Mandatory	This is defining the mime type. If the requested mime type is not supported by the ASPSP, the request is rejected. If not used, then the ASPSP is choosing the mime type. At a minimum the ASPSP shall support a JSON mime type.
documentsPreferred	Boolean	Optional	RFU: This data element indicates that the API Client prefers to also receive documents attached to an RTP. This data attribute might be ignored by the ASPSP.

7.9 RTP Trigger Criteria

Attribute	Type	Condition	Description
event	Max35Text	Optional	"afterReception" is the only supported value.
timeToExpire	Integer	Optional	only forward incoming RTP with an expiry timestamp within that period, where the period is defined by the number of minutes as provided in this attribute. This criterion is meant to forward only short term RTPs.
expiryReminder	Boolean	Optional	This criterion is fulfilled if a long term RTP is still not confirmed 2 days before the RTP expires.

7.10 Transactions

This is an extension of the current Transactions data type as provided by [XS2A-IG]. Additional attributes are marked by **yellow color**.

Attribute	Type	Condition	Description
transactionId	String	Optional	Can be used as access-ID in the API, where more details on a transaction is offered. If this data attribute is provided this shows that the AIS can get access on more details about this transaction using the GET Transaction Details Request as defined in Section 6.5.5
entryReference	Max35Text	Optional	Is the identification of the transaction as used e.g. for reference for deltafunction on application level. The same identification as for example used within camt.05x messages.
batchIndicator	Boolean	Optional	If this indicator equals <code>true</code> , then the related entry is a batch entry.
batchNumberOf Transactions	Integer	Conditional	Shall be used if and only if the <code>batchIndicator</code> is contained and equals <code>true</code> .
endToEndId	Max35Text	Optional	Unique end to end identity.
mandateId	Max35Text	Optional	Identification of Mandates, e.g. a SEPA Mandate ID
checkId	Max35Text	Optional	Identification of a Cheque
creditorId	Max35Text	Optional	Identification of Creditors, e.g. a SEPA Creditor ID
bookingDate	ISODate	Optional	The Date when an entry is posted to an account on the ASPSPs books.
valueDate	ISODate	Optional	Date at which assets become available to the account owner in case of a credit entry, or cease to be available to the account owner in case of a debit entry. Usage: If entry status is pending and value date is present, then the value date refers to an expected/requested value date.
transactionAmount	Amount	Mandatory	The amount of the transaction or batch as billed to the account.
currencyExchange	Array of Report	Optional	



Attribute	Type	Condition	Description
	Exchange Rate		
creditorName	Max70Text	Optional	Name of the creditor if a "Debited" transaction
creditorAdditional Information	Additional Party Information	Optional	
creditor Account	Account Reference	Conditional	
creditorAgent	BICFI	Optional	
ultimate Creditor	Max70Text	Optional	
ultimateCreditor AdditionalInformation	Additional Party Information	Optional	
debtorName	Max70Text	Optional	Name of the debtor if a "Credited" transaction
debtorAccount	Account Reference	Conditional	
debtorAgent	BICFI	Optional	
ultimateDebtor	Max70Text	Optional	
remittance Information Unstructured	Max140Text	Optional	
remittance Information Unstructured Array	Array of Max140Text	Optional	Remark for Future: In version 2.0 these two unstructured remittance fields might be merged.
Remittance Information Structured	Max140Text	Optional	Reference as contained in the structured remittance reference structure (without the surrounding XML structure). Remark For Future: This field will be re-typed in a future version of the interface to the structured data type Remittance or

Attribute	Type	Condition	Description
			might be omitted. For migration reasons, this is not supported in version 1.3.x.
remittance Information Structured Array	Array of Remittance	Optional	<p>NOTE:</p> <p>More details about the Remittance Data Type will be published in an Errata in due course.</p> <p>For usage of the fields e.g. for domestic elements, Berlin Group should be contacted. This would enable to publish usage of structured remittance information in the domestic payment documentation, cp. [XS2A-DP].</p>
entryDetails	Array of Entry Details	Optional	Might be used by the ASPSP to transport details about transactions within a batch.
additionalInformation	Max500Text	Optional	Might be used by the ASPSP to transport additional transaction related information to the PSU
additionalInformation Structured	Structured Additional Information	Conditional	Is used if and only if the bookingStatus entry equals "information". Every active standing order related to the dedicated payment account result into one entry.
purposeCode	Purpose Code	Optional	
bank TransactionCode	Bank Transaction Code	Optional	<p>Bank transaction code as used by the ASPSP and using the sub elements of this structured code defined by ISO20022.</p> <p>For standing order reports the following codes are applicable:</p> <p>"PMNT-ICDT-STDO" for credit transfers,</p> <p>"PMNT-IRCT-STDO" for instant credit transfers</p> <p>"PMNT-ICDT-XBST" for cross-border credit transfers</p> <p>"PMNT-IRCT-XBST" for cross-border real time credit transfers and</p>

Attribute	Type	Condition	Description
			"PMNT-MCOP-OTHR" for specific standing orders which have a dynamical amount to move left funds e.g. on month end to a saving account
proprietaryBankTransactionCode	Max35Text	Optional	proprietary bank transaction code as used within a community or within an ASPSP e.g. for MT94x based transaction reports For RTP, the value "incomingRTP" is used.
balanceAfterTransaction	Balance	Optional	This is the balance after this transaction. Recommended balance type is interimBooked.
_links	Links	Optional	The following links could be used here: transactionDetails for retrieving details of a transaction.

7.11 Additional Party Information

This data type is defined in [oFA-PDM-V2].

7.12 Structured Additional Information

This is an extension of the current Transactions data type as provided by [XS2A-IG]. Additional attributes are marked by **yellow color**.

Attribute	Type	Condition	Description
standingOrderDetails	Standing Order Details	{OR	Details of underlying standing orders.
rtpDetails	RTP Details	OR}	Details of underlying incoming RTPs

7.13 RTP Details

Attribute	Type	Condition	Description
paymentConditions	Payment Conditions	optional	Conditions for the execution of the payment.
creationDateTime	ISODatetime	optional	creationDateTime of the originator
requestedExpiryDateTime	ISODatetime	mandatory	Date and Time at which this request expires.
originatorPspId	Financial Institution Identification	Optional	Originator payment service provider in case of indirect participation regarding a related API Access Scheme or bilateral contract

7.14 Payment Condition

This data type is defined in [oFA-PDM-V2].

7.15 Financial Institution Identification

This data type is defined in [oFA-PDM-V2].

7.16 Subscription Status

Code	Description
received	The subscription data have been received and are technically correct. The data is not authorised yet.
rejected	The subscription data have been rejected e.g. since no successful authorisation has taken place.
partiallyAuthorised	The subscription is due to a multi-level authorisation, some but not all mandated authorisations have been performed yet.
valid	The subscription is accepted and valid for the related subscribed services as specified in the subscription object.
validInChange	The subscription is accepted and valid for the related subscribed services as specified in the subscription object. A new subscription entry has been submitted which is not yet (fully) authorised.
revokedByPsu	The subscription has been revoked by the PSU towards the ASPSP.
cancelledByAspsp	The subscription has been cancelled by the ASPSP.

Code	Description
expired	The subscription expired.
terminatedByTpp	The corresponding TPP has terminated the subscription by applying the DELETE method to the subscription resource.

The ASPSP might add further codes. These codes then shall be contained in the ASPSP's documentation of the XS2A interface.

7.17 Subscription Entry Status

Code	Description
received	The subscription data have been received and are technically correct. The data is not authorised yet.
rejected	The subscription data have been rejected e.g. since no successful authorisation has taken place.
partiallyAuthorised	The subscription is due to a multi-level authorisation, some but not all mandated authorisations have been performed yet.
valid	The subscription is accepted and valid for the related subscribed service as specified in the subscription object.
revokedByPsu	The subscription has been revoked by the PSU towards the ASPSP.
expired	The subscription expired.
terminatedByTpp	The corresponding TPP has terminated the subscription by applying the DELETE method to the subscription resource.

7.18 Credit Debit Code

Code	Description
CRDT	Operation is an increase.
DBIT	Operation is a decrease.

7.19 Account Entry Status

Code	Description
booked	Entry is booked on the account.
pending	Entry is pending, funds might be reserved or not.

7.20 Mime Type Code

Code	Description
application/JSON	Mime type for body encoding in JSON syntax.
application/XML	Mime type for body encoding in XML syntax.
plain/text	Mime type for body encoding in text.

7.21 Links

The structure of Links is compliant to [HAL]. There are some new entries (marked in yellow color) relative to the definition in [XS2A-IG]. Please note that now also Arrays of links are supported for certain link types.

Attribute	Type	Condition	Description
scaRedirect	href Type	Optional	A link to an ASPSP site where SCA is performed within the Redirect SCA approach.
scaOAuth	href Type	Optional	The link refers to a JSON document specifying the OAuth details of the ASPSP's authorisation server. JSON document follows the definition given in [RFC8414]
confirmation	href Type	Optional	"confirmation": Might be added by the ASPSP if either the "scaRedirect" or "scaOAuth" hyperlink is returned in the same response message. This hyperlink defines the URL to the resource which needs to be updated with <ul style="list-style-type: none"> a confirmation code as retrieved after the plain redirect authentication process with the ASPSP authentication server or an access token as retrieved by submitting an authorization code after the integrated OAuth based authentication process with the ASPSP authentication server.

Attribute	Type	Condition	Description
startAuthorisation	href Type	Optional	A link to an endpoint, where the authorisation of a transaction or the authorisation of a transaction cancellation shall be started with a POST command. No specific data is needed for this process start.
startAuthorisationWithPsuIdentification	href Type	Optional	The link to an endpoint where the authorisation of a transaction or of a transaction cancellation shall be started, where PSU identification shall be uploaded with the corresponding call.
updatePsuIdentification	href Type	Optional	The link to the payment initiation or account information resource, which needs to be updated by the PSU identification if not delivered yet.
startAuthorisationWithProprietaryData	hrefType	Optional	<p>A link to the endpoint, where the authorisation of a transaction or of a transaction cancellation shall be started, and where proprietary data needs to be updated with this call. The TPP can find the scope of missing proprietary data in the ASPSP documentation.</p> <p>The usage of this hyperlink is not further specified in the specification but is used analogously to e.g. the startAuthorisationWithPsuIdentification hyperlink.</p>
updateProprietaryData	href Type	Optional	The link to the payment initiation or account information resource, which needs to be updated by the proprietary data.
startAuthorisationWithPsuAuthentication	href Type	Optional	The link to an endpoint where the authorisation of a transaction or of a transaction cancellation shall be started, where PSU authentication data shall be uploaded with the corresponding call.
updatePsuAuthentication	href Type	Optional	The link to the payment initiation or account information resource, which needs to be updated by a PSU password and eventually the PSU identification if not delivered yet.
updateAdditionalPsuAuthentication	href Type	Optional	The link to the payment initiation or account information resource, which needs to be updated by an additional PSU password.
startAuthorisationWithEncryptedPsuAuthentication	href Type	Optional	The link to an endpoint where the authorisation of a transaction or of a transaction cancellation



Attribute	Type	Condition	Description
			shall be started, where encrypted PSU authentication data shall be uploaded with the corresponding call.
updateEncryptedPsuAuthentication	href Type	Optional	The link to the payment initiation or account information resource, which needs to be updated by an encrypted PSU password and eventually the PSU identification if not delivered yet.
updateAdditionalEncryptedPsuAuthentication	href Type	Optional	The link to the payment initiation or account information resource, which needs to be updated by an additional encrypted PSU password.
startAuthorisationWithAuthenticationMethodSelection	href Type	Optional	This is a link to an endpoint where the authorisation of a transaction or of a transaction cancellation shall be started, where the selected SCA method shall be uploaded with the corresponding call.
selectAuthenticationMethod	href Type	Optional	This is a link to a resource, where the TPP can select the applicable second factor authentication methods for the PSU, if there were several available authentication methods.
startAuthorisationWithTransactionAuthorisation	href Type	Optional	A link to an endpoint, where an authorisation of a transaction or a cancellation can be started, and where the response data for the challenge is uploaded in the same call for the transaction authorisation or transaction cancellation at the same time in the Embedded SCA Approach.
authoriseTransaction	href Type	Optional	The link to the payment initiation or consent resource, where the "Transaction Authorisation"Request" is sent to. This is the link to the resource which will authorise the payment or the consent by checking the SCA authentication data within the Embedded SCA approach.
self	href Type	Optional	The link to the payment initiation resource created by the request itself. This link can be used later to retrieve the transaction status of the payment initiation.
status	href Type	Optional	A link to retrieve the status of the transaction resource.

Attribute	Type	Condition	Description
scaStatus	href Type	Optional	A link to retrieve the status of the authorisation or cancellation-authorisation sub-resource.
account	href Type	Optional	A link to the resource providing the details of one account
balances	href Type	Optional	A link to the resource providing the balance of a dedicated account.
transactions	href Type	Optional	A link to the resource providing the transaction history of a dedicated account.
cardAccount	href Type	Optional	A link to the resource providing the details of one card account.
cardTransactions	href Type	Optional	A link to the resource providing the transaction history of a dedicated card account.
transactionDetails	href Type	Optional	A link to the resource providing details of a dedicated transaction.
first	href Type	Optional	Navigation link for paginated account reports.
next	href Type	Optional	Navigation link for paginated account reports.
previous	href Type	Optional	Navigation link for paginated account reports.
last	href Type	Optional	Navigation link for paginated account reports.
download	href Type	Optional	Download link for huge AIS data packages.
entryStatusRevoked	Array of href Type	Optional	Links to entry status endpoints where the entry status is revoked.

7.22 Lean Account Report

Attribute	Type	Condition	Description
booked	Array of transactions	Conditional	booked entries on the related account
pending	Array of transactions	Optional	pending entries on the related account

Attribute	Type	Condition	Description
information	Array of transactions	Optional	information entries related to the addressed account.



8 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
- [XS2A-IG] NextGenPSD2 XS2A Interoperability Framework, Implementation Guidelines, The Berlin Group Joint Initiative on a PSD2 Compliant XS2A Interface, version 1.3.11, published 24 September 2021
- [oFA-PDM-V2] openFinance API Framework, Payment Data Model for Version 2.x, version 1.0, 24 September 2021
- [oFA-OR-PAIS] openFinance API Framework, Operational Rules, Push Account Information Services, Version 1.0, published 21 December 2021
- [oFA-OR-ADM] openFinance API Framework, Operational Rules for Administrative Service, Version 0.9, published for market consultation 26 May 2021
- [oFA-RSNS] openFinance API Framework, Extended Services, Resource Status Notification Service, Version 1.2., published 24 September 2021
- [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
- [HAL] Kelley, M., "HAL - Hypertext Application Language", 2013-09-18, http://stateless.co/hal_specification.html
- [RFC2426] Dawson, F. and T. Howes, T., "vCard MIME Directory Profile", September 1998, <https://tools.ietf.org/html/rfc2426>
- [RFC3230] Mogul, J. and A. Van Hoff, "Instance Digests in HTTP", RFC 3230, DOI 10.17487/RFC3230, January 2002, <https://www.rfc-editor.org/info/rfc3230>
- [RFC4648] Josefsson, S., "The Base16, Base32, and Base64 Data Encodings", October 2006, <https://tools.ietf.org/html/rfc4648>
- [RFC5843] Bryan, A, "Additional Hash Algorithms for HTTP Instance Digests", RFC 5843, DOI 10.17487/RFC5843, April 2010, <https://www.rfc-editor.org/info/rfc5843>



- [RFC6749] Hardt, D., "The OAuth 2.0 Authorization Framework", October 2012, <https://tools.ietf.org/html/rfc6749>
- [RFC7807] M. Nottingham, Akamai, E. Wilde, „Problem Details for HTTP APIs", March 2016, <https://tools.ietf.org/html/rfc7807>

